

SKRIPSI

**RANCANG BANGUN SISTEM INFORMASI *E-LAUNDRY*
DENGAN IMPLEMENTASI BERBASIS *WEB*
(*PROGRAMMING*)**



NAMA : ELANG SETIA RYANANDA

NIM : 1755201013

**PROGRAM STUDI S1 TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PAHLAWAN TUANKU TAMBUSAI
RIAU
2021**

SKRIPSI

RANCANG BANGUN SISTEM INFORMASI *E-LAUNDRY* DENGAN IMPLEMENTASI BERBASIS *WEB* (*PROGRAMMING*)



NAMA : ELANG SETIA RYANANDA

NIM : 1755201013

Diajukan Sebagai Persyaratan untuk Memperoleh Gelar Sarjana S1 Teknik Informatika

**PROGRAM STUDI S1 TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PAHLAWAN TUANKU TAMBUSAI
RIAU
2021**

BAB I

PENDAHULUAN

A. Latar Belakang

Teknologi Informasi dan Komunikasi pada masa sekarang ini telah mengalami perkembangan dengan pesat. Hal ini diakui oleh banyaknya aktivitas-aktivitas kehidupan manusia yang mencapai standar baru. Kecepatan dan ketepatan menjadi syarat utama dari segala bentuk proses dan kebutuhan yang dilakukan oleh manusia, terlebih dalam kebutuhan Informasi. Media yang paling banyak digunakan dalam memenuhi kebutuhan Informasi dan Komunikasi tersebut adalah komputer dan telepon seluler yang semakin disempurnakan dengan adanya Internet. Internet menjembatani perpindahan informasi dari berbagai tempat tanpa dibatasi ruang dan waktu dalam waktu yang singkat.

Website merupakan media yang digunakan untuk menampung data teks, gambar, suara, dan animasi yang dapat ditampilkan di internet dan dapat diakses oleh komputer yang terhubung dengan internet secara global. *Website* juga merupakan media informasi berbasis jaringan komputer yang dapat diakses dimana saja dengan biaya relatif murah dan *website* dapat mengimplementasikan Bahasa pemrograman (*Web Programming*). PHP (*Hypertext Preprocessor*) merupakan Bahasa pemrograman berbasis *Web* yang memiliki kemampuan untuk memproses dan mengolah data secara dinamis.

Dampak positif teknologi di dunia bisnis kini tidak hanya bisa dirasakan oleh bisnis dengan skala besar. Usaha kecil menengah yang dijalankan dengan melibatkan teknologi informasi dan komunikasi dalam menjalankan usahanya

akan dapat mempermudah para pelaku usaha untuk menunjang aktivitas bisnisnya. Keterlibatan teknologi informasi dan komunikasi dalam hal ini akan membuat usaha menjadi lebih mudah, lebih cepat, dan lebih dapat diandalkan untuk meminimalkan adanya kesalahan manusia.

Teknologi internet saat ini, masyarakat semakin di mudahkan dalam melakukan segala macam fasilitas dan proses. Salah satu contoh yang akan penulis rancang adalah mencari Jasa Laundry di *Kota Bangkinang*, dimana masyarakat di Kota Bangkinang tidak jarang merasa kesulitan mencari jasa laundry tersebut, berdasarkan Hasil dari Survei terdapat 52 Laundry yang tersebar di Kota Bangkinang, dengan adanya Sistem informasi *E-Laundry* penulis berharap dapat meringankan pekerjaan masyarakat dengan jasa antar jemput karena pada umumnya masyarakat Bangkinang bekerja dari pagi sampai sore dan ada juga Siswa dan Mahasiswa yang masih menyewa tempat tinggal. Penulis juga ingin menambahkan bahwa sistem *E-Laundry* terdapat notifikasi apabila pesanan customer sudah selesai.

Bisnis Laundry sebagai bisnis yang berjalan dibidang jasa dirasa akan lebih mudah apabila memasukkan unsur Teknologi dan Informasi didalamnya. Berdasarkan hal tersebut, untuk itu peneliti mengambil judul **“RANCANG BANGUN SISTEM INFORMASI *E-LAUNDRY* DENGAN IMPLEMENTASI BERBASIS *WEB*“**

Perancangan *Website E-Laundry* yang digunakan oleh pengguna jasa Laundry seperti pemesanan jasa, antar jemput Laundry, mendata pakaian masuk dan mendata pelanggan baru secara tepat waktu untuk dikirimkan kepada admin.

Website ini diharapkan dapat membantu menjalankan bisnis Laundry secara efisien dan memberikan kemudahan baik kepada Costumer untuk bertransaksi dengan pemilik Laundry maupun kepada admin untuk melakukan pengolahan data.

B. Identifikasi Masalah

Identifikasi masalah dalam penelitian ini sebagai berikut:

1. Kesulitan masyarakat dalam mendapatkan Informasi tentang pelayanan usaha Laundry,
2. Belum adanya Sistem Informasi berbasis *Website* yang memungkinkan pengusaha Laundry resmi dan rumahan menjalankan usahanya,
3. Kurangnya perkembangan dalam Informasi yang masyarakat dapatkan mengenai adanya web *E-Laundry*.

C. Batasan Masalah

Batasan masalah dalam penelitian ini adalah sebagai berikut:

1. Membuat Sistem Informasi *E-Laundry* berbasis *web* dengan menggunakan Bahasa pemograman PHP,
2. Mempromosikan UMKM (Usaha Mikro Kecil Menengah) Laundry kepada masyarakat luas.

D. Rumusan Masalah

1. Bagaimana fungsi sebuah Sistem Informasi *E-Laundry* berbasis *web*?
2. Bagaimana penyebaran secara luas Informasi mengenai *Web E-Laundry*?

E. Tujuan Penelitian

Tujuan dari penelitian ini adalah sebagai berikut:

1. Mempermudah penyebaran informasi bagi customer maupun pemilik laundry,
2. Memberikan Informasi tentang jasa *E-Laundry* dalam bentuk *Website* yang dapat diakses oleh customer sehingga customer dapat berinteraksi dengan pemilik Laundry,

F. Manfaat Penelitian

Manfaat dari penelitian ini adalah sebagai berikut:

1. Dapat meringankan pekerjaan masyarakat yang sibuk bekerja,
2. Membantu UMKM (Usaha Mikro Kecil Menengah) dalam mempromosikan usaha Laundry kepada masyarakat luas,
3. Memperkenalkan usaha di dunia digital kepada masyarakat (*E- Commerce*).

BAB II TINJAUAN PUSTAKA

A. Tinjauan Pustaka

Hasil penelitian terdahulu digunakan untuk mengetahui persamaan dan perbedaan dari beberapa penelitian yang ada, Berikut adalah beberapa penelitian terdahulu yang digunakan dalam penelitian ini, yaitu:

1. Penelitian tentang layanan manajemen usaha Laundry telah di buat oleh Sadam Sidiq, (2012), yang membahas aplikasi *software as a service* layanan manajemen usaha Laundry yang bersifat *online* untuk pengelolaan usaha laundry akan perangkat lunak dan mampu memberikan pelayanan yang inovatif sesuai kebutuhan usaha Laundry yang senantiasa berkembang. Sistem ini dibangun menggunakan bahasa pemograman *PHP, Framework CodeIgniter* dan *DBMS MySQL*.
2. Penelitian tentang Sistem Transaksi Laundry pernah dilakukan oleh Brian Nurranto (2014), tetapi berbeda dengan penelitian yang dilakukan oleh Sadam Sidiq, dkk, penelitian ini bertujuan untuk untuk menghasilkan suatu sistem informasi berbasis web untuk mempermudah pengoperasian *Protect Laundry* dalam mengelola kegiatan usahanya yang menggunakan metode perhitungan perpakaian ditambah lagi antara anggota dan pihak *Protect Laundry*, khususnya dalam hal proses pengerjaan pakaian yang dilaundry pemberian informasi status pengerjaan melalui *website*.
3. Penelitian Wulandari, Eva (2015) membuat penelitian dengan tujuan untuk penulis membuat dan merancang dalam menghasilkan sistem informasi ini

penulis melakukan proses identifikasi masalah, pengumpulan data, analisa sistem, perancangan sistem, pembangunan sistem, uji coba sistem dan implementasi sistem. Rekayasa Perangkat Lunak menggunakan model *Waterfall*, yang digunakan dalam perancangan dan pembangunan sistem adalah *VB Net 2010* dan databasenya dengan *SQL Server*, untuk metode perancangan dengan menggunakan *DFD*

B. Kerangka Teori

Berikut merupakan kerangka teori pendukung dalam penelitian ini, yaitu sebagai berikut:

1. Pengertian Rancang Bangun

Rancang merupakan salah satu hal yang penting dalam membuat program. Tujuan dari perancangan ialah untuk memberi gambaran yang jelas lengkap kepada pemrogram dan ahli teknik yang terlibat. Perancangan harus berguna dan mudah dipahami sehingga mudah digunakan. Perancangan adalah sebuah proses untuk mendefinisikan sesuatu yang akan dikerjakan dengan menggunakan teknik yang bervariasi serta di dalamnya melibatkan deskripsi mengenai arsitektur serta detail komponen dan juga keterbatasan yang akan dialami dalam proses pengerjaannya.

Perancangan atau rancang menurut pendapat Pressman (2009) adalah serangkaian prosedur untuk menterjemahkan hasil analisa dan sebuah sistem ke dalam bahasa pemrograman untuk mendeskripsikan dengan detail bagaimana komponen-komponen yang ada di dalam sistem di implementasikan.

Bangun menurut pendapat Pressman (2009) adalah kegiatan menciptakan sistem baru maupun mengganti atau memperbaiki sistem yang telah ada secara keseluruhan. Jadi dapat disimpulkan bahwa Rancang Bangun adalah penggambaran, perencanaan, dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah kedalam suatu kesatuan yang utuh dan berfungsi.

Rancang bangun merupakan kegiatan menerjemahkan hasil analisa ke dalam bentuk paket perangkat lunak kemudian menciptakan sistem tersebut atau memperbaiki system yang sudah ada (Syukroni, 2017).

2. Implementasi

Pengertian implementasi yang dikemukakan oleh Pranata Wastra dan kawan-kawan (1991;256) adalah: “Aktivitas atau usaha-usaha yang dilakukan untuk semua rencana dari kebijaksanaan yang telah dirumuskan dan ditetapkan, dan dilengkapi segala kebutuhan alat-alat yang diperlukan, siapa yang melaksanakan, dimana tempat pelaksanaannya, kapan waktu pelaksanaannya, kapan waktu mulai dan berakhirnya dan bagaimana cara yang harus dilaksanakan”. Kata implementasi berasal dari bahasa Inggris yaitu *to implement* yang berarti mengimplementasikan. Implementasi adalah penyediaan sarana untuk melaksanakan sesuatu yang menimbulkan dampak atau akibat terhadap sesuatu. Sesuatu tersebut dilakukan untuk menimbulkan dampak atau akibat itu dapat berupa undang-undang, peraturan pemerintah, keputusan peradilan dan kebijakan yang dibuat oleh lembaga- lembaga pemerintah dalam kehidupan kenegaraan. Pengertian

implementasi selain menurut Webster di atas dijelaskan juga oleh Van Meter dan Van Horn dalam Wahab (2004:65), yaitu implementasi adalah tindakan-tindakan yang dilakukan baik oleh individu-individu/pejabat-pejabat atau kelompok-kelompok pemerintah atau swasta yang diarahkan pada tercapainya tujuan-tujuan yang telah digariskan dalam keputusan kebijakan.(Lestari et al., 2020).

3. RPL (Rekayasa Perangkat Lunak)

Perangkat lunak (*Software*) adalah program komputer yang terasosiasi dengan dokumentasi preangkat lunak seperti dokumentasi kebutuhan, model sistem, dan cara penggunaannya. Rekayasa perangkat lunak (RPL) merupakan pembangunan sebuah perangkat lunak dengan tujuan menghasilkan perangkat lunak yang bernilai ekonomis yang di percaya dan bekerja secara efisien menggunakan mesin. RPL lebih fokus pada praktek pengembangan perangkat lunak yang bermanfaat bagi pelanggan (*User*) dengan memenuhi kriteria sebagai berikut (Hendrajati, 2013) :

- a. Dapat terus dipelihara setelah perangkat lunak selesai dibuat seiring berkembangnya teknologi dan lingkungan (*Maintainability*).
- b. Dapat diandalkan dengan proses bisnis yang dijalankan dan perubahan yang terjadi (*Dependability dan Robust*).
- c. Efisiensi dari segi sumber daya dan penggunaan.
- d. Kemampuan untuk dipakai sesuai dengan kebutuhan (*Usability*) Jadi perangkat lunak yang baik adalah perangkat lunak yang fokus kepada pengguna atau pelanggan.

Biasanya, tidak ada perbedaan pada bidang rekayasa lain sehingga bisa diartikan apabila bidang rekayasa akan selalu berusaha untuk memperoleh output yang memiliki kinerja lebih tinggi, biaya lebih rendah serta waktu untuk menyelesaikan yang lebih cepat. Berikut akan dijelaskan tujuan dari RPL selengkapnya:

- a. Memperoleh biaya produksi perangkat lunak atau *software* yang lebih kecil.
- b. Untuk menghasilkan perangkat lunak yang mempunyai kinerja tinggi, handal dan juga tepat waktu.
- c. Untuk menghasilkan perangkat lunak atau *software* yang dapat bekerja pada segala jenis platform.
- d. Untuk menghasilkan perangkat lunak yang memiliki biaya perawatan murah serta terjangkau.

Ruang lingkup dari rekayasa perangkat lunak terdiri dari beberapa hal, yakni:

- a. *Software requirement*: Berhubungan dengan spesifikasi kebutuhan serta persyaratan perangkat lunak.
- b. *Software desain*: Mencakup proses penampilan arsitektur, antarmuka, komponen serta karakteristik lainnya dari perangkat lunak atau *software*.
- c. *Software construction*: Berhubungan dengan rincian pengembangan perangkat lunak seperti pengkodean, algoritma, pengujian sampai mencari kesalahan.

- d. *Software testing*: Meliputi pengujian pada seluruh perilaku dari perangkat lunak.
- e. *Software maintenance*: Mencakup semua jenis usaha merawat ketika perangkat lunak sudah dioperasikan atau digunakan.
- f. *Software configuration management*: Berhubungan dengan semua jenis usaha untuk merubah konfigurasi perangkat lunak yakni memenuhi kebutuhan tertentu.
- g. *Software engineering tools and methods*: Meliputi kajian teoritis tentang alat bantu serta metode RPL.

4. Website

Website atau situs dapat diartikan sebagai kumpulan halaman-halaman yang digunakan untuk menampilkan informasi teks, gambar diam atau gerak, animasi, suara, dan atau gabungan dari semuanya baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait, yang masing-masing dihubungkan dengan jaringan-jaringan halaman. Hubungan antara satu halaman web dengan halaman web yang lainnya disebut *hyperlink*, sedangkan teks yang dijadikan media penghubung disebut *hypertext* (Batubara, 2012).

Jenis-jenis web berdasarkan sifat atau stylenya yaitu: *Website* Dinamis, merupakan sebuah *website* yang menyediakan konten atau isi yang selalu berubah-ubah setiap saat. Bahasa pemrograman yang digunakan antara lain *PHP*, *ASP*, *NET* dan memanfaatkan *database MySQL* atau *MS*

SQL. Misalnya *website* www.artikel.com, www.detik.com, www.tecnomobile.co.cc, dan lain-lain.(Batubara, 2012)

Website Statis, merupakan *website* yang kontennya sangat jarang diubah. Bahasa pemrograman yang digunakan adalah *HTML* dan belum memanfaatkan *database*. Misalnya: *web Profile* organisasi, dan lain-lain.(Batubara, 2012).

Memulai pembuatan website seorang *web designer* akan menanyakan tujuan Anda membuat *website*. Apakah akan digunakan sebagai blog pribadi, *company profile*, toko *online*, portal berita dan sebagainya, berikut penjelasan mengenai *website statis* dan dinamis :

a. Website Statis

Halaman *website* statis tidak akan mengalami perubahan konten (isi *website*) maupun *layout* saat terjadi permintaan data ke *web server*. Halaman hanya akan berubah jika pengelola melakukan perubahan terhadap konten *website* secara manual. Perubahan biasanya dilakukan menggunakan bantuan program *text editor* atau program desain *website* seperti *Adobe Dreamweaver*. Setiap informasi yang ditampilkan di *website* statis diatur dengan bahasa markup seperti *HTML*. Perubahan sekecil apapun, maka hanya bisa dilakukan oleh *web designer* atau *web programmer*. Halaman *website* statis jauh lebih sederhana, cenderung lebih aman dari serangan hacker, tidak rentan dari kesalahan teknis, dan lebih mudah dibaca oleh mesin pencari (*search engine*) seperti *Google*.

Kelebihan *website* statis:

- 1) Cepat untuk dikembangkan
- 2) Efisien untuk dikembangkan
- 3) Efisien saat disimpan pada web hosting

Kekurangan *website* statis

- 1) Membutuhkan tenaga ahli khusus untuk melakukan perubahan
- 2) Situs menjadi kurang bermanfaat bagi pengunjung
- 3) Konten *website* yang usang

b. *Website* Dinamis

Website dinamis memiliki konten yang bisa beradaptasi menyesuaikan interaksi dari pengunjung. Perubahan pada data dijalankan oleh aplikasi secara berkala, seperti pada *website* berita. Perubahan-perubahan tersebut bisa dilakukan dengan teknik *client-side scripting*, menggunakan bahasa pemrograman web (*Javascript*, *ActionScript*, *DHTML*). *Client-side scripting* konten pada halaman *website* bisa dirubah dengan cepat pada komputer pengunjung tanpa meminta halaman baru ke web server. Kebanyakan *website* dinamis dikembangkan dan dijalankan pada *web server* dengan bahasa pemrograman *server-side scripting* (*ASP*, *JSP*, *Perl*, *PHP*, *Python*). Baik bahasa pemrograman *client* maupun *server-side* digunakan pada *website* yang selalu mengalami perubahan konten dan memiliki fitur interaktif yang kompleks. *Website* dinamis menawarkan fleksibilitas yang banyak, tapi proses untuk menjalankan halaman *website* dengan

gabungan konten yang unik pada setiap halaman membutuhkan web server canggih berkecepatan tinggi. Bahkan server yang memadai pun bisa kewalahan jika mendapatkan banyak permintaan data dari *website* dinamis dalam waktu yang singkat. Konten dalam *website* dinamis harus dioptimasi secara khusus agar lebih ramah dengan *search engine*, ini merupakan kekurangan lain dari *website* dinamis.

Kelebihan *website* dinamis:

- 1) Memiliki banyak fitur tambahan
- 2) Lebih mudah untuk dikembangkan
- 3) Lebih mudah menambah konten baru yang menarik pengunjung dari *search engine*
- 4) Memiliki fitur yang memudahkan pengelola dan pengguna untuk bekerjasama

Kekurangan *website* dinamis:

- 1) Membutuhkan waktu lama dan dana lebih untuk pengembangan
- 2) Membutuhkan biaya hosting lebih mahal

5. Usaha Laundry

Usaha Laundry adalah suatu jenis usaha yang bergerak di bidang jasa cuci dan setrika. Bisnis ini termasuk dalam kategori bisnis dengan perputaran yang cepat. Maksudnya rentang waktu permintaan pelanggan antara permintaan pertama dan permintaan selanjutnya pada jasa ini yang memakan waktu relatif singkat. Lebih jelasnya, pelanggan akan kembali menggunakan jasa ini ketika pakaian yang dikenakan sudah kotor. Selain itu

bisnis Laundry juga merupakan kategori bisnis yang berkelanjutan atau sebagai bisnis yang akan selalu dibutuhkan banyak orang. Manusia akan selalu mencuci pakaian selama kebutuhan sandang masih menjadi kebutuhan primer bagi manusia.(Lubis et al., 2019).

6. *Start Up*

Startup merupakan perusahaan berbasis teknologi informasi yang menyediakan barang atau jasa baik melalui media online atau offline. Startup memiliki karakteristik tersendiri yang membedakannya dengan Usaha Kecil Menengah (UKM) atau bisnis lainnya, dimana *Startup* merupakan suatu bisnis yang belum lama berdiri, tenaga kerjanya masih sangat sedikit yaitu 16 di bawah 20 orang. SDMnya mampu untuk bekerja secara *multitasking*, dikerjakan oleh usia produktif (20-35 tahun), pendapatan masih tergolong kurang tetapi masih sanggup untuk bertahan, serta bergerak dibidang teknologi dan *website* (Lubis et al., 2019).

Kelebihan *Start Up* :

- a. Waktu bekerja yang fleksibel
- b. Suasana yang tidak kaku
- c. Rekan kerja seusia sehingga mudah diajak bekerja sama
- d. Pekerjaan yang menantang
- e. Lingkungan yang sangat nyaman dan positif

Kekurangan *Start Up* :

- a. Mengerjakan pekerjaan tambahan yang bukan tugasnya
- b. Bekerja hingga malam hari dan dihubungi klien saat hari libur

- c. Pekerjaan yang bisa datang tiba-tiba
- d. Fasilitas yang diperoleh tidak sebanyak di perusahaan konvensional
- e. Keuangan perusahaan yang belum stabil

7. **PHP**

PHP merupakan bahasa pemrograman yang digunakan untuk membuat aplikasi berbasis *website*. Sebagai sebuah aplikasi, *website* tersebut hendaknya memiliki sifat dinamis dan interaktif. Memiliki sifat dinamis artinya, *website* tersebut bisa berubah tampilan kontennya sesuai kondisi tertentu (misalnya, menampilkan produk yang berbeda-beda untuk setiap pengunjung). Interaktif artinya, *website* tersebut dapat memberi *feedback* bagi *user* (misalnya, menampilkan hasil pencarian produk). Kode pemrograman standar untuk mendesain *website* sebenarnya adalah *HTML*, namun masalahnya, tanpa memanfaatkan *PHP*, kode *HTML* tidak dapat digunakan untuk membuat *website* dengan dua sifat di atas. Dengan kata lain, jika hanya mengandalkan *HTML5* saja, tampilan situs yang dilihat oleh para pengunjung satu dan lainnya relatif sama dan statis. Celah inilah yang kemudian dapat diatasi oleh *PHP*. Dengan memanfaatkan *PHP*, Anda bisa membuat sistem *username login shopping cart database* keanggotaan, portal, dan sebagainya (Enterprise, 2017).

Keunggulan beserta kelemahan yang dimiliki *PHP*, kelebihan yang dimiliki *PHP* sebagai berikut:

- a. Lebih mudah dipelajari

PHP memang jauh lebih mudah dipelajari, karena cukup dengan melakukan konfigurasi. Setelah itu penggunanya bisa menjadikan sebagai bahasa pemrograman pada tingkat *entry level*. Tingkat tersebut berada dalam tahap belajar, sehingga para pemula pengembangan *website* tidak akan merasa kesulitan. Bahkan untuk tutorial dalam mempelajari pemrograman *PHP* bisa didapatkan dengan mudah, baik itu di toko buku, melalui *online*, belajar di lembaga bimbingan khusus untuk pengembangan *website* ataupun bisa juga melalui *YouTube*. Kemudahan dalam mempelajari *PHP* membuat daya tarik tersendiri bagi masyarakat yang masih awam.

b. Mempunyai *Community* besar

Banyak sekali *website* yang seringkali menggunakan *PHP* untuk dijadikan sebagai bahasa pemrograman dalam melakukan pembuatan aplikasi *website*. Salah satu contoh *website* yang sudah terkenal dalam penggunaan *PHP* yaitu seperti *Yahoo*, *Facebook*, *Wikipedia* ataupun yang lainnya. Hal ini dikarenakan *PHP* mempunyai *community* yang cukup besar. Dengan adanya forum tersebutlah, maka akan lebih memudahkan para anggotanya dalam membahas sekaligus bertukar pikiran pada pemrograman *PHP* yang saat ini memang sudah banyak bermunculan di berbagai situs. Contohnya adalah grup *PHP* Indonesia di *Facebook*. Bagi mahasiswa jurusan IT, *PHP* dijadikan bahasa pemrograman, selain itu untuk mahasiswa jurusan *website development* juga akan mempelajari bahasa pemrograman *PHP*.

c. Lebih ringkas

Keunggulan lain dari bahasa pemrograman *PHP* adalah para *programer website* akan merasakan kemudaha tersendiri karena *PHP* dianggap jauh lebih ringkas. Salah satunya yaitu dari segi proses instalnya yang tidak membutuhkan setting secara berlebihan. Itu karena untuk konfigurasi pada *database* juga bisa dilakukan dengan cara yang lebih mudah. Proses pengembangan tidak membutuhkan waktu yang lama, maka sudah bisa membuktikan bahwa proses pembuatan *PHP* menjadi lebih praktis dan ringkas. Kemudahan itu sangat berbeda jika dibandingkan pada bahasa pemrograman yang lain, karena bahasa pemrograman yang lain memiliki proses konfigurasi agar bisa melihat *website* yang sudah terselesaikan.

d. *Open Source*

PHP sebagai bahasa pemrograman yang *open source*, maka para pengguna bisa menggunakan *PHP* dengan sangat bebas dan gratis. Para pengguna tidak diharuskan memberi royalti berbentuk apapun. Akan tetapi para penggunanya berkewajiban memiliki *PHP* atau license yang dimiliki oleh *PHP Group*. Memiliki perkembangan yang cukup cepat proses pembuatan aplikasi melalui *PHP* ini dianggap jauh lebih cepat jika dibandingkan dengan mengembangkan pada suatu aplikasi *website* dengan bahasa pemrograman yang lainnya. Di dalam bahasa pemrograman *PHP* ini memiliki perkembangan yang cukup cepat, karena memiliki banyak sekali *boiler, tools*, yang sudah

disediakan pada PHP secara *open source*. Dengan perkembangan yang cukup cepat inilah maka untuk proses yang didapatkan pun juga lebih singkat. Baik itu dari *start* bahkan sampai dengan *finish* pada sebuah project dalam pembuatan aplikasi *Web*.

- e. Memiliki *Maintenance* yang lebih mudah

Web yang sudah menggunakan bahasa pemrograman *PHP* proses *update* bisa dilakukan dengan sangat mudah dan baik. Hal ini karena *PHP* memiliki sifat interpreter. Selain itu, untuk melakukan *upgrade* pada versi yang baru penggunaanya tidak perlu melakukan konfigurasi secara ulang. Sementara pada bahasa pemrograman yang lainnya, harus menentukan konfigurasi ulang apabila melakukan *upgrade* pemrograman yang baru. Terlebih lagi *PHP* juga mampu berjalan di berbagai jenis *web server*, misalnya seperti *nginx*, *apache* dan *IIS*.

Kekurangan yang dimiliki *PHP* sebagai berikut:

- a. Memiliki banyak kompetisi

PHP memang memiliki banyak komunitas, yang mana hal ini justru membawa kompetisi yang tergolong lebih ketat. Para *web developer* akan semakin bertambah setiap harinya. Para pebisnis yang berharap produknya bisa terkenal, bisa menggunakan *PHP* dan harus mengembangkan aplikasi bisnisnya agar dapat bersaing secara kompetitif.

- b. Tidak dilengkapi dengan tipe data

PHP ini tidak mempunyai tipe data, sehingga pada bahasa pemrograman sering muncul bug-bug yang tidak berkaitan dengan standar tipe data *PHP*. Sementara untuk kekurangan yang satu ini memiliki hubungan terhadap pengalaman *developer* yang sedang menggunakan *PHP* sebagai bahasa pemrograman.

c. Rawan pembajakan

Sifat dari *PHP* adalah interpreter, sehingga cenderung lebih rawan terhadap pembajakan. Salah satu penyebabnya yaitu *shortcode* pada aplikasi bahasa pemrograman *PHP* ini dapat diubah dan dimodifikasi dengan mudah. Sebab itulah *PHP* kurang tepat digunakan sebagai pengembang aplikasi.

d. Kurang prestesius

Dibandingkan pada bahasa pemrograman yang lainnya, untuk *PHP* ini dianggap kurang prestisius, karena memiliki kesan yang kurang aman maupun memiliki celah yang terbatas. Penyebabnya yaitu karena pengembang belum mengetahui secara penuh tentang standar dalam pembuatan aplikasi menggunakan *PHP* yang benar.

8. Sistem Informasi

Sistem adalah kumpulan dari komponen-komponen yang memiliki unsur keterkaitan antara satu dan lainnya. informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi penerimanya. Menurut Dani Eko Hendrianto (2014) Sistem Informasi adalah suatu sistem didalam suatu organisasi yang mempertemukan kebutuhan pengolahan data

transaksi harian, mendukung operasi, bersifat manajerial, dan kegiatan strategi dari suatu organisasi serta menyediakan pihak luar tertentu dengan laporan- laporan yang diperlukan (Simargolang & Nasution, 2018).

9. XAMPP

XAMPP Menurut Buana (2014:4), “XAMPP adalah perangkat lunak *open source* yang diunggah secara gratis dan bisa dijalankan di semua semua operasi seperti *windows, linux, solaris, dan mac*” Menurut Nugroho (2013), di dalam folder utama *xampp*, terdapat beberapa folder penting yang perlu diketahui. Untuk lebih memahami setiap fungsinya, Anda dapat melihat penjelasannya sebagai berikut (Siradjuddin, 2018).

Tabel 2. 1 Keterangan folder utama Xampp

Folder	Keterangan
<i>Apache</i>	Folder utama dari <i>Apache Web Server</i>
<i>Htdocs</i>	Folder utama untuk menyimpan data- data latihan web, baik <i>PHP</i> maupun <i>HTML</i> biasa
<i>Manual</i>	Berisi sub folder yang di dalamnya terdapat manual program dan <i>database</i> , termasuk manual <i>PHP</i> dan <i>MySQL</i>
<i>MYSQL</i>	Folder utama untuk <i>database MySQL Server</i> . <i>PHP</i> Folder utama untuk program <i>PHP</i>

Sumber: Nugroho 2013

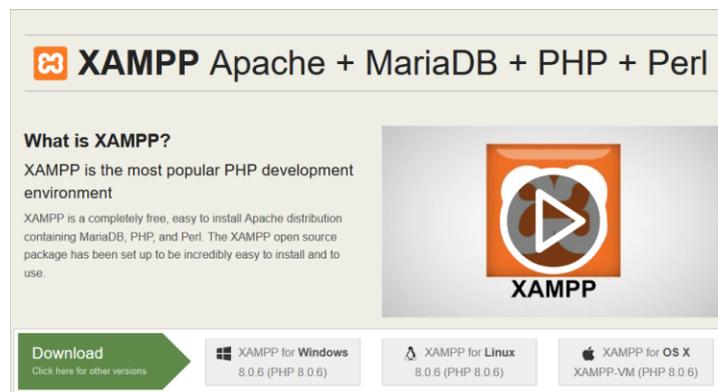
XAMPP adalah perangkat lunak bebas, yang mendukung banyak sistem operasi, merupakan kompilasi dari beberapa program. Fungsinya adalah sebagai server yang berdiri sendiri (*localhost*), yang terdiri atas program Apache HTTP Server, MySQL database, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan Perl. Nama XAMPP merupakan singkatan dari X (empat sistem operasi apapun), Apache, MySQL, PHP dan Perl. Program ini tersedia dalam GNU General Public License dan bebas, merupakan web server yang mudah digunakan yang

dapat melayani tampilan halaman web yang dinamis. Untuk mendapatkannya dapat mendownload langsung dari web resminya. (Riyadli et al., 2020) .

Cara penginstalan *XAMPP* sebagai berikut:

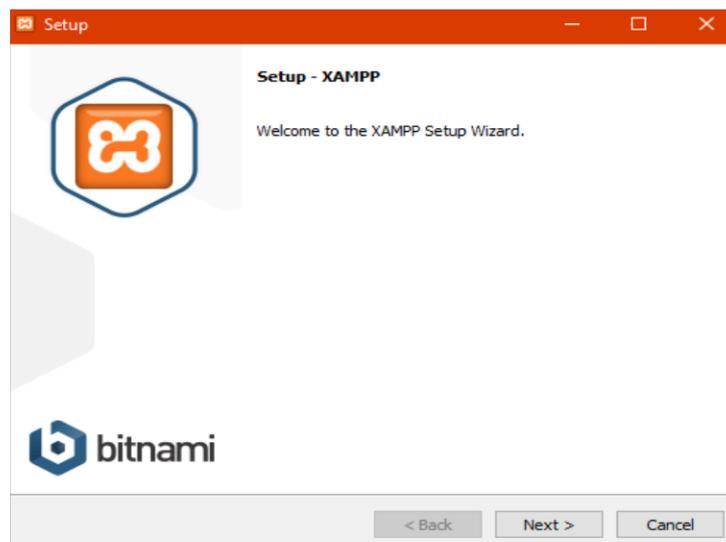
a. Download *Installer XAMPP* melalui *website*

<https://www.apachefriends.org/index.html>



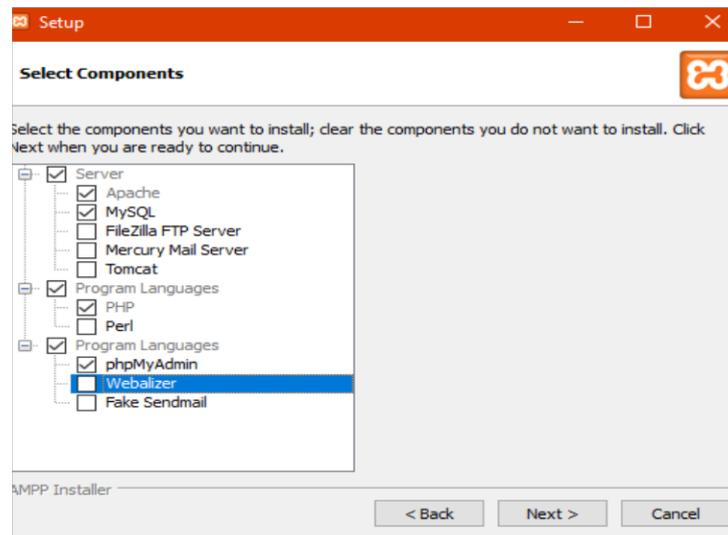
Gambar 2.1 Website Apache Friends

b. Install aplikasi yang telah diunduh kemudian pilih *next* seperti gambar dibawah



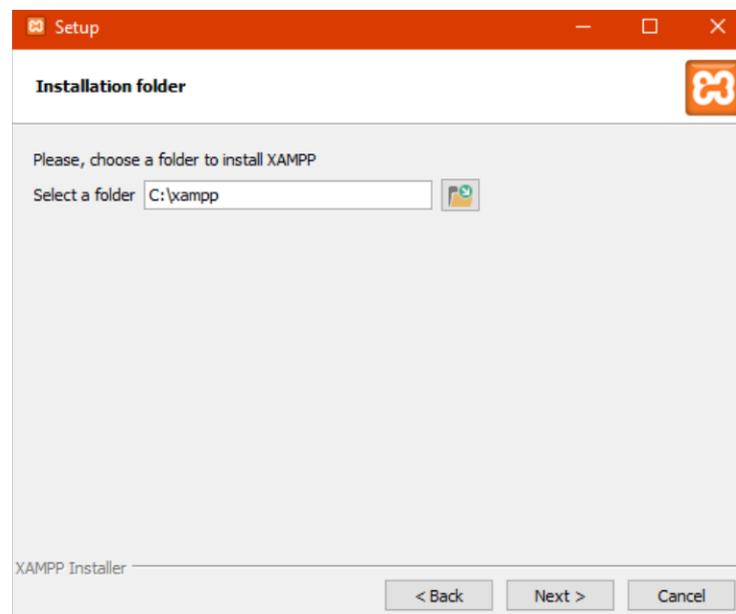
Gambar 2.2 Setup Xampp

- c. Kemudian pilih MySQL dan PhpMyAdmin lalu pilih *next*



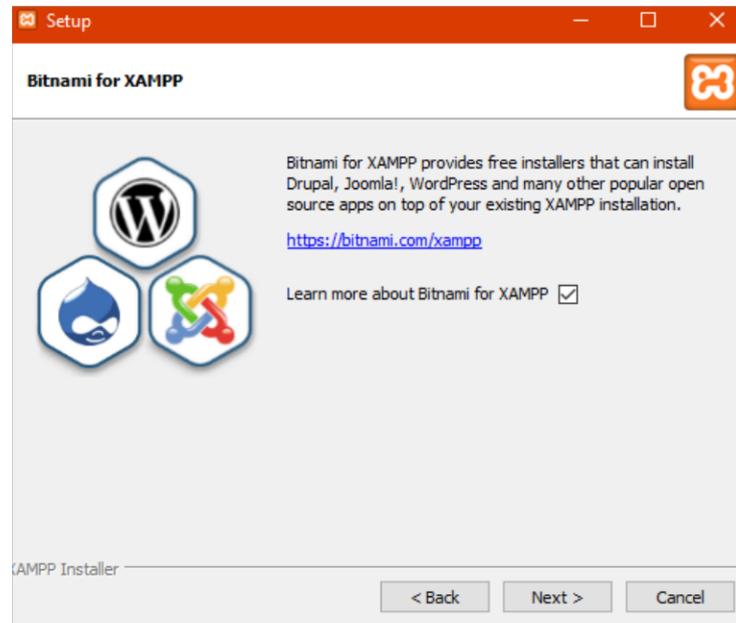
Gambar 2. 3 Select Components

- d. Berikutnya pilih *next* , apabila anda ingin merubah penyimpanan pilih *select a folder*



Gambar 2. 4 Instalation folder

- e. Selanjutnya pilih *next* , tujuannya untuk memudahkan anda untuk *install WordPress, Drupal, dan Joomla* secara otomatis



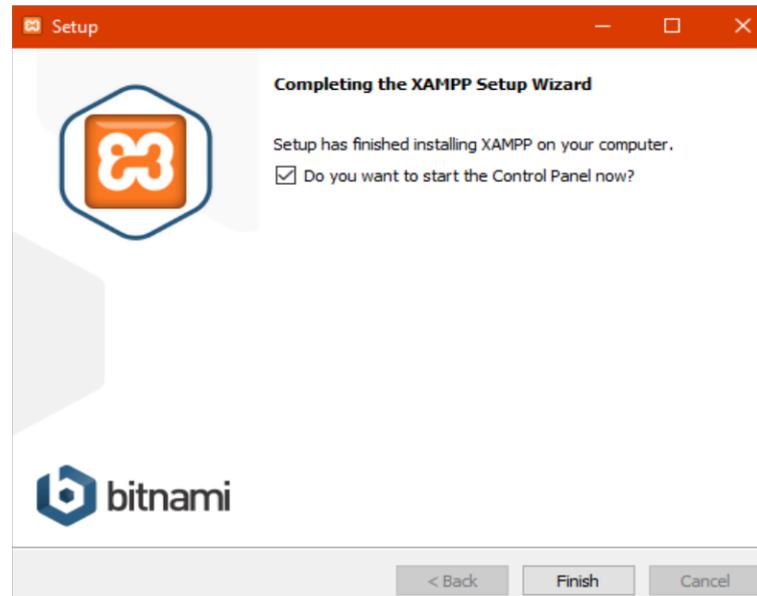
Gambar 2. 5 Bitnami for XAMPP

- f. Kemudian pilih *next* maka penginstalan akan dimulai



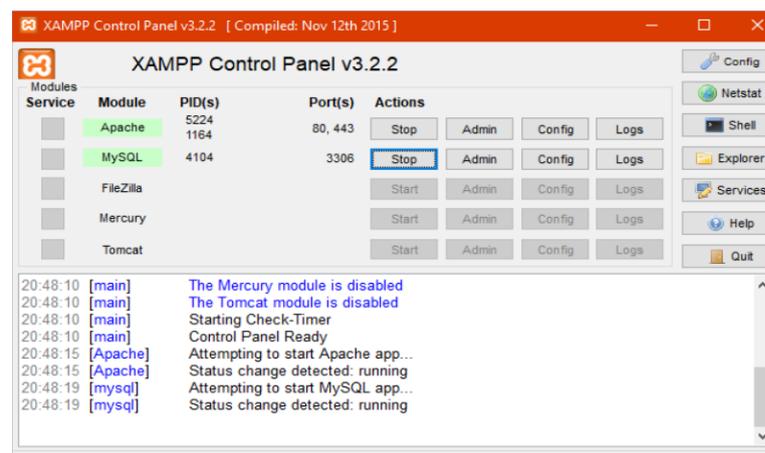
Gambar 2. 6 Setup installing Xampp

- g. Setelah berhasil diinstal pilih *Finish*



Gambar 2. 7 Completing Installation

- h. Setelah itu *Control Panel* akan terbuka otomatis, kemudian pilih *Start* dibagian Apache dan MySQL , apabila berhasil maka akan berwarna hijau seperti terlihat pada gambar dibawah ini.



Gambar 2. 8 XAMPP Control Panel

10. *UML (Unified Modelling Language)*

UML (Unified Modelling Language) adalah salah satu alat bantu yang sangat handal di dunia pengembangan sistem yang berorientasi obyek. Hal ini disebabkan karena *UML* menyediakan bahasa pemodelan visual yang memungkinkan bagi pengembang sistem untuk membuat cetak biru atas visi mereka dalam bentuk yang baku, mudah dimengerti serta dilengkapi dengan mekanisme yang efektif untuk berbagi (*sharing*) dan mengkomunikasikan rancangan mereka dengan yang lain. *UML* merupakan kesatuan dari bahasa pemodelan yang dikembangkan oleh Booch, *Object Modelling Technique (OMT)* dan *Object Oriented Software Engineering (OOSE)*. Metode Booch dari Grady Booch sangat terkenal dengan nama metode *Object Oriented Design*. (Septiani & Kuryanti, 2018).

Bagian-bagian dari *UML* dalam diagram/grafik berikut:

a. *Use Case Diagram*

Use case diagram merupakan pemodelan untuk kelakuan sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar *use case* digunakan untuk mengetahui fungsi apa saja yang ada didalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu. Syarat penamaan pada *use case* adalah nama didefinisikan sesimpel mungkin dan dapat dipahami. Ada dua hal utama pada *use case* yaitu pendefinisian apa yang disebut *actor* dan *usecase*.(Kasus et al., 2018) .

Tabel 2. 2 *Simbol Use Case Diagram*

No	Simbol	Nama	Keterangan
1		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
2		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya
3		<i>System</i>	Mensefesifikasikan paket yang menampilkan sistem secara terbatas.
4		<i>Use case</i>	Deskripsi dari urutan aksi aksi yang ditampilkan system yang menghasilkan suatu hasil yang trukur bagi suatu aktor

Sumber : Buku *Systems & Analysis Design* By Satzinger

b. *Class Diagram*

Kelas sebagai suatu set objek yang memiliki atribut dan perilaku yang sama, kelas kadang disebut kelas objek (Whitten, 2004:410). *Class* memiliki tiga area pokok yaitu :

- 1) Nama, kelas harus mempunyai sebuah nama,
- 2) Atribut, adalah kelengkapan yang melekat pada kelas. Nilai dari suatu kelas hanya bisa diproses sebatas atribut yang dimiliki,
- 3) Operasi, adalah proses yang dapat dilakukan oleh sebuah kelas, baik pada kelas itu sendiri ataupun kepada kelas lainnya. (Suendri, 2018).

11. *Database*

Database atau basis data adalah koleksi data yang bisa mencari secara menyeluruh dan secara sistematis memelihara informasi (Janner, 2007:2). Sedangkan menurut Abdul Kadir (2014:218), “basis data (*database*) adalah pengorganisasi sekumpulan data yang saling terkait sehingga memudahkan

aktivitas untuk memperoleh informasi. Basis data dimaksudkan untuk mengatasi problem pada sistem yang memakai pendekatan berbasis berkas.” Untuk mengelola *database* diperlukan perangkat lunak yang disebut *Database Management System*. *DBMS* adalah paket program (*Software*) yang dibuat agar memudahkan dan mengefesienkan input, edit dan hapus serta pengambilan informasi terhadap *database* (Robi Yanto, 2016).

Fungsi *Database* bagi orang awam database hanyalah sebuah kumpulan data, namun ternyata database memiliki fungsi antara lain sebagai berikut :

- a. *Database* berfungsi untuk mengelompokkan data agar lebih mudah dipahami.
- b. Menghindari terjadinya redudansi atau pengulangan data.
- c. Memudahkan dalam menyimpan, mengedit, dan menghapus data
- d. Mejamin integritas data.
- e. Solusi dalam proses penyimpanan data, terutama data yang besar.
- f. Menunjang kinerja aplikasi yang membutuhkan sebuah penyimpanan data.

Manfaat Database antara lain:

- a. Tidak terjadi redudansi data.
- b. Integritas data terjaga.
- c. Independensi data terjaga.
- d. Kemudahan berbagi data.
- e. Kemudahan akses data.

12. Pengenalan *HTML*

HTML bertanggung jawab mengatur tampilan halaman website, Meskipun mengandung beragam “kode” (tanda/mark/tag), tetapi *HTML* tidak bias disebut Bahasa pemrograman karena Bahasa pemrograman umumnya memiliki struktur tertentu, seperti logika if, pengulangan, *variable*, *debugger*, dan penulisan *syntax* yang benar-benar cermat. Struktur tersebut tidak ditemukan dalam *HTML*, Jika anda belajar Bahasa pemrograman *PHP*, apabila di dalam penulisan kode terdapat kekurangan satu karakter, misalnya tanda kutip, seluruh program tidak bias berjalan. Di dunia *HTML*, kesalahan penulisan kode tidak hanya akan menghasilkan pesan kesalahan. Maksimal, kesalahan kode hanya akan membuat tampilan website yang anda buat tampak tidak seperti yang diharapkan.

Sebutan lazim untuk *HTML* adalah Bahasa markup (markup language) seperti yang ada di dalam singkatan *HTML* itu sendiri. Itu artinya, *HTML* adalah Bahasa struktur untuk menandai bagian-bagian dari sebuah halaman. (Enterprise, HTML, PHP, dan MySQL untuk pemula, 2018).

Struktur dasar *HTML*, *HTML* setidaknya memiliki struktur dasar yang terdiri dari:

a. *Tag DTD* atau *DOCTYPE*

Tag paling awal dari struktur dokumen *HTML* adalah *DTD* atau biasa di sebut *DOCTYPE*. *DTD* sendiri adalah singkatan dari *Document Type Declaration* yang berfungsi untuk memberi tahu browser bahwa dokumen yang akan ditampilkan adalah dokumen yang berjenis *HTML*. *DTD*

memiliki berbagai versi cara penulisan yang umumnya tergantung dari versi *HTML* yang anda gunakan. Pada contoh diatas, cara penulisan *DTD* mengikuti versi *HTML*. Sebelum *HTML 5*, *DTD* justru ditulis dari teks yang cukup Panjang dan *njlimet*. Sebagai contoh, lihat *DTD* untuk *Xhtml 1.0*. Dalam beberapa kasus, *DTD* bias diabaikan. Jika kita tidak menulis *DTD* atau *DOCTYPE* diawal struktur *HTML*, browser akan tetap memproses dan menampilkan halaman *website* kita seperti biasa. Namun sebenarnya, browser menampilkan halaman *HTML* tersebut pada mode khusus yang disebut *Quirk Mode*. Itu artinya, browser menampilkan halaman *web* tidak sama persis seperti yang kita inginkan karena dianggap halaman tersebut kemungkinan besar merupakan halaman *web* yang *absolute* (usang).

b. Tag *HTML*

Setelah menulis tag *DTD* atau *DOCTYPE*, tag berikutnya yang harus anda buat adalah tag `<html>`. Tag ini merupakan tag pembuka dari seluruh halaman *web* yang akan dibuat. Selanjutnya, semua kode *HTML* yang membentuk desain sebuah halaman website harus berada didalam tag `<html>` ini. Tag `<html>` harus ditulis berpasangan, dimulai dengan `<html>` dan diakhiri dengan `</html>`, sementara kode-kode *HTML* lainnya ada diantara tag pembuka dan penutup tersebut.

c. Tag *HEAD*

Tag *Tang* ditulis dengan bentuk `<head>` ini berfungsi secara teknis. Bagian atau elemen yang ditulis pada tag `<head>` umumnya berbentuk keterangan teknis, seperti defenisi dan judul halaman, kode-kode *CSS*,

JavaScript, deskripsi halaman, dan kode-kode lainnya yang tidak terlihat sebagai salah satu desain didalam sebuah halaman. Judul halaman *website* ditulis menggunakan tag <title> sesudah anda menulis tag <head>. Title ini biasanya ditampilkan pada bagian paling atas jendela browser atau bagian Tab (*pada Google Chrome*).

d. *Tag BODY*

Tag <body> digunakan untuk meletakkan semua elemen yang akan terlihat di dalam halaman website pada saat halaman itu diakses oleh browser, Sama seperti dokumen biasa yang mengandung beberapa elemen, seperti paragraph, table, link, gambar dan sebagainya, kemunculan elemen itu ditulis diantara tag <body> dan </body>. Konsentrasi seorang *web designer* tertuju pada elemen di antara tag <body> dan </body>.

13. *Waterfall Model*

Metode Air terjun (*WaterFall*) Model SDLC air terjun (*waterfall*) sering juga disebut model sekuensial linier (*sequential linier*) atau alur hidup klasik (*classic life cycle*). Model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut dimulai dari analisis, desain, pengkodean, pengujian, dan tahap pendukung (*support*) (Satzinger, J.W, Jackson, R.B, dan Burd, 2011).

a. *Perencanaan*

Proses pengumpulan kebutuhan dilakukan secara intensif untuk menspesifikasika kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh *user*. Spesifikasi

kebutuhan perangkat lunak pada tahap ini perlu untuk didokumentasikan.

b. Analisis

Desain perangkat lunak adalah proses multi langkah yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur pengkodean. Tahap ini mentranslasi kebutuhan perangkat lunak dari tahap analisis kebutuhan ke representasi desain agar dapat diimplementasikan menjadi program pada tahap selanjutnya. Desain perangkat lunak yang dihasilkan pada tahap ini juga perlu didokumentasikan.

c. Desain

Desain harus ditranslasikan kedalam program perangkat lunak. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain.

d. Implementasi

Pengujian fokus pada perangkat lunak secara dari segi logik dan fungsional dan memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk menurunkan kesalahan (*error*) dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan.

e. Pendukung atau Pemeliharaan (*maintenance*)

Perangkat lunak mengalami perubahan ketika sudah dikirimkan ke user. Perubahan bisa terjadi karena adanya kesalahan yang muncul

dan tidak terdeteksi saat pengujian atau perangkat lunak harus beradaptasi dengan lingkungan baru. Tahap pendukung atau pemeliharaan dapat mengulangi proses pengembangan mulai dari analisis spesifikasi untuk perubahan perangkat lunak yang sudah ada, tapi tidak untuk membuat perangkat lunak baru.

15. Pengujian *Black Box*

Pengujian ialah proses melakukan pengujian sebuah program dengan tujuan menemukan suatu bug/kesalahan. *Test* yang dikatakan berhasil adalah bila test tersebut dapat membongkar suatu kesalahan yang awalnya tidak dideteksi (Mustaqbal et al., 2015), jenis pengujian yang akan dilakukan pada penelitian ini adalah *Black Box Testing* yang merupakan pengujian yang berfokus pada spesifikasi fungsional dari *software*. *Tester* dapat mendefinisikan kumpulan kondisi input dan melakukan pengujian pada spesifikasi fungsional program (Mustaqbal et al., 2015).

Metode *Blackbox Testing* merupakan salah satu metode yang mudah digunakan karena hanya memerlukan batas bawah dan batas atas dari data yang di harapkan, Estimasi banyaknya data uji dapat dihitung melalui banyaknya field data entri 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 (Cholifah et al., 2018).

16. *Framework Laravel*

Framework Laravel adalah sebuah kerangka kerja *open source* yang diciptakan oleh Taylor Otwell. *Laravel* merupakan framework bundle, migrasi dan artisan CLI (*Command Line Interface*) yang menawarkan seperangkat alat dan arsitektur aplikasi yang menggabungkan banyak fitur terbaik dari kerangka kerja seperti Codeigniter, Yii, ASP.NET MVC, Ruby on Rails, Sinatra dan lain-lain. *Laravel* memiliki seperangkat sangat kaya fitur yang akan meningkatkan kecepatan pengembangan web (Handika & Purbasari, 2018).

Fitur pada *Framework Laravel* yang membedakan antara *Framework php* ini dengan *Framework* lainnya sebagai berikut:

- a. *Dependency Management*, Adalah sebuah fitur yang berguna untuk memahami fungsi wadah layanan (IoC) memungkinkan objek baru dihasilkan dengan pembalikan controller dan merupakan bagian inti untuk mempelajari aplikasi Web modern,
- b. *Modularity*, adalah sejauh mana sebuah komponen aplikasi Web dapat dipisahkan dan digabungkan kembali. *Modularity* dapat membantu kamu untuk mempermudah proses update, Bukan hanya itu, Dalam membangun dan mengembangkan Website, fitur ini memungkinkan kamu untuk menyempurnakan dan meningkatkan fungsionalitas dari web tersebut,
- c. *Authentication*, adalah bagian integral dari aplikasi web modern, *Laravel* menyediakan otentikasi di luar kotak, dengan menjalankan

perintah sederhana. Kamu juga dapat membuat sebuah sistem yang otentikasinya berfungsi secara penuh dan proses konfigurasi otentikasi sudah berjalan secara otomatis,

- d. *Caching*, adalah sebuah teknik untuk menyimpan data di lokasi penyimpanan sementara dan dapat diambil dengan cepat saat dibutuhkan. Dalam *laravel*, hampir semua data *caching* berasal dari tampilan ke rute, Sehingga dapat mengurangi waktu pemrosesan dan meningkatkan kinerja,
- e. *Routing*, *Routing Laravel* dapat digunakan untuk membuat aplikasi yang tenang dengan mudah. Dalam *framework* ini semua *request* dipetakan dengan bantuan rute. Kamu juga dapat mengelompokkan rute, menamainya, menerapkan filter pada mereka dan mengikat data model Anda kepadanya,
- f. *Restful Controller*, Adalah sebuah fitur yang berfungsi memisahkan logika dalam melayani HTTP GET and POST. Kamu juga dapat membuat pengontrol sumber daya yang dapat digunakan untuk membuat CRUD dengan mudah,
- g. *Testing and Debugging*, *Laravel* hadir dengan menggunakan PHP Unit di luar kotak yang berfungsi untuk menguji aplikasi kamu. *Framework* ini dibangun dengan pengujian dalam pikiran sehingga mendukung pengujian dan debugging terlalu banyak,

- h. *Template Engine, Blade* adalah template engine *Laravel, Blade* berfungsi untuk menyediakan sejumlah fungsi pembantu untuk memformat data kamu dalam bentuk tampilan,
- i. *Configuration Management Features*, Dalam *laravel* semua file konfigurasi kamu disimpan di dalam direktori *config*, kamu dapat mengubah setiap konfigurasi yang tersedia,
- j. *Eloquent ORM*, *Laravel* berbasis pada *Eloquent ORM* yang menyediakan dukungan untuk hampir semua mesin basis data. Fitur ini juga berfungsi untuk menjalankan berbagai operasi database di dalam website dan mendukung berbagai sistem database seperti *MySQL* dan *SQLite*.

Kelebihan *Laravel* Dibanding *Framework* Lainnya:

- a. Templat Layout yang Lebih Ringan

Laravel menyediakan sebuah templat layout yang sangat ringan, Terdapat juga sebuah fitur yang memungkinkan untuk menambahkan *template* tambahan yang ringan. Bagi para *developer* dapat melakukan pengembangan secara *powerfull*. Jika tadi sudah disinggung tentang template engine, dalam pembuat *template*, *Template engine* tersebut sangat berfungsi,

- b. Dukungan *Framework MVC Laravel*

Struktur *MVC Laravel* dapat dengan mudah mengelola basis data seperti input data ke basis data, melakukan pembaruan data dan pengelolaan data lainnya. Dengan *MVC* ini maka kamu dapat mengatur

tampilan ke pengguna dan dapat mengontrol antara pengolahan basis data dan mengatur tampilan ke pengguna,

c. *Laravel Mempunyai Banyak Library Object Oriented*

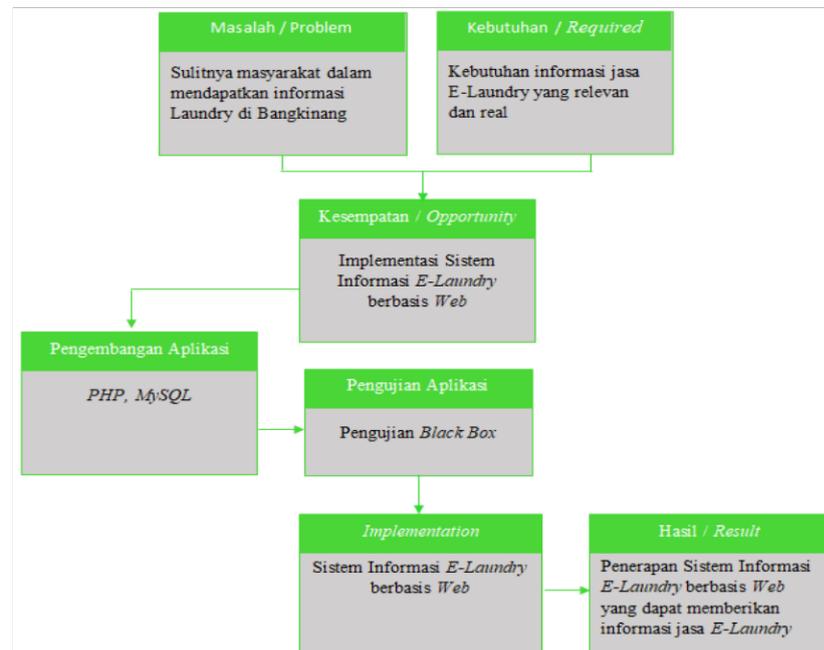
Keunggulan laravel dibanding *framework* lainnya yaitu terdapat pada banyaknya *library object oriented* yang tersedia, dan *library* tersebut tidak terdapat pada *library framework* lainnya. Sehingga *framework* ini mempunyai nilai plus pada poin ini dibanding *framework php* lainnya. Salah satu *library* yang sangat berguna dan paling penting pada laravel yaitu *library authentication*. *Library* ini menyediakan fitur *Bcrypt hashing*, Fitur tersebut merupakan salah satu fitur tercanggih di bidang otentifikasi saat ini,

d. *Modul dalam Laravel Bersifat Individu dan Independen*

Aplikasi *laravel*, Terdapat lebih dari 20 modul *library* yang digunakan dan masing-masing modul bersifat individu. Selain bersifat individu modul tersebut juga bersifat independen dalam mengadopsi prinsip PHP modern. Dengan begitu maka modul-modul tersebut akan membuat aplikasi web menjadi lebih informatif, modular, dan responsif.

C. **Kerangka Konsep**

Kerangka konsep didalam pembuatan Implementasi Sistem Informasi E-Laundry berbasis web ini mengacu pada perumusan masalah yang sudah dirumuskan dan mengacu pada materi Sistem Informasi. Berikut merupakan kerangka konsep:



Gambar 2. 9 Kerangka Konsep

Kerangka konsep diatas dideskripsikan sebagai berikut :

1. Masalah / *Problem*

Pada penelitian ini yang jadi masalah adalah sulitnya masyarakat dalam mendapatkan Informasi tentang rekomendasi usaha Laundry di Bangkinang

2. Kebutuhan / *Required*

Kebutuhan Informasi jasa *E-Laundry* yang relevan dan real agar memudahkan masyarakat untuk mendapatkan rekomendasi jasa Laundry yang terbaik

3. Kesempatan / *Opportunity*

Implementasi Sistem Informasi *E-Laundry* Berbasis *Web* dapat menjadi salah satu solusi yang tepat untuk mencari rekomendasi jasa *Laundry* terbaik yang ada di Bangkinang

4. Pengembangan aplikasi / *Application Development*

Aplikasi ini dibangun dengan menggunakan Bahasa pemrograman *PHP* dengan *framework Laravel*

5. Pengujian / Testing

Tahap pengujian sistem dalam penelitian ini menggunakan dua kali pengujian yaitu pengujian *black box testing* dan uat, dimana pengujian *black box* berguna untuk menguji semua fitur sudah berjalan dengan seharusnya sebelum sistem di gunakan *user*, dan pengujian uat (*user acceptance test*) bertujuan untuk menghasilkan dokumen yang dapat dijadikan bukti bahwa sistem/aplikasi yang dibuat dapat diterima oleh pengguna.

6. *Implementation*

Tahap *implementasi* akan dilakukan setelah pengujian Sistem dilakukan, sehingga sistem sudah *fix* dari *bug* dan dapat di implementasikan kepada penggunanya

7. Hasil / *Result*

Hasil yang di harapkan dari Sistem yang telah dibuat yaitu *Implementasi* Sistem Informasi *E-Laundry* Bebasis *Web* yang dapat memberikan rekomendasi jasa *Laundry* kepada masyarakat .

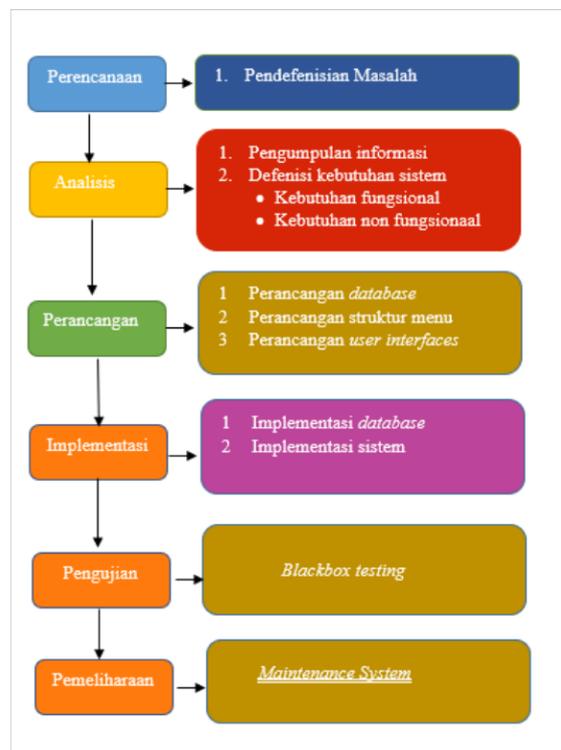
BAB III

METODOLOGI PENELITIAN

Metode Penelitian berasal dari dua suku kata yaitu metode berasal dari Bahasa Yunani *methodos* yang berarti cara atau jalan yang ditempuh, dan penelitian berasal dari kata *research* “re” adalah kembali “*search*” mencari. Mencari kembali yang dimaksud adalah secara terus-menerus melakukan penelitian melalui proses pengumpulan informasi dengan tujuan meningkatkan, memodifikasi atau mengembangkan sebuah penyelidikan atau kelompok penyelidikan. Metode menurut Sugiyama (2008) adalah “suatu tingkat representasi tinggi dari jejaring teori (*theoretical network*) yang biasanya didesain dengan menggunakan simbol atau analogis secara fisik.” Sedangkan Metode penelitian menurut Sugiyono (2007) adalah “cara- cara ilmiah untuk mendapatkan data yang valid, dengan tujuan dapat ditemukan, dikembangkan dan dibuktikan, suatu pengetahuan tertentu sehingga pada gilirannya dapat digunakan untuk memahami, memecahkan, dan mengantisipasi masalah (Nana & Elin, 2018) .

Model *Waterfall* merupakan salah satu model pengembangan perangkat lunak yang ada di dalam model *SDLC* (*Sequential Development LifeCycle*). Menurut Sukamto dan Shalahuddin (2013:26) mengemukakan bahwa “ *SDLC* atau *Software Development Life Cycle* atau sering disebut juga *System Development Life Cycle* adalah proses mengem-bangkan atau mengubah suatu sistem perangkat lunak dengan menggunakan model-model dan metodologi yang digunakan orang untuk mengembangkan sistem-sistem perangkat lunak sebelumnya, berdasarkan *best practice* atau cara-cara yang sudah teruji

baik.”Sedangkan Sukamto dan Shalahuddin (2013:28) di jelaskan bahwa model waterfall sering juga disebut model sekuensi linear atau alur hidup klasik. Pengembangan sistem dikerjakan secara terurut mulai dari analisis, desain, pengkodean, pengujian dan tahap pendukung. Untuk lebih jelasnya metodologi penelitian ini dapat dilihat pada gambar dibawah ini .



Gambar 3. 1 Metodologi Penelitian

A. Perencanaan

Tahap perencanaan di Skripsi ini dibuat dengan dua aktifitas yaitu aktifitas pendefinisian masalah dan pembuatan jadwal. Pendefinisian masalah tentang penjelasan latar belakang dan perumuskan tujuan yang akan dicapai pada penelitian ini, lalu diberikan solusi dari permasalahan tersebut. Adapaun rumusan masalah yang ada pada penelitian ini adalah bagaimana membangun dan

menyebarkan sebuah sistem informasi *E-Laundry* berbasis web. Teknik yang digunakan didalam pendefinisian masalah disini adalah Metode *Kipling*, dimana dalam perumusan masalah dilakukan pertanyaan menggunakan 5W + 1H untuk membantu memicu pemikiran dan mengatasi masalah.

B. Analisis

Tujuan utama dari kegiatan analisis adalah untuk memahami dan mendokumentasikan bisnis kebutuhan dan persyaratan pemrosesan sistem baru. Analisis pada dasarnya adalah penemuan proses.

1. Pengumpulan Informasi

Tahap analisis pengumpulan informasi yang dilakukan pada penelitian ini dibagi menjadi dua yaitu kualitatif dan kuantitatif :

a. Data Kualitatif

Pengertian data kualitatif menurut Sugiyono (2015) adalah data yang berbentuk kata, skema, dan gambar. Data kualitatif penelitian ini berupa nama dan alamat obyek penelitian,

b. Data Kuantitatif

Pengertian data kuantitatif menurut Sugiyono (2015) adalah data yang berbentuk angka atau data kualitatif yang diangkakan. Data kuantitatif penelitian ini berupa kuesioner ke masyarakat Bangkinang yang bersedia menjadi responden dan mengisi kuesioner. Berikut merupakan hasil dari kuesioner yang dibagikan kepada masyarakat :

Keterangan Pilihan Jawaban:

- SS = Sangat Setuju
 S = Setuju
 CS = Cukup Setuju
 TP = Tidak Pernah
 A = Ada
 TD = Tidak Ada

Tabel 3. 1 Pertanyaan kuesioner

NO	Pertanyaan	Pilihan Jawaban					
		SS	S	CS	TP	A	TD
1	Apakah anda sering mencuci pakaian di suatu Laundry ?						
2	Apakah dengan adanya Jasa Laundry dapat membantu meringankan pekerjaan anda dalam urusan mencuci pakaian ?						
3	Apakah anda sering bingung dalam memilih Jasa Laundry ?						
4	Apakah di sekitar tempat tinggal anda ada sebuah Jasa Laundry ?						
5	Apakah anda sering menggunakan sebuah <i>Website</i> ?						
6	Apakah anda tertarik apabila ada sebuah layanan <i>Website</i> untuk Jasa laundry ?						
7	Apakah dengan adanya layanan <i>Website</i> Laundry dapat membantu anda untuk mencari Jasa Laundry terbaik ?						
8	Jika Aplikasi Laundry ini dibangun apakah sesuai dengan SDM kabupaten Kampar ?						
9	Jika Aplikasi Laundry ini dibangun apakah sesuai dengan SDM kabupaten Kampar ?						
10	Apakah anda setuju apabila ada jasa antar jemput disebuah <i>Website</i> Laundry ?						
11	Apakah anda setuju apabila terdapat metode pembayaran <i>online</i> disebuah <i>Website</i> Laundry ?						
12	Apakah menurut anda penting jika terdapat fitur notifikasi di <i>Website</i> Laundry apabila pakaian anda telah selesai dicuci ?						

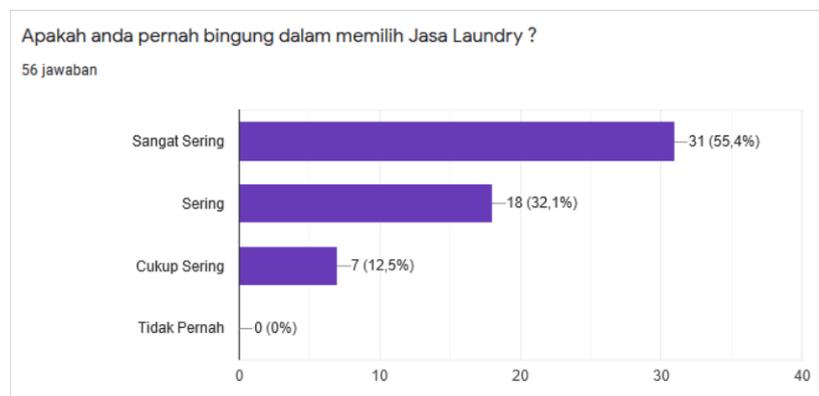
Hasil jawaban responden dari kuesioner ini juga dapat dilihat dalam bentuk diagram :



Gambar 3. 2 Diagram Kuesioner Pertanyaan ke 1



Gambar 3. 3 Diagram Kuesioner Pertanyaan ke 2



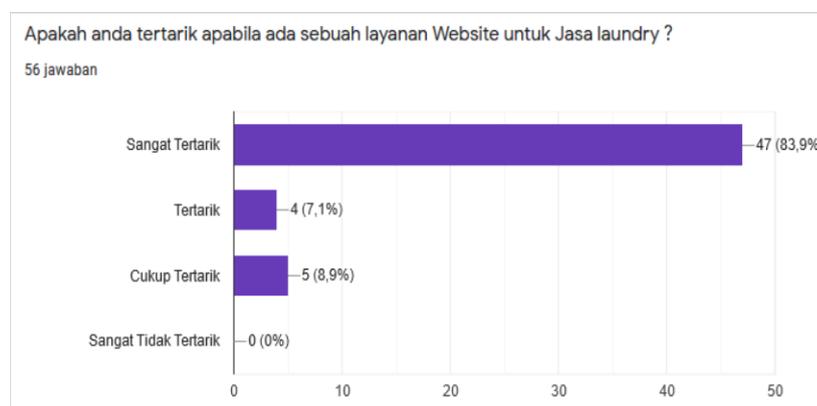
Gambar 3. 4 Diagram Kuesioner Pertanyaan ke 3



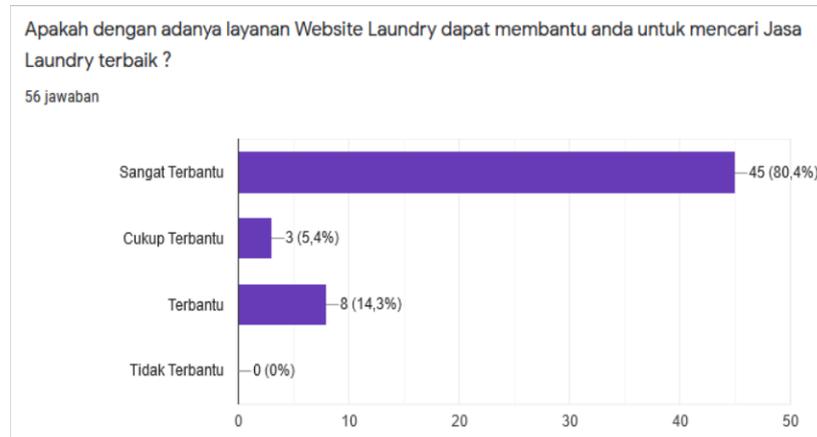
Gambar 3. 5 Diagram Kuesioner Pertanyaan ke 4



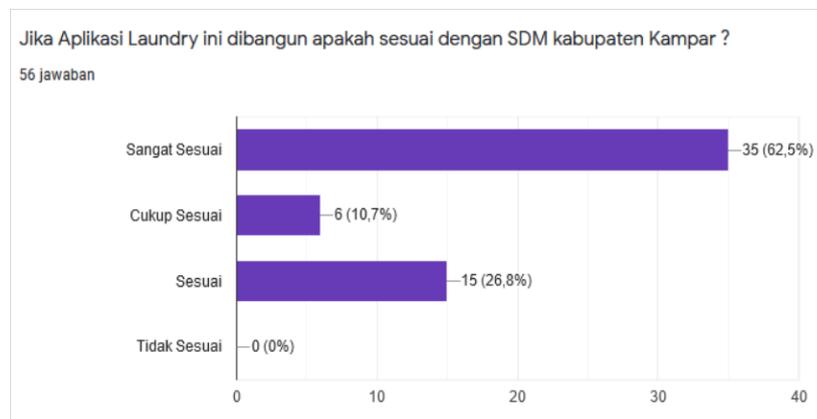
Gambar 3. 6 Diagram Kuesioner Pertanyaan ke 5



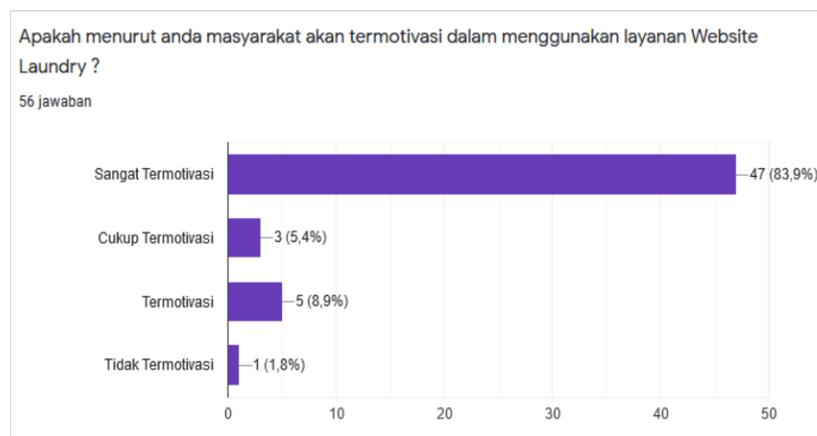
Gambar 3. 7 Diagram Kuesioner Pertanyaan ke 6



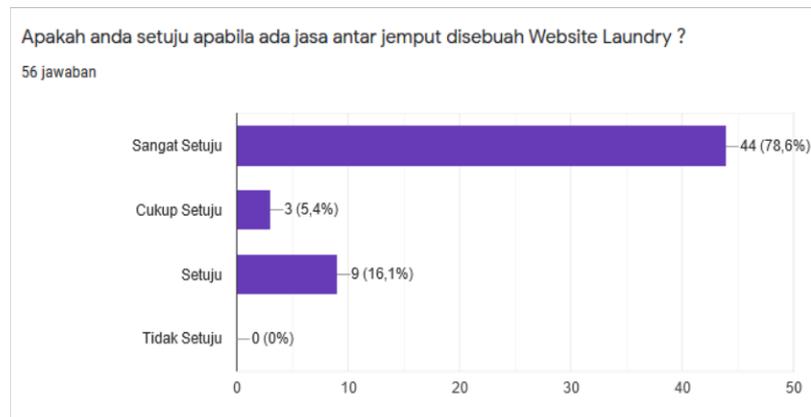
Gambar 3. 8 Diagram Kuesioner Pertanyaan ke 7



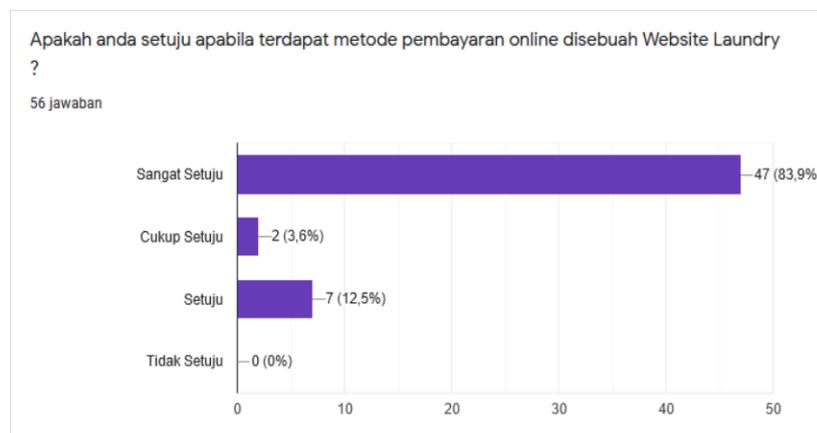
Gambar 3. 9 Diagram Kuesioner Pertanyaan ke 8



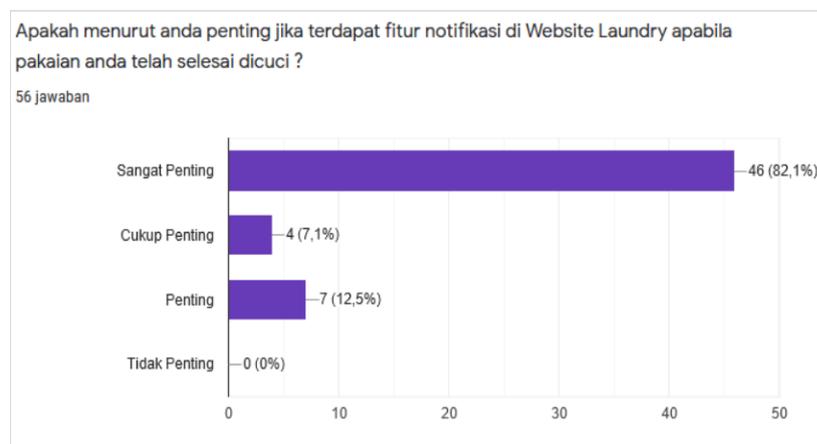
Gambar 3. 10 Diagram Kuesioner Pertanyaan ke 9



Gambar 3. 11 Diagram Kuesioner Pertanyaan ke 10



Gambar 3. 12 Diagram Kuesioner Pertanyaan ke 11



Gambar 3. 13 Diagram Kuesioner Pertanyaan ke 12

2. Definisi Kebutuhan Sistem

Pendefinisian kebutuhan sistem diolah dengan cara mendefinisikan kebutuhan fungsional dan non fungsional sistem. Berikut masing-masing penjelasannya.

a. Kebutuhan Fungsional Sistem

Kebutuhan fungsional adalah kebutuhan pada sistem yang merupakan layanan dalam sistem yang harus disediakan, serta gambaran proses dari reaksi sistem terhadap masukan sistem dan yang akan dikerjakan oleh sistem. Kebutuhan fungsional didapatkan melalui hasil wawancara. Kebutuhan fungsional dibuat berdasarkan diagram UML yaitu *use case*, *use case* skenario, *activity* diagram, dan *class* diagram. *Activity* diagram dibuat berdasarkan diagram *use case* yang telah dibuat. *Tools* yang digunakan dalam membuat kebutuhan fungsional sistem ini adalah *Microsoft Visio 2019*.

b. Kebutuhan Nonfungsional Sistem

Kebutuhan non fungsional sistem dibagi menjadi kebutuhan fungsional *software*, *brainware*, *hardware*, *netware* dan *dataware*.

C. Perancangan

Perancangan pada tugas akhir ini dibagi menjadi 3 tahap perancangan yaitu, perancangan *database*, perancangan struktur menu dan perancangan *user interface*.

1. Perancangan *Database*

Perancangan internal ini adalah termasuk sebagai bagian penting karena didalam ini akan menyimpan data-data informasi tentang *Laundry*. Perancangan *database* menggunakan *Class Diagram* yang dibangun menggunakan *Microsoft Visio 2019*.

2. Perancangan Struktur Menu

Perancangan stuktur menu menggunakan *tools Microsoft visio 2019* diharapkan agar perancangan stuktur *menu* dapat membantu sebagai gambaran mengenai skema perancangan program.

3. Perancangan *User Interface*

Perancangan sistem harus layak pakai oleh pengguna awam maupun ahli, pada kegiatan analisis perancangan *user interface* sudah dilakukan, namun pada tahap desain adalah menggabungkan seluruh potongan-potongan *prototype* sehingga didapat gambaran sistem nyata. Semua elemen digabungkan untuk menghasilkan antarmuka pengguna terintegrasi yang terdiri dari *html, css, bootstrap, java script* Perancangan *user interface* menggunakan *tools Microsoft studio 2010*

D. Implementasi

Ada 2 tahap pengimplementasian, yang pertama adalah implementasi *database*, implementasi *system*.

1. Implementasi *Database*

Implementasi *Database* dirancang melalui rancangan yang telah ditentukan sebelumnya, pengimplementasian *database* dilakukan pada sebuah *server local* yaitu pada *phpmyadmin* (MySQL).

2. Implementasi *System*

Implementasi *system* dengan bahasa pemrograman PHP dilakukan setelah melakukan sebuah perancangan. Pada fase ini *system* akan dibangun dengan kode program, dengan menggunakan bahasa PHP dengan *framework laravel* dan *tools visual studio code* sebagai *text editor* dan *Xampp* sebagai server lokal, untuk membangun *system*.

E. Pengujian

Penafsiran dan penarikan kesimpulan dilakukan berdasarkan tiap langkah dalam pembuatan program, simpulan akhir ditentukan dengan berhasil tidaknya program dapat dijalankan sesuai dengan rancangan. Indikator keberhasilan didasarkan pada penerapan rancangan basis data , yaitu dengan membandingkan sebelum memakai rancangan basis data (manual) dengan sesudah memakai basis data yang telah terkomputerisasi. Setelah kegiatan pengkodean selesai, sistem akan diuji untuk memastikan sistem dapat bekerja sesuai dengan tujuan yang diharapkan. Pengujian sistem yaitu menggunakan metode *Black Box Testing* dan *User Acceptance Test (UAT)*. *Blackbox Testing* dalam Pengujian akan menguji sistem yang berfokus pada spesifikasi fungsional dari perangkat lunak.

F. Pemeliharaan

Perangkat lunak mengalami perubahan ketika sudah dikirimkan ke *user*. Perubahan bisa terjadi karena adanya kesalahan yang muncul dan tidak terdeteksi saat pengujian atau perangkat lunak harus beradaptasi dengan lingkungan baru. Tahap pendukung atau pemeliharaan dapat mengulangi proses pengembangan

mulai dari analisis spesifikasi untuk perubahan perangkat lunak yang sudah ada, tapi tidak untuk membuat perangkat lunak baru.

BAB IV

HASIL DAN PEMBAHASAN

A. Analisa Sistem Berjalan

Analisa sistem berjalan merupakan langkah awal yang harus dilaksanakan sebelum memberikan sistem usulan, berdasarkan analisa ini nantinya akan dijadikan alasan kenapa sistem usulan dibutuhkan. Analisa sistem yang sudah ada dapat berupa identifikasi permasalahan, dari hasil identifikasi tersebut telah ditemukan beberapa masalah yang dapat diselesaikan. Seperti sulitnya masyarakat mencari jasa Laundry yang terbaik dan pemilik Laundry yang kesulitan mempromosikan jasa Laundry

B. Analisa Sistem Usulan

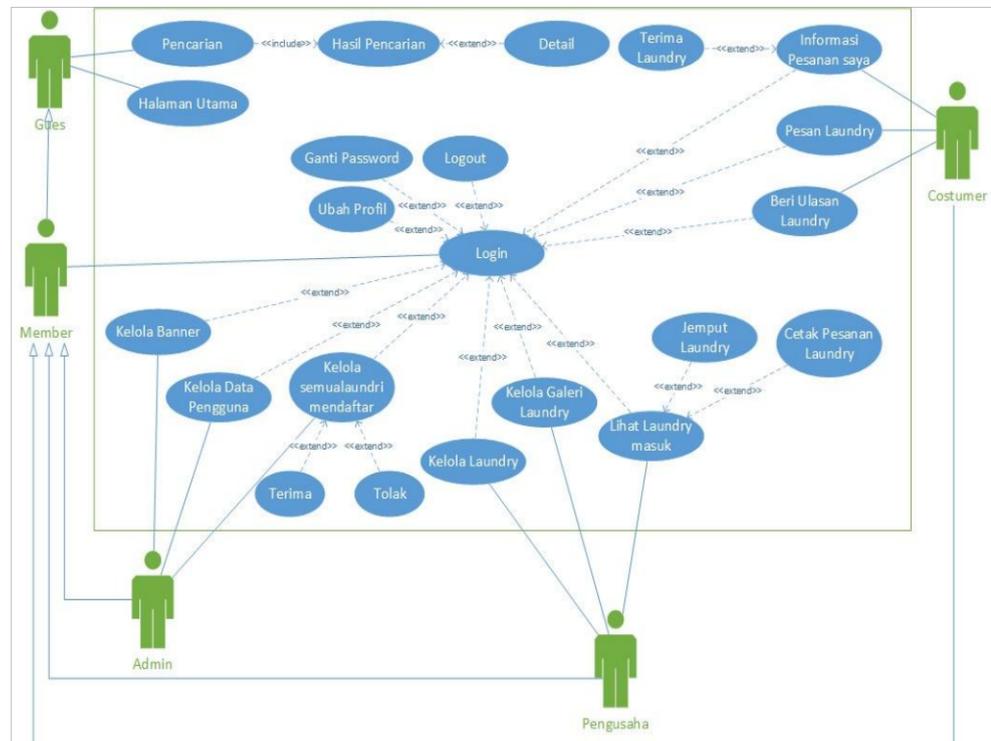
Perancangan sistem usulan yang akan dibangun berguna untuk mempermudah pemilik *Laundry* untuk mempromosikan jasa *laundry* dan membantu masyarakat dalam mencari jasa *Laundry* sehingga masyarakat tidak kesulitan dalam mencari informasi jasa *Laundry*, sehingga pencarian informasi jasa *Laundry* lebih efektif dan efisien. Sistem yang akan dibangun ini menggunakan bahasa pemrograman PHP dengan *framework Laravel*

C. Perencanaan Sistem Dengan UML (*Unified Modelling Language*)

Perancangan sistem akan digunakan *unified modelling language* (UML), digunakan untuk memvisualisasikan dan menggambarkan sebuah sistem informasi, diagram UML yang akan digunakan ialah *use case diagram* sebagai *tools* untuk memperlihatkan hubungan pengguna dengan sistem, dan *class diagram* sebagai *tools* untuk menggambarkan struktur sebuah sistem.

1. Use case Diagram

Perancangan *use case diagram*, dapat dilihat pada gambar berikut:



Gambar 4. 1 Use case Diagram

2. Deskripsi Use case Diagram

Pada tahap ini menjelaskan deskripsi *use case* antara *actor* dan sistem.

- a) Penjelasan deskripsi *use case* pengunjung pada sistem informasi *E-Laundry* sebagai berikut:

Tabel 4. 1 Deskripsi Use case Actor dan Sistem

No	Use case	Deskripsi
1	Halaman utama	Menampilkan halaman utama untuk seluruh pengunjung
2	Pencarian	Dapat melakukan pencarian Laundry
3	Hasil pencarian	pengunjung dapat melihat hasil pencarian Laundry
4	Detail Laundry	pengunjung dapat melihat detail Laundry yang dicari

- b) Penjelasan deskripsi *Use case* member pada sistem informasi *E-Laundry* sebagai berikut:

Tabel 4. 2 Deskripsi Use case Member

No	Use case	Deskripsi
1	<i>Login</i>	pengunjung dapat melakukan <i>login</i> kedalam sistem.
2	<i>Setting profie</i>	pengguna dapat melakukan <i>setting profile</i> .
3	<i>Ganti password</i>	pengguna dapat melakukan perubahan <i>password</i> .
4	<i>Logoff</i>	Pengunjung dapat melakukan <i>logoff</i>

- c) Penjelasan deskripsi *Use case* admin pada sistem informasi *E-Laundry* sebagai berikut:

Tabel 4. 3 Deskripsi *Use case* Admin

No	Use case	Deskripsi
1	Kelola <i>banner</i>	Admin dapat mengelola <i>banner</i>
2	Kelola data pengguna	Admin dapat mengelola data pengguna .
3	Kelola semua Laundry mendaftar	Admin dapat mengelola semua Laundry yang mendaftar.
4	Terima Laundry	Admin dapat menerima Laundry yang mendaftar.
5	Tolak Laundry	Admin dapat menolak Laundry yang mendaftar.

- d) Penjelasan deskripsi *Use case* pemilik Laundry pada sistem informasi *E-Laundry* sebagai berikut:

Tabel 4. 4 Deskripsi *Use case* Pemilik

No	Use case	Deskripsi
1	Kelola Laundry	Pemilik Laundry dapat mengelola Laundry
2	Kelola <i>gallery</i> Laundry	Pemilik Laundry dapat mengelola <i>gallery</i> Laundry
3	Lihat Laundry masuk	Pemilik Laundry dapat melihat <i>Laundry</i> masuk
4	Jemput Laundry	Pemilik Laundry dapat menjemput pakaian customer
5	Cetak pesanan laundry	Pemilik Laundry dapat mencetak Riwayat pesanan customer

- e) Penjelasan deskripsi *Use case* customer pada sistem informasi *E-Laundry* sebagai berikut:

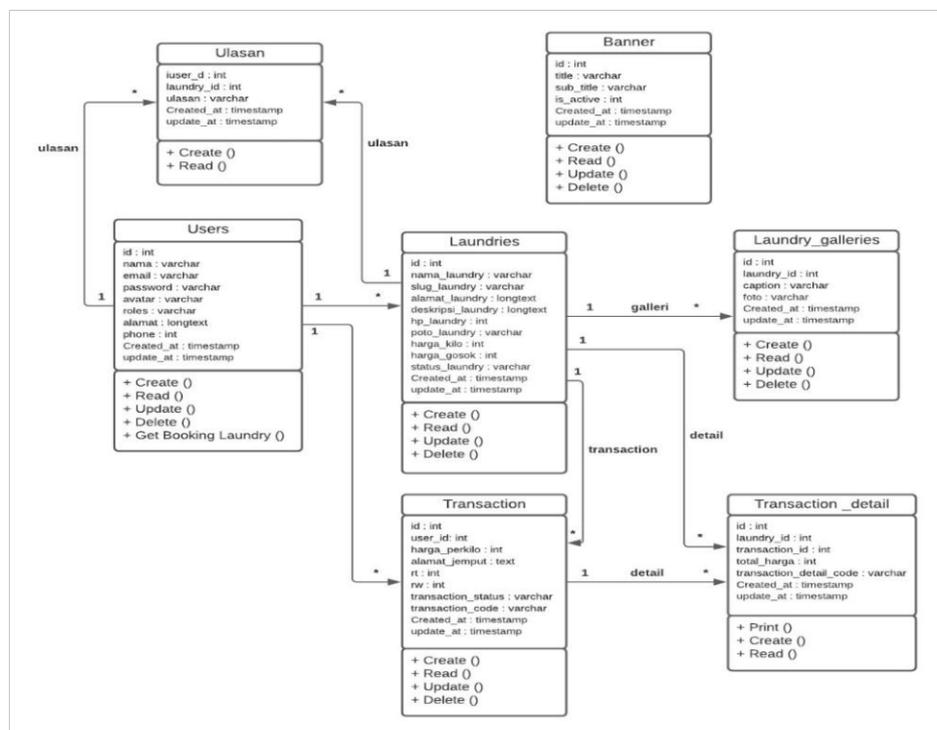
Tabel 4. 5 Deskripsi *Use case* Customer

No	Use case	Deskripsi
1	Pesan Laundry	Customer dapat memesan Laundry
2	Beri ulasan Laundry	Customer dapat memberi ulasan Laundry

No	Use case	Deskripsi
3	Informasi pesanan saya	Customer dapat melihat informasi pesanan
4	Terima Laundry	Customer dapat menerima pesanan Laundry

3. Class Diagram

Class diagram merupakan penjelasan lengkap dari beberapa *class* yang ditangani oleh sistem. Dimana tiap-tiap *class* dipasangkan dengan beberapa *attribute* dan *method* yang diperlukan. Berikut adalah skema dari model *class diagram* sistem informasi *E-Laundry*



Gambar 4. 2 Class Diagram

4. Deskripsi Class Diagram

	Nama kelas	Atribut	Method
1	<i>Users</i>	<i>id : int</i> <i>nama : string email : varchar</i> <i>email : varchar</i> <i>password : varchar</i> <i>avatar: varcahr</i> <i>roles : varchar</i> <i>alamat : longtext</i> <i>Phone : integer</i> <i>created_at : timestamp</i> <i>update_at : timestamp</i>	+ <i>Login ()</i> + <i>Register ()</i> + <i>Create ()</i> + <i>Update ()</i> + <i>Delete ()</i> + <i>read ()</i> + <i>getBooking laundry()</i> + <i>changepassword ()</i>
2	<i>Laundries</i>	<i>id : int</i> <i>nama_laundry : varchar</i> <i>Slug_Laundry : varchar</i> <i>alamat_Laundry : longtext</i> <i>deskripsi_Laundry : longtext</i> <i>hp_laundry : int</i> <i>poto_laundry :varchar</i> <i>harga_kilo : int</i> <i>harga_gosok : int</i> <i>status_laundry : varchar</i> <i>create_at : timestamp</i> <i>update_at : timestamp</i>	+ <i>Create ()</i> + <i>Read ()</i> + <i>Update ()</i> + <i>Delete ()</i>
3	<i>Laundry_Galleries</i>	<i>id : int</i> <i>laundry_id : int</i> <i>email : varchar</i> <i>caption : varchar</i> <i>foto : varcahr</i> <i>created_at : timestamp</i> <i>update_at : timestamp</i> <i>Phone : integer</i> <i>created_at : timestamp</i> <i>update_at : timestamp</i>	+ <i>Create ()</i> + <i>read ()</i> + <i>Update ()</i> + <i>Delete ()</i>
4	<i>Transaction</i>	<i>id : int</i> <i>user_id : int</i> <i>harga-perkilo : int</i> <i>alamat_jemput : text</i> <i>rt : int</i> <i>rw : int</i> <i>transaction_status : varchar</i> <i>transaction_code : varchar</i> <i>create_at : timestamp</i> <i>update_at : timestamp</i>	+ <i>Create ()</i> + <i>Read ()</i> + <i>Update ()</i> + <i>Delete ()</i>

	Nama kelas	Atribut	Method
5	<i>Transaction_detail</i>	<i>Id : int</i> <i>Laundry_id : int</i> <i>transaction_id : id</i> <i>total_harga : int</i> <i>transaction_detail_code : varchar</i> <i>create_at : timestamp</i> <i>update_at : timestamp</i>	<i>+ Print ()</i> <i>+ Create ()</i> <i>+ Read ()</i>
6	<i>Banner</i>	<i>id : int</i> <i>title : varchar</i> <i>sub_title : varchar</i> <i>is_active : int</i> <i>create_at : timestamp</i> <i>update_at : timestamp</i>	<i>+ Create ()</i> <i>+ Read ()</i> <i>+ Update ()</i> <i>+ Delete ()</i>
7	<i>Ulasan</i>	<i>User_id : int</i> <i>Laundry_id : int</i> <i>Ulasan : varchar</i> <i>create_at : timestamp</i> <i>update_at : timestamp</i>	<i>+ Create ()</i> <i>+ Read ()</i>

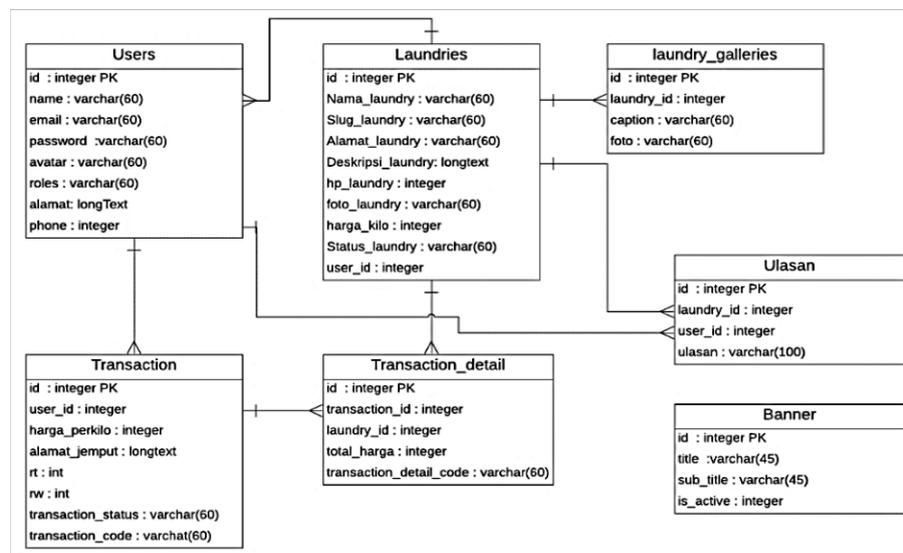
Tabel 4. 6 Deskripsi Class Diagram

D. Perencanaan Database

Sistem yang dibangun terdapat beberapa bisnis proses yang ingin diselesaikan adalah rancang bangun sistem informasi *E-Laundry* dengan implementasi berbasis *Web*.

1. Merancang Relasi Antar Tabel

Tahapan ini tampungan data dan relasi tabel pada sistem informasi *E-Laundry* berbasis *Website* yang dibuat dengan menyesuaikan antara bisnis proses dan sistem yang akan dibangun, maka dapat dibentuk suatu *Database* yang nantinya akan digunakan untuk menyimpan keseluruhan proses yang diperlukan seperti



Gambar 4. 3 Perancangan *Database*

2. Database dan dan struktur table

Database yang digunakan pada sistem ini adalah *Laundry*. *Laundry* memiliki tabel yang saling berelasi untuk mendukung efisiensi pada setiap kata hubungannya daftar tabel yang ada pada sistem

a) Tabel *Users*

Tabel *users* digunakan untuk menampung semua data-data pengguna system

Tabel 4. 7 Tabel *Users*

No	Nama kolom	Tipe data	Keterangan
1	<i>Id</i>	<i>Integer</i>	<i>Primary key</i>
2	<i>Name</i>	<i>Varchar(60)</i>	Nama pengguna
3	<i>Email</i>	<i>Varchar(60)</i>	<i>Email</i> pengguna
4	<i>Password</i>	<i>Varchar(60)</i>	<i>Password</i> pengguna
5	<i>Avatar</i>	<i>Varchar(60)</i>	Foto pengguna
6	<i>Roles</i>	<i>Varchar(60)</i>	<i>Role</i> pengguna
7	Alamat	<i>Longtext</i>	Alamat Pengguna
8	<i>Phone</i>	<i>Integer</i>	Nomor telepon pengguna

b) Tabel *Laundries*

Tabel *Laundries* digunakan untuk menampung semua data-data

Laundry

Tabel 4. 8 Tabel *Laundries*

No	Nama kolom	Tipe data	Keterangan
1	<i>Id</i>	<i>Integer</i>	<i>Primary key</i>
2	Nama_Laundry	<i>Varchar(60)</i>	Nama Laundry yang didaftar
3	Slug_Laundry	<i>Varchar(60)</i>	Digunakan untuk mesin pencarian Laundry
4	Alamat_Laundry	<i>Longtext</i>	Alamat dari Laundry
5	<i>Deskripsi_Laundry</i>	<i>Longtext</i>	Digunakan untuk mendeskripsikan Laundry
6	Hp_Laundry	<i>Integer</i>	Digunakan untuk menambah nomor telepon
7	Foto_Laundry	<i>Varchar(600)</i>	Digunakan untuk menambahkan foto Laundry
8	Harga_kilo	<i>Integer</i>	Digunakan untuk menambah harga per kilo
9	Status_Laundry	<i>Varchar(60)</i>	Digunakan untuk menambahkan status pesanan
10	User_id	<i>Integer</i>	<i>Foreign key</i>

c) Tabel *Laundry_galleries*

Tabel *Laundry_galleries* digunakan untuk menampung semua data-data Laundry

Tabel 4. 9 Tabel *Laundry galleries*

No	Nama kolom	Tipe data	Keterangan
1	<i>Id</i>	<i>Integer PK</i>	<i>Primary key</i>
2	<i>Laundry_id</i>	<i>Integer</i>	<i>Foreign key</i>
3	Caption	<i>Varchar(60)</i>	Caption Laundry
4	Foto	<i>Varchar(60)</i>	Foto Laundry

d) Tabel Ulasan

Tabel ulasan digunakan untuk menampung semua data-data Laundry

Tabel 4. 10 Tabel Ulasan

No	Nama kolom	Tipe data	Keterangan
1	<i>Id</i>	<i>Integer PK</i>	<i>Primary key</i>
2	<i>Laundry_id</i>	<i>Integer</i>	<i>Foreign key</i>
3	User_id	<i>integer</i>	<i>Foreign key</i>
4	Ulasan	<i>Varchar(100)</i>	Ulasan Laundry

e) Tabel *Banner*

Tabel *Banner* digunakan untuk menampung semua data-data Laundry

Tabel 4. 11 Tabel *Banner*

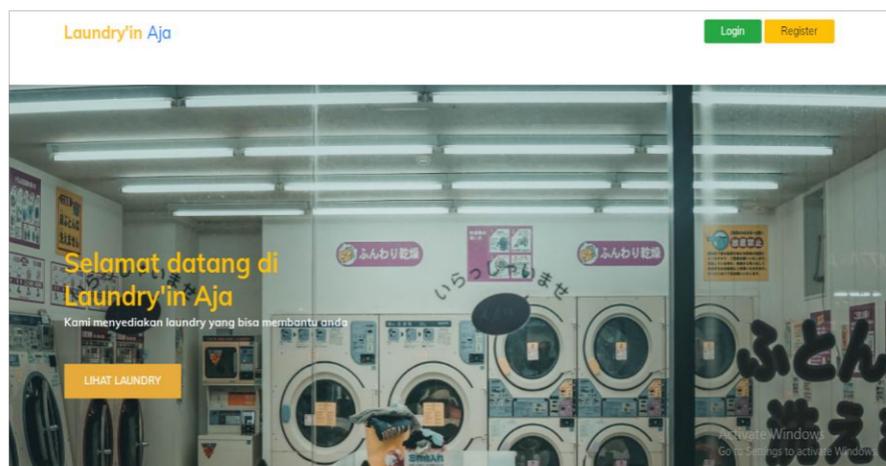
No	Nama kolom	Tipe data	Keterangan
1	<i>Id</i>	<i>Integer PK</i>	<i>Primary key</i>
2	Title	<i>Varchar(45)</i>	<i>Foreign key</i>
3	Sub_title	<i>Varchar(45)</i>	<i>Foreign key</i>
4	Is_active	<i>Integer</i>	<i>Primary key</i>

E. Implementasi *User Interface*

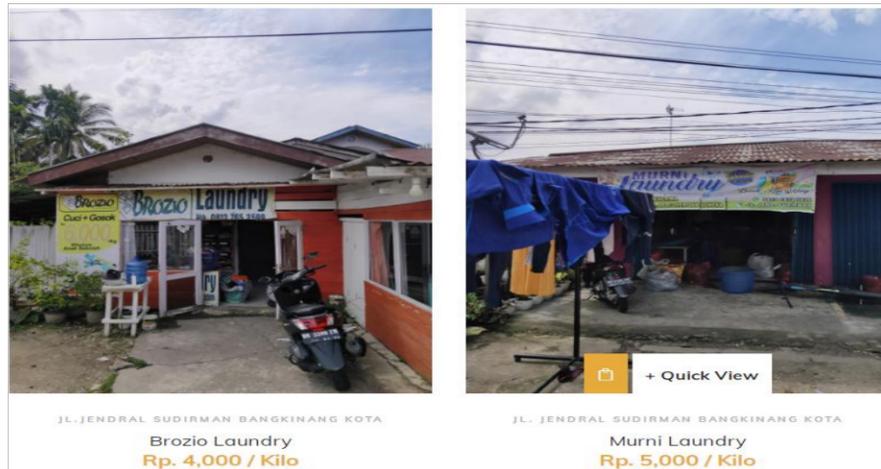
Implementasi adalah tahap penerapan sekaligus pengujian bagi sistem berdasarkan hasil analisa dan perancangan sistem, pada tahap ini merupakan tahap implementasi hasil menjadi sebuah Sistem Informasi *E-Laundry*

1. Halaman utama

Halaman utama adalah tempat penyampaian informasi *Laundry* terbaru, pada halaman utama ditampilkan dua *Laundry* yang baru mendaftar yang sebelumnya telah di verifikasi oleh admin untuk ditampilkan pada halaman utama *website*



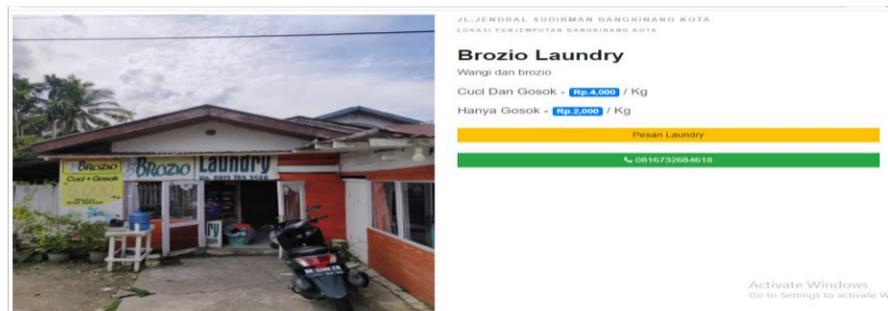
Gambar 4. 4 Halaman Utama



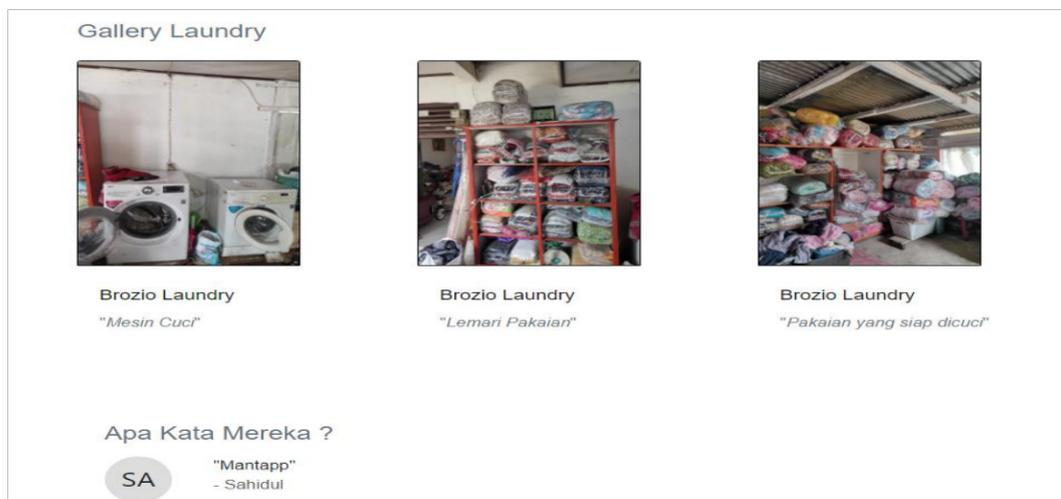
Gambar 4. 5 Halaman Utama dan Detail

2. Halaman Detail Laundry

Pada halaman detail Laundry ditampilkan detail-detail tentang Laundry



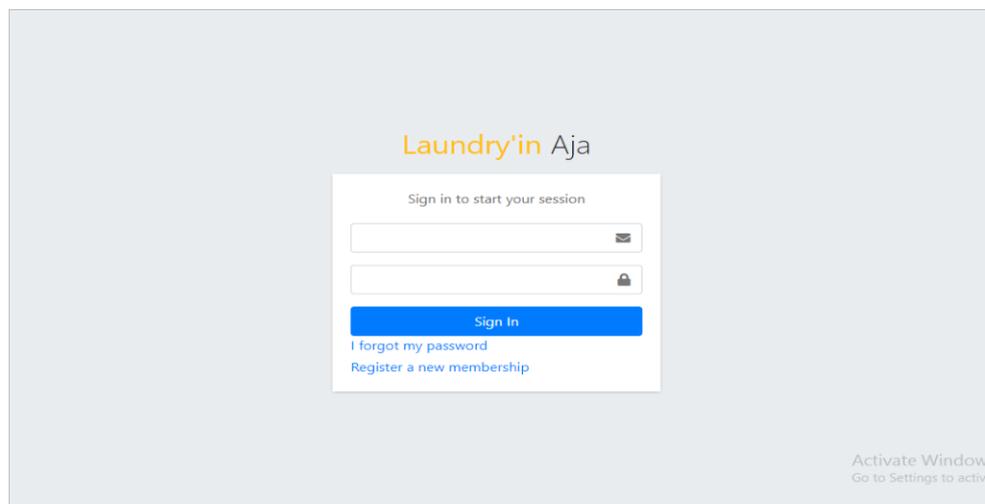
Gambar 4. 6 Halaman Detail Laundry



Gambar 4. 7 Halaman Detail Laundry

3. Halaman Login

Login merupakan tampilan untuk melanjutkan penggunaan sistem seperti mengakses *dashboard* ataupun melakukan *booking Laundry*, pada saat melakukan *login username* dan *password* yang dimasukan harus sesuai dengan *value* yang ada pada *database*,



Gambar 4. 8 Halaman *Login*

4. Halaman *Register*

Merupakan halaman untuk melakukan pendaftaran pada *website*. *form registrasi* yang tersedia sesuai dengan data pribadi, pada saat melakukan pendaftaran pengunjung *website* dapat memilih hak akses seperti pemilik Laundry ataupun Customer.

Gambar 4. 9 Halaman Register

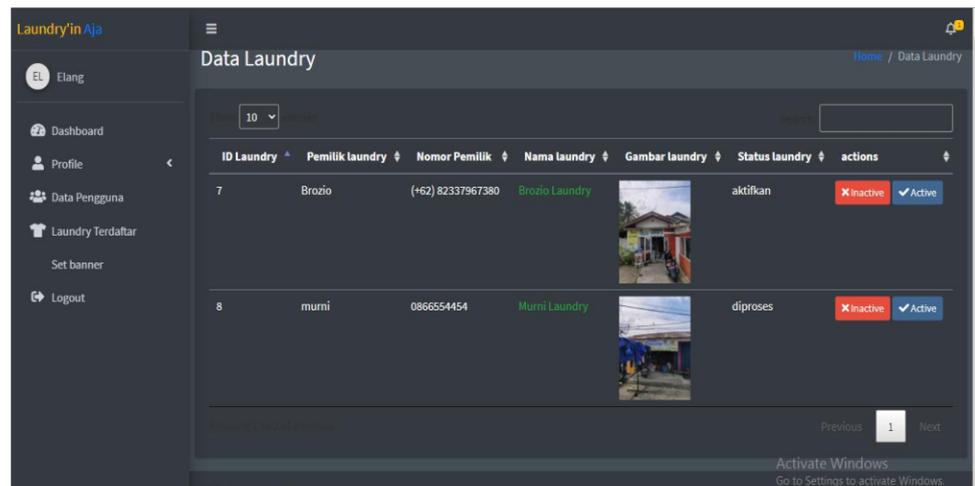
5. Halaman *Form* tambah *Laundry*

Halaman *Form* Tambah *Laundry* adalah tempat untuk mengisi *Form* pendaftaran *Laundry*

Gambar 4. 10 Halaman Form tambah *Laundry*

6. Halaman *Dashboard* Admin

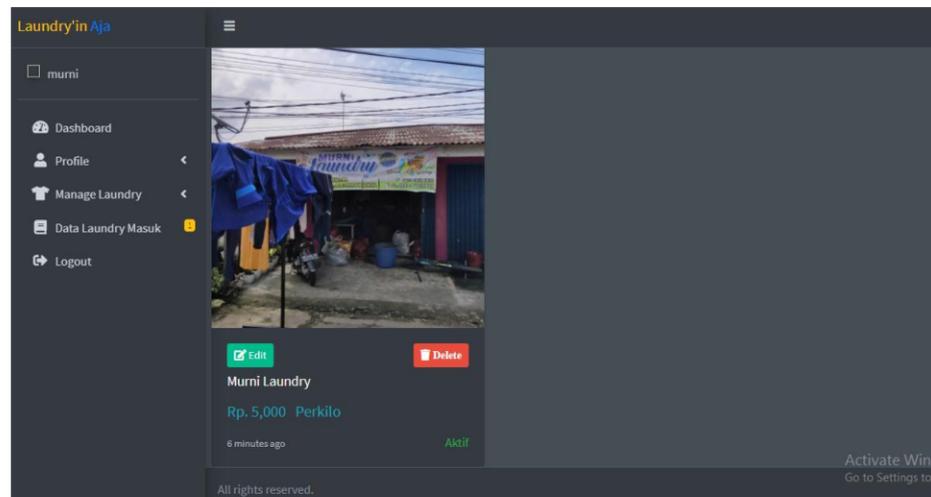
Dashboard Admin merupakan tempat penerimaan / verifikasi *Laundry* yang mendaftar.



Gambar 4. 11 Halaman Dashboard Admin

7. Halaman *Dashboard Pemilik Laundry*

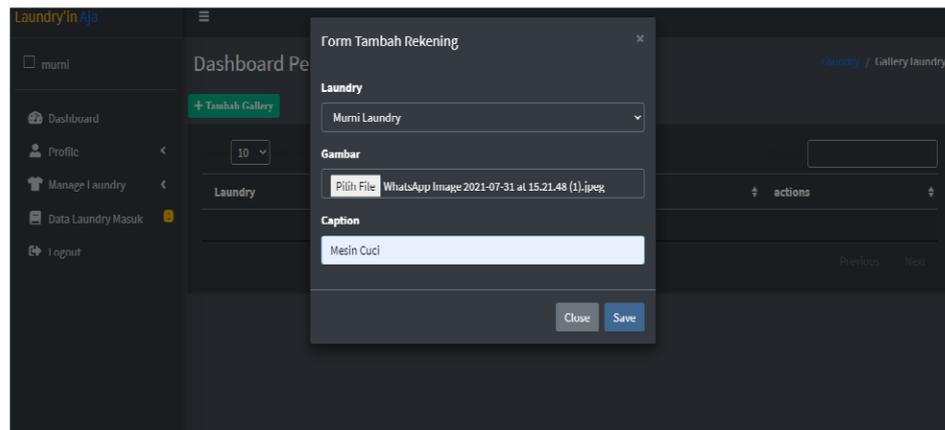
Terdapat notifikasi dari Admin bahwa *Laundry* yang telah mendaftar sudah diverifikasi oleh Admin



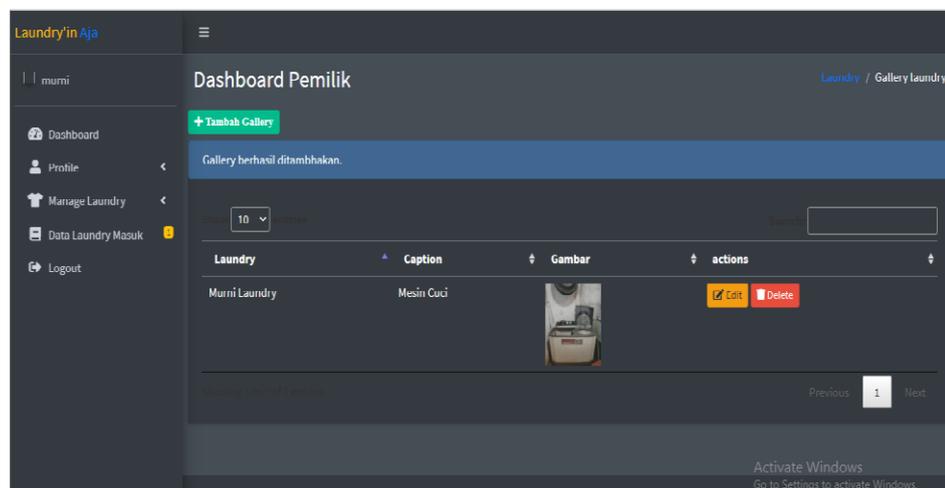
Gambar 4. 12 Halaman Dashboard Pemilik Laundry

8. Halaman *form tambah gallery Laundry*

Pemilik *Laundry* dapat menambahkan foto *gallery* sesuai kebutuhan



Gambar 4. 13 Halaman *form* tambah *gallery Laundry*



Gambar 4. 14 Halaman *form* tambah *gallery Laundry*

9. Halaman *booking Laundry*

Halaman ini ditampilkan untuk *Customer* yang telah mendaftar sebelumnya dan ingin melakukan *booking Laundry*

Booking laundry - Brozio Laundry

Home > Detail

Nama Pemesan
Sahidul

Alamat jemput
Jl. Ali Rasyid

Rt
01

Rw
02

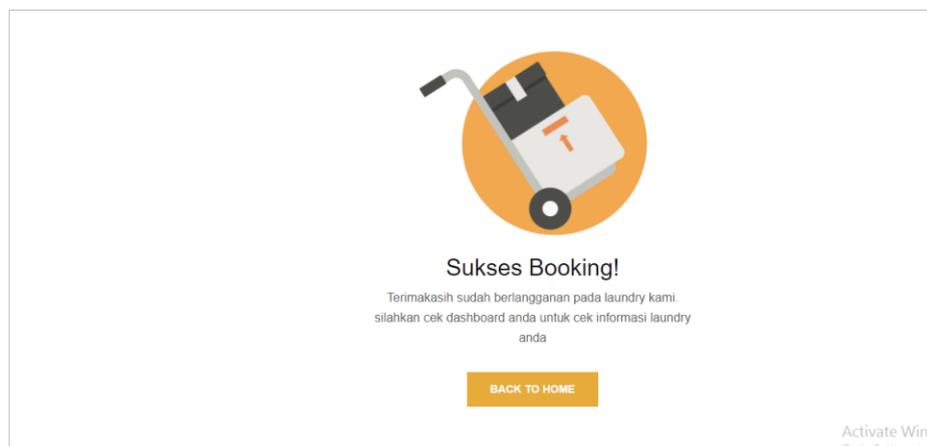
Hanya Gosok Rp. 2,000 / Kilo Cuci Dan Gosok Rp. 4,000 / Kilo

Berapa kilo
9

Keterangan / Detail Penjemputan
Brozio Laundry Hanya Menerima Jemputan Pada Wilayah Bangkinang Kota
pisahkan baju putih..

Close Booking

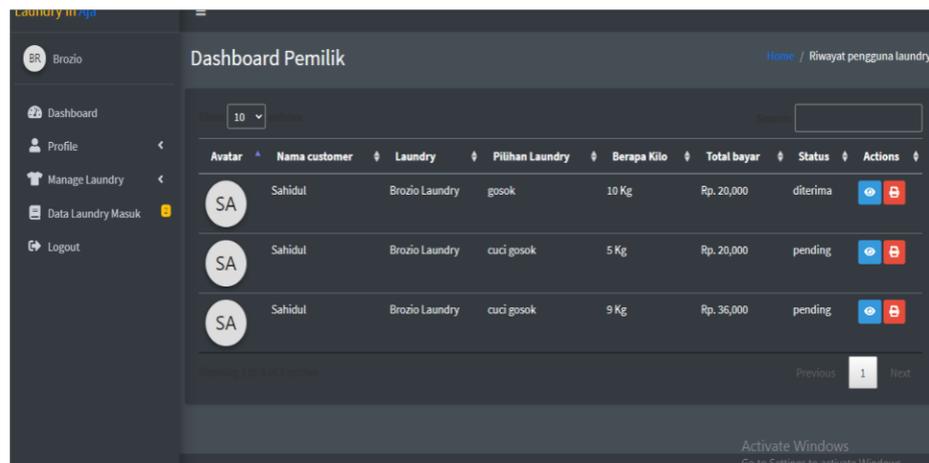
Gambar 4. 15 Halaman *booking Laundry*



Gambar 4. 16 Halaman Sukses *Booking*

10. Halaman *Dashboard Laundry* masuk

Halaman ini menampilkan *Laundry* yang telah *dibooking* masuk kedalam *dashboard* pemilik *Laundry*

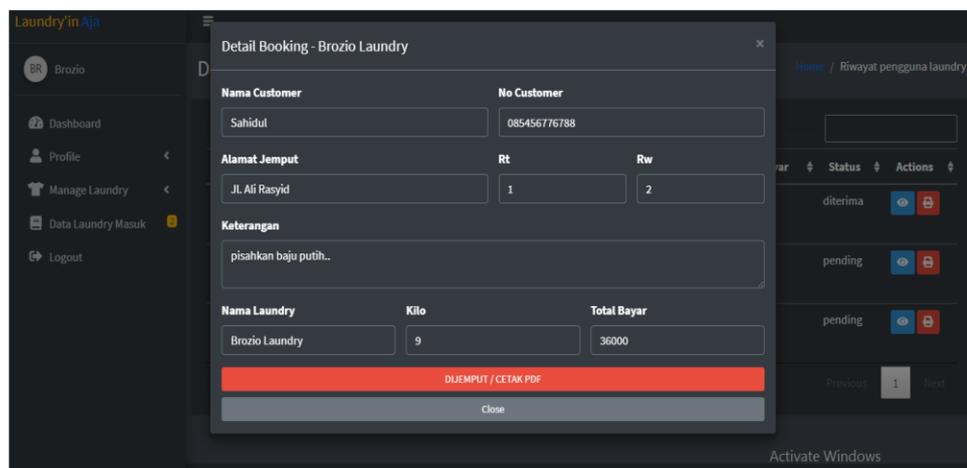


Avatar	Nama customer	Laundry	Pilihan Laundry	Berapa Kilo	Total bayar	Status	Actions
SA	Sahidul	Brozio Laundry	gosok	10 Kg	Rp. 20,000	diterima	 
SA	Sahidul	Brozio Laundry	cuci gosok	5 Kg	Rp. 20,000	pending	 
SA	Sahidul	Brozio Laundry	cuci gosok	9 Kg	Rp. 36,000	pending	 

Gambar 4. 17 Halaman Dashboard *Laundry* masuk

11. Halaman *form detail booking* pemilik *Laundry*

Halaman ini menampilkan *form* yang telah *dibooking* oleh *customer* dan pencetakan struk pembayaran *Laundry*



Detail Booking - Brozio Laundry

Nama Customer: Sahidul, No Customer: 085456776788

Alamat Jemput: Jl. Ali Rasyid, Rt: 1, Rw: 2

Keterangan: pisahkan baju putih..

Nama Laundry: Brozio Laundry, Kilo: 9, Total Bayar: 36000

DIEMPUT / CETAK PDF

Close

Gambar 4. 18 Halaman *form detail booking* pemilik *Laundry*

Invoice #Booking -898514		Mon, 02 Aug 2021	
Laundry: Brozio Laundry. Jl. Jendral Sudirman Bangkinang Kota 0816732684618 brozio@gmail.com		Pelanggan: Sahidul Jl. Ali Rasyid 085456776788 sahidul@gmail.com	
Produk	Harga / Kg	Berat	Subtotal
cuci gosok	36000	9 Kg	Rp 36000
Total			Rp 36000

Gambar 4. 19 Halaman Nota Pemesanan

12. Halaman notifikasi *customer*

Halaman ini menampilkan notifikasi bahwa pakaian sudah selesai dicuci

Laundry'in Aja

Sahidul

Dashboard Customer

Home / Riwayat laundry saya

Terimakasih sudah berlangganan dengan kami. Silahkan berikan ulasan anda tentang laundry kami pada menu ulasan.

Selamat datang **Sahidul**

- Jika anda sudah menerima pakaian laundry anda silahkan klik tombol diterima
- Ketika anda sudah menekan tombol terima anda dapat melakukan booking kembali
- Dan berikan ulasan Pada laundry.

10

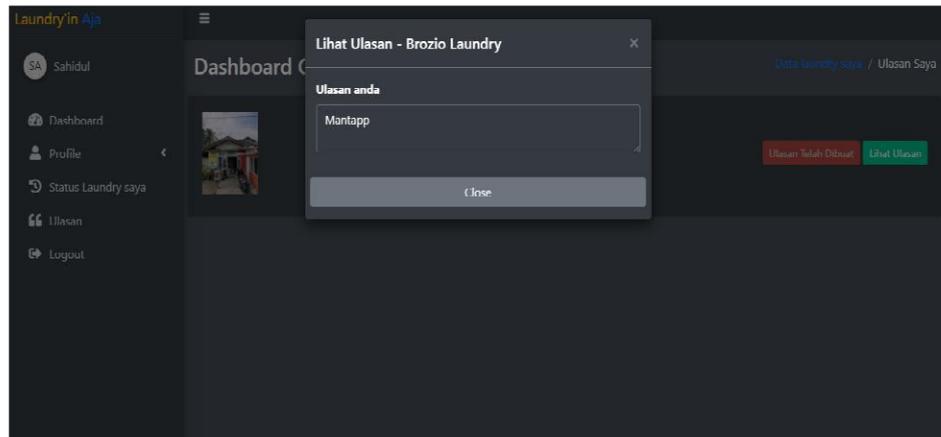
Foto laundry	Nama laundry	Berapa kilo	Status Booking	Total harga		Actions
	Brozio Laundry 4000 / kg	10 Kg	diterima	Rp. 20,000	16 hours ago	Booking

All rights reserved. Activate Windows. Go to Settings to activate Windows. Version 3.1.0.

Gambar 4. 20 Halaman notifikasi *customer*

13. Halaman ulasan *customer*

Halaman ini menampilkan ulasan yang diberikan oleh customer kepada pemilik



Gambar 4. 21 Halaman ulasan *customer*

F. Hasil Pengujian Black Box

Pengujian sistem dilakukan menggunakan metode *Black Box testing*. Metode *Black box testing* merupakan pengujian kualitas perangkat lunak yang berfokus pada fungsionalitas perangkat lunak. Pengujian *black box testing* bertujuan untuk menemukan fungsi yang tidak benar, kesalahan antarmuka, kesalahan pada struktur data, kesalahan performansi, kesalahan inisialisasi dan terminasi. Teknik yang digunakan untuk melakukan *test cases*. Tabel pengujian sebagai berikut:

Tabel 4. 12 Pengujian *Black Box*

No	Deskripsi pengujian	Prosedur pengujian	Kriteria hasil evaluasi	
			Benar	Tidak
1.	Tampilan Login	1. Input <i>username</i> dan <i>password</i> 2. Klik tombol <i>login</i>	✓	
2.	Tampilan <i>Register</i>	1. <i>Input form register</i> 2. Klik tombol <i>register</i>	✓	
3.	Tampilan <i>Dashboard Admin</i>	1. Klik tampilan data pengguna 2. Klik tampilan Laundry terdaftar 3. Klik set data banner 4. Notifikasi Laundry masuk 5. Verifikasi Laundry yang mendaftar 6. Klik tombol <i>Logout</i>	✓	

4.	Tampilan <i>Dashboard</i> Pemilik Laundry	<ol style="list-style-type: none"> 1. Klik pengaturan <i>profile</i> 2. Klik <i>manage</i> Laundry 3. Klik daftar laundry 4. Klik upload gallery Laundry 5. Tampilan Laundry masuk 6. Klik tombol <i>Logout</i> 	✓	
5.	Tampilan <i>Dashboard Customer</i>	<ol style="list-style-type: none"> 1. Klik pengaturan <i>profile</i> 2. Klik tombol status Laundry 3. Tampilan status Laundry 4. Klik masukan ulasan 	✓	
6.	Tampilan halaman utama <i>Website</i> Laundry	<ol style="list-style-type: none"> 1. Tampilan <i>Banner Website</i> 2. Tampilan pilihan Laundry 3. Klik detail Laundry 4. <i>Search</i> pilihan Laundry 	✓	
7.	Tampilan halaman kedua <i>Website</i>	<ol style="list-style-type: none"> 1. Tampilan deskripsi Laundry 2. Klik tombol pesan Laundry 3. Tampilan gallery Laundry 4. <i>Input Form booking</i> Laundry 	✓	

BAB V

PENUTUP

A. Kesimpulan

Berdasarkan dari hasil analisa data pada bab sebelumnya, dapat diperoleh beberapa kesimpulan, diantaranya adalah:

1. *Website* ini menyampaikan informasi *Laundry* yang ada disekitar Bangkinang Kota bagi *customer* dan dapat mempermudah pemilik *Laundry* dalam melakukan proses promosi *Laundry*-nya,
2. *Website* ini juga dapat memudahkan *customer* yang ingin melakukan *booking Laundry* serta berkomunikasi dengan pemilik *Laundry*, dengan cara melakukan pendaftaran pada *Website* melalui halaman registrasi.

B. Saran

Saran untuk penelitian selanjutnya adalah sebagai berikut :

1. Mempermudah komunikasi antara Customer dan pemilik *Laundry*, sistem ini juga dapat dikembangkan kedepannya.
2. Memberikan kemudahan dalam proses pencarian dan pemesanan *Laundry*, sistem ini dapat dikembangkan dengan membuat sistem informasi berbasis *android*.
3. Mempermudah alur transaksi antara Customer dan pemilik *Laundry* sistem ini dapat dikembangkan lagi dengan membuat fitur yang lain

DAFTAR PUSTAKA

- Batubara, F. A. (2012). *Perancangan Website Pada Pt . Ratu Enim Palembang*. 15–27.
- Cholifah, W. N., Yulianingsih, Y., & Sagita, S. M. (2018). *Pengujian Black Box Testing pada Aplikasi Action & Strategy Berbasis Android dengan Teknologi Phonegap. STRING (Satuan Tulisan Riset Dan Inovasi Teknologi)*, 3(2), 206. <https://doi.org/10.30998/string.v3i2.3048>
- Cimperman, R. (2006). *UAT Defined: A Guide to Practical User Acceptance Testing (Digital Short Cut)*. Pearson Education.
- Handika, I. G., & Purbasari, A. (2018). *Pemanfaatan Framework Laravel Dalam Pembangunan Aplikasi E-Travel Berbasis Website*. Konferensi Nasional Sistem Informasi, 1329–1334.
- Ii, B. A. B. (2015). *Bab ii tinjauan pustaka dan dasar teori 2.1*. 6–19.
- Kasus, S., Sandaran, P. T., & Abadi, S. (2018). *No Title*. 3(1), 9–18.
- Lestari, D. Y., Kusnandar, I., Muhafidin, D., Studi, P., Ilmu, M., Negara, A., & Tinggi, S. (2020). *Pengaruh Implementasi Kebijakan Terhadap Transparansi Pengadaan Barang / Jasa Pemerintah*. 7, 180–193.
- Lubis, M. A., Anroni, Y. E., & Lisa, T. A. (2019). *Startup Jasa Jemput Antar Laundry Berbasis Web*. 6(1), 117–129.
- Munthe, R., Insap Santosa, P., & Ferdiana, R. (2015). *Usulan Metode Evaluasi User Acceptance Testing (UAT) dalam Pengembangan Perangkat Lunak*. Prosiding Seminar Nasional Pendidikan Teknik Informatika, September, 3. <http://pti.undiksha.ac.id/senapati>
- Mustaqbal, M. S., Firdaus, R. F., & Rahmadi, H. (2015). *Pengujian Aplikasi Menggunakan Black Box Testing Boundary Value Analysis (Studi Kasus : Aplikasi Prediksi Kelulusan SNMPTN)*. 1(3), 31–36.
- Nana, D., & Elin, H. (2018). *Memilih Metode Penelitian Yang Tepat: Bagi Penelitian Bidang Ilmu Manajemen*. Jurnal Ilmu Manajemen, 5(1), 288. <https://jurnal.unigal.ac.id/index.php/ekonologi/article/view/1359>
- Perry, W. E. (2007). *Effective Methods for Software Testing, CafeScribe: Includes Complete Guidelines, Checklists, and Templates*. John Wiley & Sons.
- Riyadli, H., Arliyana, A., & Saputra, F. E. (2020). *Rancang Bangun Sistem Informasi Keuangan Berbasis WEB*. Jurnal Sains Komputer Dan Teknologi Informasi, 3(1), 98–103. <https://doi.org/10.33084/jsakti.v3i1.1770>

- Satzinger, J.W, Jackson, R.B, dan Burd, S. D. (2011). *Systems and analysis design in a changing world fifth edition*.
- Septiani, M., & Kuryanti, S. J. (2018). *Sistem Pakar untuk Mendiagnosa Penyakit Saluran Pernapasan pada Anak*. Publikasi Jurnal & Penelitian Teknik Informatika, 2(2), 23–27.
- Simargolang, M. Y., & Nasution, N. (2018). *Aplikasi Pelayanan Jasa Laundry Berbasis WEB (Studi Kasus : Pelangi Laundry Kisaran)*. 2(1).
- Siradjuddin, H. K. (2018). *IJIS Indonesian Journal on Information System e-ISSN 2548-6438 p-ISSN 2614-7173. Sistem Informasi Pariwisata Sebagai Media Promosi Pada Dinas Kebudayaan Dan Pariwisata Kota Tidore Kepulauan*, 3(September 2018), 46–55.
- Suendri. (2018). No Title. *Implementasi Diagram UML (Unified Modelling Language) Pada Perancangan Sistem Informasi Remunerasi Dosen Dengan Database Oracle (Studi Kasus: UIN Sumatera Utara Medan)* Suendri, 6341(November), 1–9.
- Syukroni, M. F. (2017). *Rancang Bangun Knowledge Management Sistem Berbasis Web Pada Madrasah Mualimin Al-Islamiah Uteran Geger Madiun*. Teknik Informatika Universitas Muhammadiyah Ponorogo, 7–35.