

1. Pendahuluan

Language Training Center (LTC) adalah salah satu unit yang berada di bawah naungan Universitas Kristen Satya Wacana (UKSW), sesuai dengan visinya *To Serve, Support, and Promote the Teaching and Learning of Foreign Languages and Cultures*, LTC hadir untuk melayani serta mendukung proses belajar dan mengajar dalam bahasa asing dan budaya. Sebagai salah satu bentuknya LTC melayani untuk program kursus bagi mahasiswa UKSW serta terbuka untuk pelajar minimal SMA, mahasiswa dari luar UKSW bahkan masyarakat umum. Dari data peserta yang dimiliki oleh LTC, banyak peminat yang berasal dari luar kota maupun dalam kota, kebutuhan akan peningkatan pelayanan harus lebih ditingkatkan. Peserta mendaftar dengan mengisi *form registration* dalam bentuk *hard copy*, setelah selesai mengisi *form registration*, *receptionist* harus memasukkan data ke dalam *excel* sesuai program kursus yang diambil. Pendaftaran menggunakan *form registration* dalam bentuk *hard copy* dan untuk mengetahui tagihan pembayaran peserta harus bertanya ke *receptionist*. Hal ini membuat calon peserta dan peserta harus datang langsung ke LTC, sehingga menyulitkan untuk yang tinggal di luar daerah Salatiga.

Dengan semakin majunya teknologi pada masa sekarang ini, kebutuhan akan informasi yang cepat dan akurat sangat diperlukan di segala bidang. Salah satu bentuk perkembangan teknologi pada bidang komputer pada zaman ini adalah *website*. *Website* merupakan salah satu layanan yang menyediakan informasi secara *update* yang dapat diakses tanpa dibatasi oleh jarak dan waktu [1].

Berdasarkan permasalahan yang ada maka dibuatlah Aplikasi Manajemen Kursus Berbasis *Web* menggunakan *Framework* Laravel. Aplikasi ini berisikan tentang manajemen data peserta, data *payment* dan data pengajar sehingga calon peserta lebih mudah dalam melakukan pendaftaran, melihat jadwal kursus dan peserta bisa mengetahui tagihan pembayaran. Dari sisi administrator, akan mempermudah dalam manajemen data peserta dan data pembayaran. Aplikasi ini berbasis *web* sehingga mempermudah *user* dalam mendapatkan informasi.

Rumusan masalah penelitian ini adalah bagaimana membangun suatu *website* yang dapat membantu calon peserta, sehingga dapat memperoleh informasi, bisa melakukan pendaftaran, dapat melihat tagihan. Dan dari segi administrator dapat mengelola data peserta dan data *payment*. Manfaat yang diharapkan adalah dapat menyediakan *website* yang tidak hanya berfungsi sebagai *web profile*, namun juga dapat digunakan sebagai media pengelolaan yang dapat menampilkan konten yang aktual dan berguna bagi pengguna maupun administrator.

2. Tinjauan Pustaka

Salah satu penelitian yang membahas tentang *website* menggunakan *framework* Laravel adalah penelitian yang berjudul Membangun *website* Sistem Informasi Sekolah dengan menggunakan *Framework* Laravel pada SMA Negeri 4 Pontianak, membahas tentang sistem informasi sekolah menggunakan *Framework* Laravel dan *My SQL* sebagai *database* untuk memberikan informasi mengenai SMA Negeri 4 Pontianak dan membantu dalam administrasi pada SMA tersebut. Berdasarkan penelitian tersebut dihasilkan sebuah sistem yang bisa memproses administrasi yang dibutuhkan di SMA tersebut. Sistem ini menggunakan *framework* Laravel dan *SQL* sebagai *database* [2].

Penelitian lain yang membahas tentang *website* menggunakan *framework* Laravel adalah yang berjudul Analisis dan Perancangan Toko *Online* Geraihana berbasis SEO menggunakan Laravel. Penelitian ini telah membahas tentang pembuatan sistem *web ecommerce*. Hasil dari penelitian ini adalah sebuah sistem yang dapat mempermudah dalam melakukan promosi dan penjualan barang toko Geraihana. Pengguna sistem ini ada dua yaitu administrator dan *user*. Proses penjualan atau pembelian barang bisa dilakukan oleh administrator dan *user*, karena memang sistem ini bertujuan untuk mempermudah administrator untuk melakukan penjualan dan mempermudah pembeli agar lebih cepat dan akurat dalam membeli barang [3].

Salah satu penelitian yang membahas tentang *website* sistem informasi manajemen kursus berbasis *web* adalah penelitian yang berjudul Sistem Informasi Akademik Lembaga Kursus dan Pelatihan Berbasis *Web*, membahas tentang sistem informasi lembaga kursus menggunakan *PHP* dan *DBMS SQL* sebagai *database* untuk membantu pengelolaan data akademik Lembaga Kursus dan Pelatihan (LKP) *GET-House of English*. Berdasarkan penelitian tersebut dihasilkan sebuah sistem yang bisa memproses data pengguna, data siswa, data nilai, data jadwal dan pengelolaan laporan [4].

Perbedaan dengan penelitian yang dibahas sebelumnya adalah membahas mengenai sistem informasi sekolah dan toko *online*, sementara penelitian ini membahas mengenai aplikasi manajemen kursus. Perbedaan lainnya menggunakan *PHP* biasa dalam pembangunan sistem, sementara penelitian ini menggunakan *framework* Laravel dalam membangun sistem. Berdasarkan penelitian-penelitian yang pernah dilakukan terkait sistem informasi berbasis *web*, maka akan dilakukan penelitian yang membahas tentang Perancangan Aplikasi Manajemen Kursus Berbasis *Web* menggunakan *Framework* Laravel. Aplikasi dibagi menjadi dua tampilan, yang pertama untuk *user*, dan yang kedua untuk *admin web*. Penelitian ini diharapkan dapat memberikan kemudahan kepada *customer* untuk mengakses informasi melalui *web* serta memberi kemudahan kepada administrator dalam mengelolah data *customer*.

Sistem Aplikasi Manajemen Kursus dalam bentuk *website* yang dapat diartikan sebagai kumpulan halaman yang menampilkan informasi data teks, data gambar diam atau gerak, data animasi, suara, video dan atau gabungan dari semuanya, baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait dimana masing-masing dihubungkan dengan jaringan-jaringan halaman (*hyperlink*). Bersifat statis apabila isi informasi *website* tetap, jarang berubah, dan isi informasinya searah hanya dari pemilik *website*. Bersifat dinamis apabila isi informasi *website* selalu berubah-ubah, dan isi informasinya interaktif dua arah berasal dari pemilik serta pengguna *website*. Contoh *website* statis adalah berisi profil perusahaan, sedangkan *website* dinamis adalah seperti *facebook*, *multiply*, dan lain-lainnya. Dalam sisi pengembangannya, *website* statis hanya bisa di *update* oleh pemiliknya saja, sedangkan *website* dinamis bisa di *update* oleh pengguna maupun pemilik [5].

Sistem yang dibangun dalam penelitian ini menggunakan *framework* Laravel. *Framework* Laravel salah satu *framework* berbasis *PHP* (*Hypertext Preprocessor*) yang dibangun dengan konsep MVC (*Model-View-Controller*). Dilengkapi juga dengan *command line tool* yang bernama *artisan* yang bisa digunakan untuk *packing bundle* dan *instalasi bundle* melalui *command prompt*. Laravel memiliki beberapa fitur baru yang tidak dimiliki oleh *framework* lainnya, salah satunya adalah *Blade Templating* yang memungkinkan *developer* untuk membuat *template* yang lebih terstruktur dan dinamis, sehingga lebih mudah dalam pengembangan. *Framework* ini juga bersifat *expressif* dan *simple* yang bertujuan agar saat pertama kali digunakan oleh seorang *programmer* diharapkan akan langsung mengetahui

kegunaan dari sintaks tersebut. Salah satu yang membuat Laravel begitu *simple* adalah *Eloquent ORM* yang akan menyederhanakan proses *query* ke *database*, penerapan dari pola *active record* menyediakan metode internal untuk mengatasi kendala hubungan antara objek *database*. Berikut adalah beberapa fitur yang ada pada laravel: 1) *Reverse routing* mendefinisikan hubungan antara *link* dan *route* sehingga jika suatu saat ada perubahan pada *route* secara otomatis akan tersambung dengan *link* yang relevan. 2) *Automatic pagination* menyederhanakan tugas dari penerapan halaman menggantikan penerapan yang manual dengan metode otomatis yang terintegrasi ke laravel. 3) *Restful Controller* memberikan sebuah pilihan untuk memisahkan logika dalam melayani HTTP GET dan permintaan POST [6].

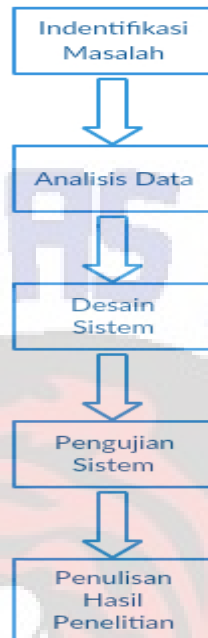
Aplikasi ini dilengkapi dengan fitur *SMS Gateway*, istilah *Gateway*, bila dilihat arti secara kamus Inggris-Indonesia adalah pintu gerbang. Namun pada dunia komputer, *gateway* bisa diartikan sebagai jembatan penghubung antar satu sistem dengan sistem yang lain yang berbeda, sehingga dapat terjadi pertukaran data antar sistem tersebut. Dengan demikian *SMS Gateway* dapat di artikan sebagai penghubung untuk lalu lintas data *SMS*, baik yang dikirimkan maupun yang diterima. *SMS (Short Messaging Services) Data SMS (Short Messaging Services)* yang kita kirim atau yang kita terima sebenarnya memiliki format tersendiri untuk dapat diterjemahkan oleh sebuah *mobile phone*. Format atau mode yang dipakai untuk mengirim dan menerima *SMS* ada dua yaitu *Mode Text* dan *Mode PDU (Protocol Data Unit)* [7].

Sistem Informasi adalah mencakup sejumlah komponen (manusia, komputer, teknologi informasi, dan prosedur kerja), ada sesuatu yang diproses (data menjadi informasi), dan dimaksudkan untuk mencapai suatu sasaran atau tujuan. Pengaturan orang, data, proses, dan teknologi informasi yang berinteraksi untuk mengumpulkan, memproses, menyimpan, dan menyediakan sebagai output informasi yang diperlukan untuk mendukung sebuah organisasi [4].

Pengertian Kursus adalah dalam dunia nonformal, istilah kursus tidak pernah dapat dilepaskan dari dunia pendidikan, karena kursus merupakan salah satu aktifitas pendidikan nonformal dalam upaya memberikan pertolongan kepada warga yang memerlukan sesuatu keterampilan dalam waktu yang relatif singkat. Dilihat dari segi sistematik istilah kursus itu sendiri merupakan terjemahan dari *course* yang berasal dari bahasa inggris dan secara harfiah kursus itu berarti mata pelajaran atau rangkaian pelajaran. Menurut Gressner (1998:26) mengemukakan bahwa yang dimaksud dengan kursus adalah kegiatan pendidikan yang berlangsung di dalam masyarakat yang dilakukan dengan sengaja, terorganisir, dan sistematik untuk memberikan satu mata pelajaran atau rangkaian pelajaran tertentu dalam waktu yang relatif singkat, agar mereka memperoleh pengetahuan, keterampilan dan sikap yang dapat dimanfaatkan untuk mengembangkan dirinya dan masyarakat. Dengan demikian kursus ini dibangun dan diselenggarakan sejajar dengan kebutuhan yang dirasakan oleh masyarakat baik untuk menambah keterampilan, usaha sosial ekonomi, pengisi waktu luang ataupun upaya pengembangan diri seseorang [8].

3. Metode Penelitian dan Perancangan Sistem

Penelitian yang dilakukan, diselesaikan melalui tahapan penelitian yang terbagi dalam beberapa tahapan, yaitu: (1) Identifikasi masalah, (2) Analisis Data, (3) Desain Sistem, (4) Pengujian sistem, (5) Pembuatan laporan penelitian.

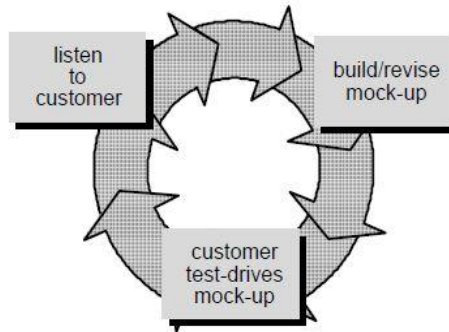


Gambar 1 Tahapan Penelitian

Gambar 1 merupakan tahapan-tahapan penelitian yang dapat dijelaskan sebagai berikut: *Tahap pertama*: identifikasi masalah adalah merumuskan masalah yang akan dijadikan sebagai objek penelitian. Perumusan masalah dilakukan dengan terlebih dahulu melihat kondisi aktual di lapangan. Berdasarkan masalah yang telah dirumuskan langkah selanjutnya adalah menentukan tujuan dari penelitian. Tujuan penelitian ini merupakan sasaran yang nantinya ingin diwujudkan dari penyelesaian permasalahan yang diteliti;

Tahap kedua: pengumpulan dan analisis data, metode pengumpulan data terdiri dari dua yaitu: observasi dan studi literatur. Observasi dilakukan dengan mengumpulkan data dan informasi yang dibutuhkan dalam perancangan, pengembangan sistem serta proses bisnis yang diterapkan. Studi literatur digunakan untuk mengumpulkan data dari penelitian terdahulu, pembelajaran dari berbagai macam literatur seperti buku, jurnal dan teori-teori yang mendukung penelitian, yang akan digunakan dan data penunjang lainnya yang berkaitan dengan sistem. Analisis kebutuhan aplikasi dilakukan untuk mengetahui kebutuhan pengguna terhadap aplikasi yang dibangun.

Tahap ketiga: desain sistem, metode yang digunakan dalam perancangan dan pembuatan sistem ini adalah *Prototype Model*. *Prototype model* merupakan proses iteratif dalam pengembangan sistem dimana kebutuhan diubah kedalam sistem yang bekerja dan secara terus-menerus diperbaiki melalui kerjasama antara pengguna dan analis;



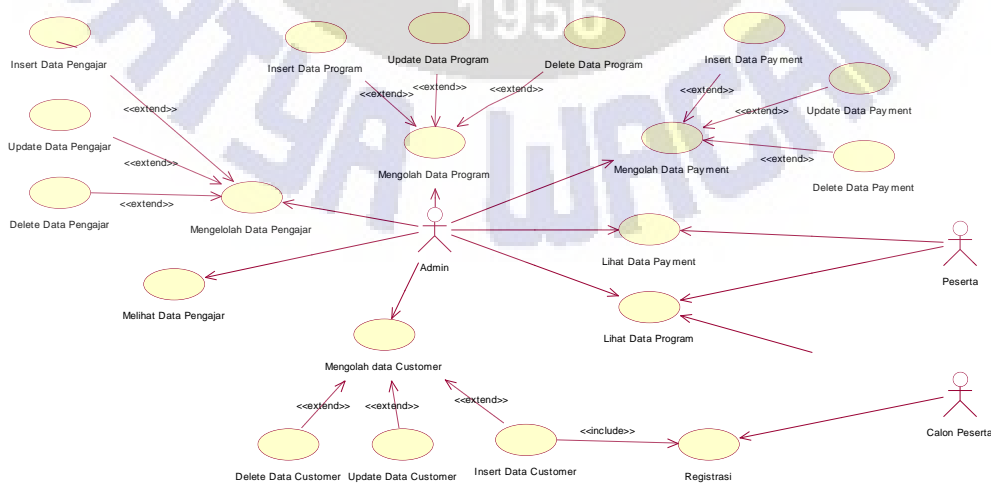
Gambar 2 Prototype Model

Tahap *listen to customer* dilakukan wawancara dengan dengan pihak LTC maupun peserta, hasil dari wawancara yang dilakukan adalah: 1) Peserta mengharapkan adanya *website* yang dapat bisa untuk melakukan pendaftaran, melihat jadwal, dan bisa mengetahui tagihan, 2) LTC mengharapkan aplikasi yang mudah untuk digunakan dalam mengolah data peserta dan data pembayaran, 3) Website yang dapat berfungsi juga sebagai *web profile*. Berdasarkan data dan kebutuhan sistem yang telah dikumpulkan lalu dilakukan *build/revise mock-up* yaitu dilakukan pembuatan *prototype sistem*. Tahap *customer test-drives mock-up* dilakukan evaluasi oleh pengguna terhadap sistem yang telah dibuat.

Tahap keempat: pengujian sistem adalah proses untuk memastikan bahwa aplikasi yang dikembangkan bebas dari kesalahan, dilakukan uji coba pada aplikasi tersebut. Pada tahap ini juga akan dilakukan evaluasi terhadap hasil penelitian yang dilakukan. Evaluasi dilakukan mencakup evaluasi hasil dan manfaat dengan membandingkan hasil yang didapatkan dengan kebutuhan pengguna;

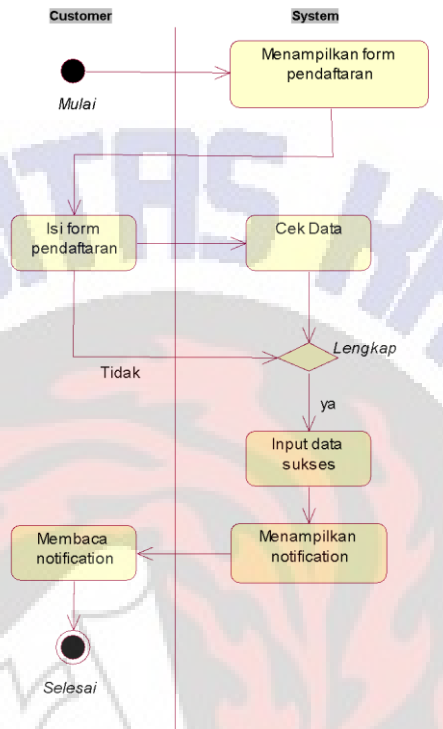
Tahap kelima: Langkah terakhir dari penelitian ini adalah membuat laporan penelitian. Laporan ini berisi hal-hal yang dikerjakan selama penelitian dan hasil yang didapatkan pada saat melakukan penelitian.

Perancangan proses dalam sistem yang dibangun menggunakan *Unified Modeling Language (UML)* untuk memberikan gambaran *visual*, rancangan, dan spesifikasi sistem. Dalam sistem yang dibangun digunakan beberapa UML yaitu: *usecase diagram*, *activity diagram*, *sequence diagram* dan *class diagram*.



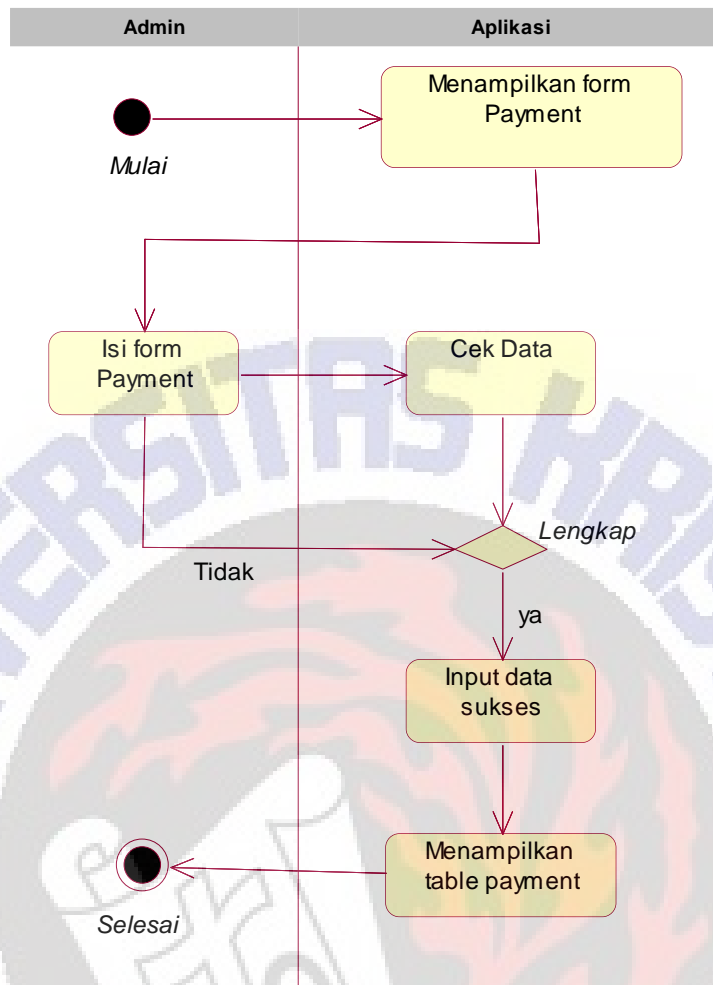
Gambar 3 Use Case Diagram Sistem

Gambar 3 menunjukkan *use case diagram* dari sistem. *Use case diagram* terdiri dari 2 aktor yaitu administrator dan peserta. Berdasarkan *use case diagram* tersebut administrator dapat mengelola data program, mengolah data *customer* dan mengolah data *payment*. Calon peserta dapat melakukan *registrasi*, melihat jadwal (data program) dan bisa melihat data *payment*.



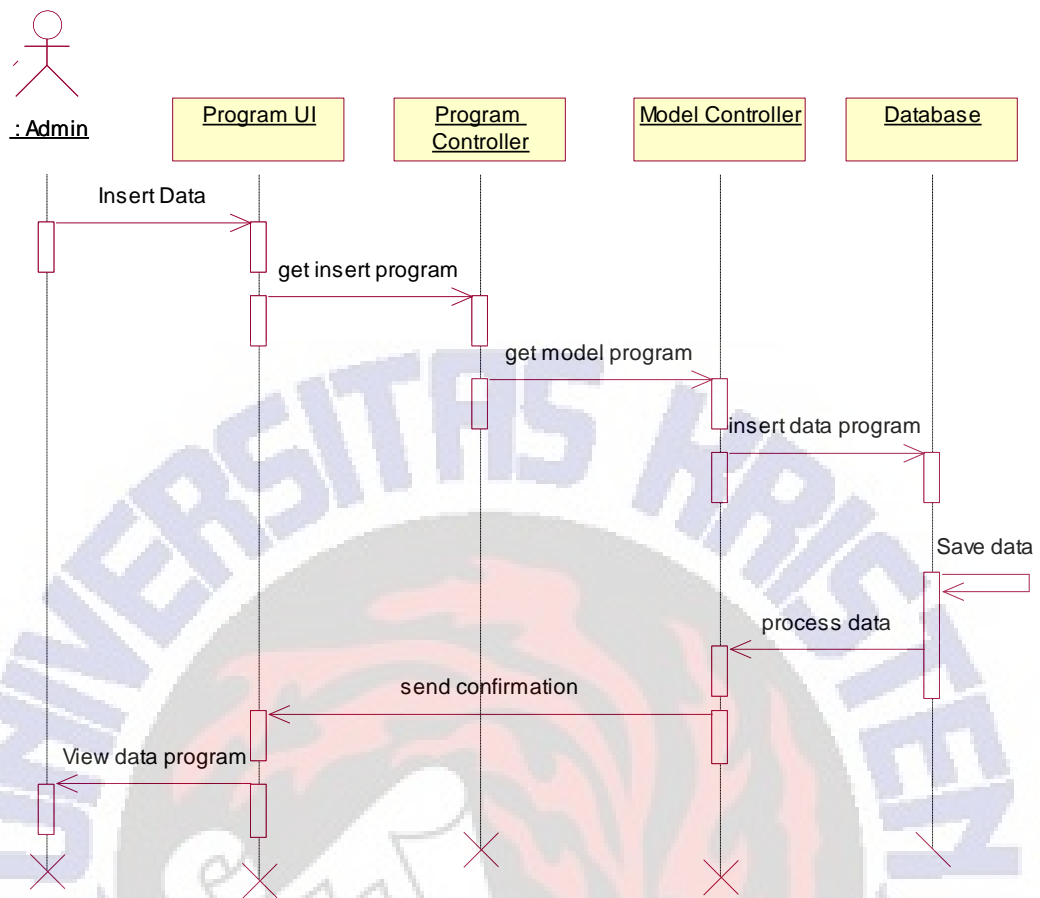
Gambar 4 Activity Diagram Pendaftaran

Gambar 4 menunjukkan aktivitas sistem dalam hal melakukan pendaftaran sebagai *customer*. Dimulai dengan sistem akan menampilkan *form* pendaftaran dan administrator atau calon peserta akan mengisi *form* pendaftaran. Sistem melakukan pengecekan mengenai data yang dimasukkan sudah sesuai kebutuhan sistem atau belum. Sistem akan menampilkan pesan bahwa data yang di masukkan belum sesuai kebutuhan sistem. Data yang telah sesuai dengan kebutuhan sistem akan disimpan ke dalam *database* dan proses selesai.



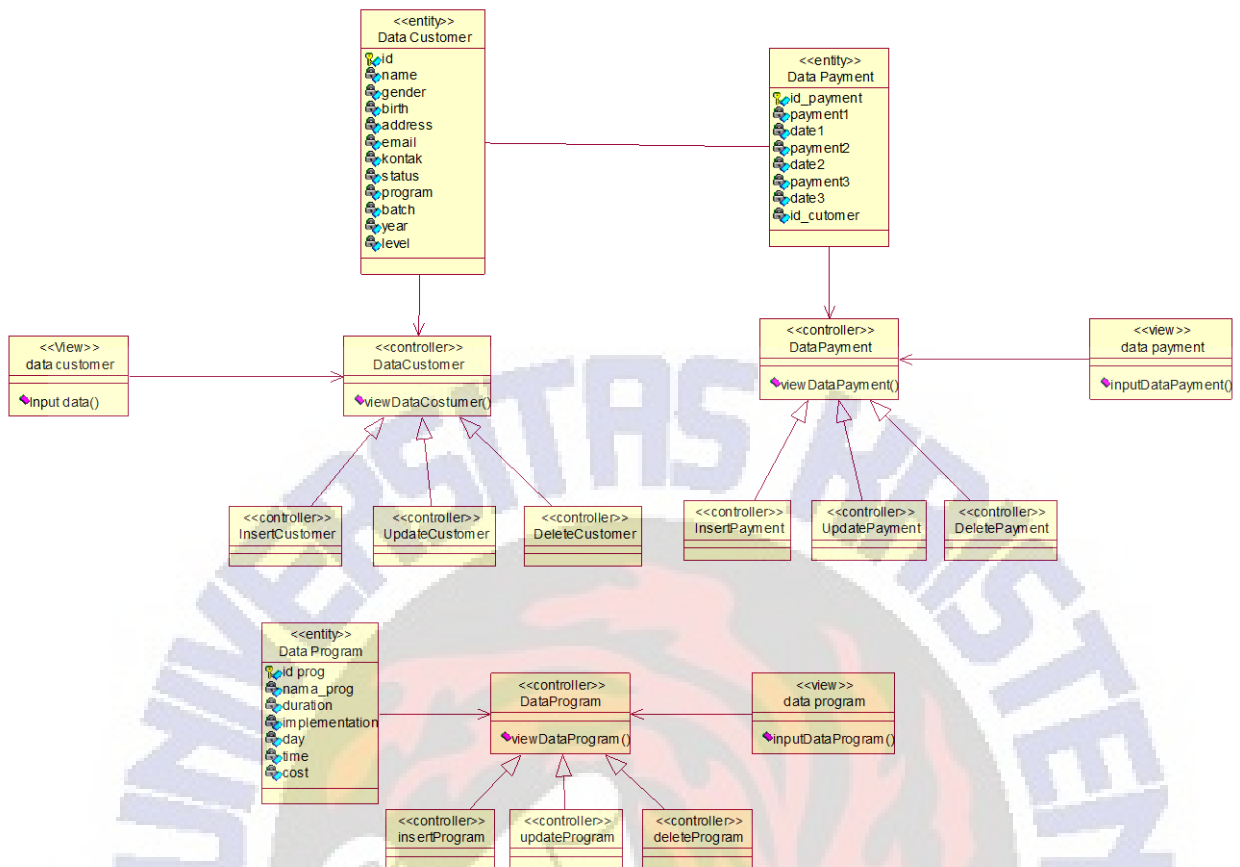
Gambar 5 Activity Diagram Pembayaran

Gambar 5 menunjukkan aktivitas sistem dalam hal memasukkan data pembayaran dari customer sebagai administrator. Sistem akan menampilkan *form payment* dan administrator akan mengisi *form payment*. Sistem melakukan pengecekan apakah data yang dimasukkan sudah sesuai kebutuhan atau belum. Sistem akan menampilkan pesan bahwa data yang di masukkan belum sesuai kebutuhan sistem. Data yang telah sesuai dengan kebutuhan sistem akan disimpan ke dalam *database* dan proses selesai.



Gambar 6 *Sequence Diagram* Input Data Program

Gambar 6 menunjukkan *sequence diagram* input data program. Administrator memasukkan data program. Sistem mengirim data program dari *controller* ke *model* dan disimpan ke *database*. Sistem mengirim konfirmasi, dan menampilkan data program kepada administrator.

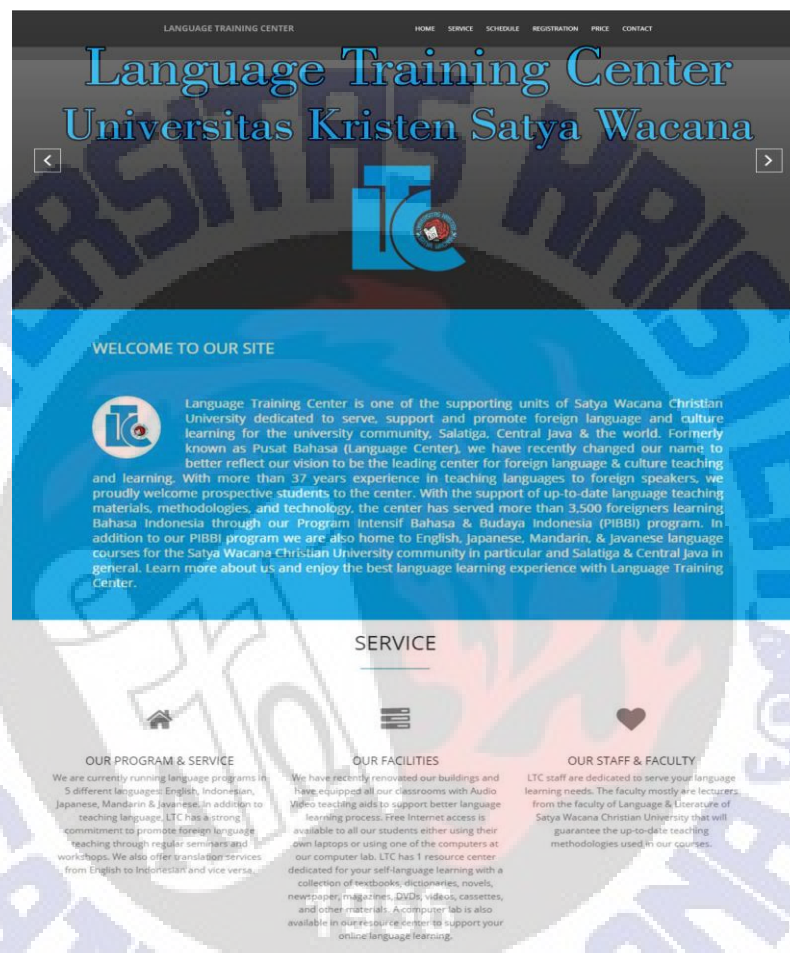


Gambar 7 Class Diagram Aplikasi

Gambar 7 merupakan *class diagram* yang menunjukkan terdiri dari 3 bagian yaitu *entity*, *controller*, dan *view*. Aplikasi memiliki 3 *entity* yaitu *customer*, *program* dan *payment*, yang masing masing memiliki *controller* yang bertugas untuk menangani ketiga *entity* tersebut.

4. Hasil dan Pembahasan

Penelitian ini menghasilkan sebuah *website*, yang penjelasannya akan dibahas pada bagian hasil dan pembahasan, menjelaskan tentang hasil dari sistem yang telah dibangun sesuai dengan metode perancangan sistem. Aplikasi *web* yang dibangun dapat melakukan proses pendaftaran peserta, melihat jadwal kursus dan peserta bisa melihat total tagihan. Dari sisi administrator dapat mengelolah data peserta dan data tagihan sesuai program yang dipilih oleh peserta.



Gambar 8 Tampilan Home

Gambar 8 merupakan tampilan halaman beranda *website* yang menggunakan *single page* sehingga *website* hanya memiliki 1 tampilan namun memiliki beberapa *content*. Ada 5 *content* yang dimiliki oleh *website* ini, yaitu: *Home*, *Service*, *Schedule*, *Registration*, *Payment Check* dan *Contact*. Pada gambar di atas memperlihatkan tampilan *Home* yang berisikan mengenai penjelasan tentang program yang ada di LTC dan tampilan *Service* yang menjelaskan mengenai layanan dan fasilitas yang diberikan LTC kepada peserta.

SCHEDULE

English Conversation Class			
40 Jam			
Pelaksanaan	Hari	Waktu	Biaya
Februari - April	Selasa & Kamis	16:00 - 17:40	725000
General English			
40 Jam			
Pelaksanaan	Hari	Waktu	Biaya
Februari - April	Rabu & Jumat	16:00 - 17:40	725000
IELTS Preparation			
60 Jam			
Pelaksanaan	Hari	Waktu	Biaya
Februari - April	Senin, Selasa, Kamis & Jumat	14:00 - 15:40	925000
TOEFL Preparation			
60 Jam			
Pelaksanaan	Hari	Waktu	Biaya
Februari - April	Senin, Selasa, Kamis & Jumat	14:00 - 15:40	925000

REGISTRATION

Batch 1: Februari - April
Batch 2: Mei - Juli
Batch 3: Oktober - November

Nama

Gender Male Female

Tanggal lahir

Riwayat

Email

Kontak

Status Pelajar Mahasiswa Umum

Pilihan Program

Batch

Tahun

DAFTAR

Gambar 9 Tampilan *Schedule* dan *Registration*

Gambar 9 merupakan *content* dari *Schedule* yang memperlihatkan jadwal program yang ada di LTC, nama program, durasi, pelaksanaan, hari, waktu dan biaya. Calon peserta bisa mengetahui pelaksanaan kursus dengan melihat tabel jadwal yang sudah tersedia. Dan juga *content* dari *Registration* yang menampilkan *form* pendaftaran untuk calon peserta. Untuk melakukan pendaftaran, calon peserta wajib mengisi *form* pendaftaran sesuai yang ada pada tampilan *Registration*.

Kirim

No	Name	Gender	Birth	Address	Email	Kontak	Status	Program	Batch	Year	Edit	Payment	Delete
1	Aviv Isaedy	male	13 Maret 1967	Jl Somopuro Kidul no 4A, Salatiga	avisnaedy@yahoo.com	0817250149	umum	English Conversation Class	1	2016	Edit	Payment	Delete
2	Florentina Elyonova	female	12 Desember 1998	Jl. Merdeka Selatan III No. 18	elyonovaflorentina@yahoo.com	082243369280	pelajar	English Conversation Class	1	2016	Edit	Payment	Delete
3	Christa Rahmita	female	6 September 1994	Turen	christa@gmail.com	085656214286	mahasiswa	Korea	1	2017	Edit	Payment	Delete
4	HANS HANANTO	male	18 Maret 1986	JL DIPONEGORO 37 SALATIGA	hananto.andreas@staff.uksw.edu	081329167550	umum	Jepang	1	2017	Edit	Payment	Delete
5	Paulina Putri Issabella	female	31 Agustus 1996	Jalan Kemiri Raya nomor 7	paulina.issabella@gmail.com	082137739625	mahasiswa	Jepang	1	2017	Edit	Payment	Delete
6	Kevin	male	13 September 1994	Bitung	kevin@gmail.com	085656214286	umum	TOEFL Preparation	1	2017	Edit	Payment	Delete
7	Amelinda Zabrina Iko Putri	female	9 Juli 1996	Jl. Pinang raya AS 51 no. 13 kranggan permai - Jat	poyo.lucu@gmail.com	087886105247	mahasiswa	Jepang	1	2017	Edit	Payment	Delete
8	Shally Arvani Letuna	female	31 Agustus 1997	Jl. Kemiri Sari No.4, Salatiga	232015313@student.uksw.edu	085294570980	mahasiswa	General English	1	2017	Edit	Payment	Delete
9	Ardi Pratama Patencko	male	25 Juni 1993	Kemiri 2 no 799, salatiga	672014603@student.uksw.edu	085242312512	umum	General English	1	2017	Edit	Payment	Delete

© 2016 Language Training Center - All Rights Reserved

Gambar 10 Tampilan Data peserta

Tampilan menu administrator untuk mengelola data peserta, seperti ditunjukkan pada gambar 10. Bagian data peserta, administrator dapat mengelola data peserta seperti melihat data peserta, menghapus dan mengubah data peserta, dan memasukkan data peserta yang baru mendaftar. Administrator juga dapat melihat data sesuai dengan program yang diambil oleh peserta. Apabila data sudah banyak, administrator bisa menggunakan fitur *search by name* untuk menemukan data peserta yang dimaksud.

Kode Program 1 Source Code untuk menampilkan data peserta

```

1. public function showall()
2. {
3.     $user = customer::all();
4.     return view("admin/viewall", compact('user'));
5. }
6.
7. public function GE()
8. {
9.     $user = DB::table('customers')->where('program', '=', 'General English')->get();
10.    return view("admin/GE", compact('user'));
11. }
12.
13. public function ECC()
14. {
15.    $user = DB::table('customers')->where('program', '=', 'English Conversation Class')->
    >get();
16.    return view("admin/ECC", compact('user'));
17. }

```

Kode Program 1 merupakan *source code* untuk menampilkan data peserta yang sudah melakukan pendaftaran. Potongan *source code* diatas terbagi dalam 3 *class*, pertama *class showall* untuk menampilkan semua data peserta yang telah mendaftar, baris pertama adalah nama fungsinya, baris ke 3 pendeklarasikan nama model *costumer* untuk mengambil semua data dalam tabel dan baris 4 untuk kembali ke tampilan admin/viewall. *Class GE* untuk menampilkan data perserta yang mendaftar program *General English*, baris ke 9 pendeklarasikan untuk mengambil data pada tabel *customers*, data yang di ambil hanya data peserta yang mendaftar program *General English*, baris ke 10 untuk kembali ke tampilan admin/GE. *Class ECC* untuk menampilckkan data peserta yang mendaftar program *English Conversation Class*, baris ke 15 pendeklarasikan untuk mengambil data pada tabel *customers*, data yang di ambil hanya data peserta yang mendaftar program *English Conversation Class*, baris ke 10 untuk kembali ke tampilan admin/ECC.

Kode Program 2 Source Code untuk membuat *table payments* menggunakan fitur *migration* Laravel

```

1. public function up()
2. {
3.     Schema::create('payments', function (Blueprint $table) {
4.         $table->increments('id_payment');
5.         $table->char('payment1',25);
6.         $table->char('date1',50);
7.         $table->char('payment2',25)->nullable();
8.         $table->char('date2',50)->nullable();
9.         $table->char('payment3',25)->nullable();
10.        $table->char('date3',50)->nullable();
11.        $table->integer('total');
12.        $table->integer('id_customers')->unsigned();
13.        $table->timestamps();
14.        $table->foreign('id_customers')->references('id')->on('customers')
15.        ->onUpdate('cascade')->onDelete('cascade');
16.    }); }

```

Kode Program 2 merupakan *source code* untuk membuat *database* menggunakan fitur *migration* Laravel. *Migration* mempermudah *programer* untuk membuat *table* sehingga tidak perlu lagi *create table* melalui aplikasi *server*. Membuat *table* melalui *command prompt* dengan menjalankan perintah *php artisan make:migration create_table_payment – create=payments* secara otomatis akan tercipta sebuah *file*. Di dalam *file* tersebut *programer* bisa menuliskan *source code* seperti pada gambar di atas untuk membentuk struktur *table* yang ingin dibuat. Fitur ini juga mempermudah *programmer* apabila program dikerjakan oleh beberapa orang dan menggunakan lebih dari 1 komputer, hanya dengan menjalankan perintah *php artisan migrate* di *command prompt* maka dengan otomatis *table* yang dibuat akan langsung tercipta pada *database*. Melalui *migration* juga *programer* bisa melakukan *update* dan *delete* kolom atau baris pada *table* melalui *file* yang telah dibuat, dengan mengubah sesuai kebutuhan. Baris ke 3 perintah untuk membuat tabel *payments* dengan stuktur sebagai berikut, baris ke 4 sampai ke 11 pendeklarasikan tipe data, nama kolom, dan panjang karakter yang dapat di tampung kolom tersebut. Baris ke 12 adalah kunci tamu dalam kolom tersebut. Baris ke 14 dan 15 pendeklarasikan *id_costumers* adalah kunci tamu dan nilainya di ambil dari kolom *id* yang ada pada tabel *customers*.

Kode Program 3 Source Code untuk menambuat route pada Laravel

```
1. Route::get('/insert', 'Testing@index');
2. Route::post('/store', 'Testing@store');
3. Route::get('/showall', 'Testing@showall');
4. Route::get('/customer/{name}', 'Testing@show');
5. Route::get('/edit/{name}', 'Testing@edit');
6. Route::patch('edit/update/{id}', 'Testing@update');
7. Route::get('delete/{id}', 'Testing@destroy');
8. Route::get('/payment/{name}', 'Testing@payment')
```

Kode Program 3 merupakan *source code* untuk fitur *route* Laravel. Laravel memiliki file tersendiri untuk mengatur *route*, dalam laravel fungsi routing yaitu menangani *request* kemudian mengarahkan aplikasi untuk memanggil halaman/*resource* tertentu yang biasanya dalam bentuk *URL*. Baris ke 1 sampai 8 fungsinya sama untuk menangani *request* kemudian akan diarahkan ke sebuah fungsi yang ada *controller*. Sebagai contoh */insert* adalah nama *URL* yang akan di akses, *testing@index* untuk *testing* adalah nama *controller* dan *index* adalah nama fungsi yang akan di akses pada *controller testing*.

No	Name	Payment 1	Date 2	Payment 2	Date 2	Payment 3	Date 3	Bill	Pays off	Remaining Payment	Update	Delete
1	Titis Maulidiyati	Rp 250000	18 Maret 2017	Rp 300000	24 Maret 2017	Rp		Rp 725000	Rp 550000	Rp -175000	Update	Delete
2	Joshua Marthen Manuputty	Rp 250000	15 Maret 2017	Rp 250000	25 Maret 2017	Rp 10 April 2017	225000	Rp 725000	Rp 500010	Rp -224990	Update	Delete
3	Shyllia dea Wattimena	Rp 200000	12 Maret 2017	Rp 500000	29 Maret 2017	Rp		Rp 925000	Rp 700000	Rp -225000	Update	Delete
4	Gama Erika Cendekia	Rp 400000	20 Maret 2017	Rp 400000	26 Maret 2017	Rp 125000	1 April 2017	Rp 925000	Rp 925000	Rp 0	Update	Delete
5	Mesti fajar Romadhoni	Rp 300000	17 Maret 2017	Rp 100000	27 Maret 2017	Rp		Rp 725000	Rp 400000	Rp -325000	Update	Delete
6	Anugerah Dwi Putra Kiding	Rp 250000	4 Maret 2017	Rp 250000	21 Maret 2017	Rp 225000	8 April 2017	Rp 725000	Rp 725000	Rp 0	Update	Delete
7	Lydia Evangelin Driyanto	Rp 500000	23 Maret 2017	Rp 250000	25 Maret 2017	Rp 175000	04 April 2017	Rp 925000	Rp 925000	Rp 0	Update	Delete
8	Pascal Palit	Rp 50000	2 April 2017	Rp 100000	7 April 2017	Rp		Rp 725000	Rp 150000	Rp -575000	Update	Delete
9	Amelinda Zabrina Iko Putri	Rp 50000	04 Maret 2017	Rp		Rp		Rp 725000	Rp 50000	Rp -675000	Update	Delete
10	Kevin	Rp 200000	23 Maret 2017	Rp 50000	27 Maret 2017	Rp		Rp 925000	Rp 250000	Rp -675000	Update	Delete
Total									Rp 6200010	Rp -3299990		

© 2016 Language Training Center. All Rights Reserved

Gambar 11 Tampilan Data *Payment*

Tampilan menu administrator untuk mengelola data *payment*, seperti ditunjukkan pada gambar 11. Bagian data *payment*, administrator dapat mengelola data *payment* seperti melihat data *payment*, menghapus dan mengubah data *payment*. Administrator juga dapat melihat data *payment* sesuai dengan program yang diambil oleh peserta. Apabila data sudah banyak, administrator bisa menggunakan fitur *search by name* untuk menemukan data *payment* yang dimaksud.

Send Message

Name: a
Aviv Isnaedy

Recipient Number:

Message:

- Fiorentina Elyonova
- Christa Rahmita
- HANS HANANTO
- Paulina Putri Issabella
- Amelinda Zabrina Iko Putri
- Shally Arvani Letuna
- Ardi Pratama Patengko
- Berkelina Ayu Permatasari
- Nobsianus Rigenes Sampe Asang

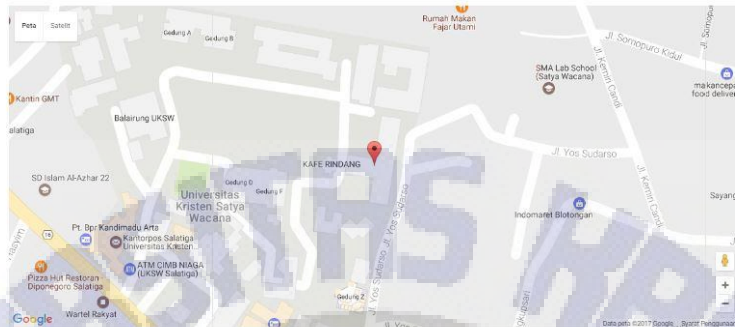
Gambar 12 Tampilan SMS Gateway

Tampilan *form send message* seperti yang ditunjukkan gambar 12, *form* ini berfungsi untuk mengirimkan *sms* ke peserta dengan mengisi nama, nomor tujuan dan pesan. Administrator memasukkan nama peserta secara otomatis akan menampilkan nama-nama peserta sesuai huruf atau nama yang di masukkan. Data tersebut diambil dari *database customer* sehingga menampilkan nama peserta yang sudah terdaftar. Saat nama di pilih secara otomatis nomor telepon peserta akan langsung terisi di *field recipient number*.

CONTACT

✍ Language Training Center.
PO Box 50711
Jl. Diponegoro 52-60
Gedung Perpustakaan UKSW, Lantai 6
Indonesia

☎ Phone: (0298) 313-859
✉ ltc@adm.uksw.edu



Gambar 11 Tampilan Map

Tampilan kontak seperti yang ditunjukkan gambar 11, lokasi LTC ditampilkan dalam bentuk peta menggunakan *Google Maps API*, penggunaan *map* ini bertujuan untuk mempermudah para calon peserta terlebih khusus yang berasal dari luar kota Salatiga. Selain itu tampilan kontak juga menampilkan nomor telepon dan alamat *e-mail* sehingga calon peserta dapat menghubungi LTC untuk informasi yang lebih lengkap.

Kode Program 4 Script untuk menampilkan peta menggunakan *Google Maps API*

```
1. <div class="col-md-7 contact-form wow animated fadeInLeft">
2. <div class="container">
3. <div id="map" style="width:100%;height:500px"></div>
4. <div id="map"></div>
5. <script>
6.     var marker;
7.     function initMap() {
8.         var uluru = {lat: -7.3188999, lng: 110.5004685};
9.         var map = new google.maps.Map(document.getElementById('map'), {
10.             zoom: 18,
11.             center: uluru
12.         });
13.         marker = new google.maps.Marker({
14.             map: map,
15.             draggable: true,
16.             animation: google.maps.Animation.DROP,
17.             position: {lat: -7.3188999, lng: 110.5004685},
18.             title:"Language Training Center(LTC) 6th Floor"
19.         });
20.         marker.addListener('click', toggleBounce);
21.     }
22. </script>
23. <script async defer
24.     src="https://maps.googleapis.com/maps/api/js?key=AIzaSyAIFIHV6iYLHgrkd0D9-b3EjoT7z-
25.     Wy_c&callback=initMap">
26. </script>
27. </div>
28. </div>
```

Kode Program 4 merupakan *script* untuk menampilkan peta pada halaman aplikasi. Peta ditampilkan dengan cara mengambil titik koordinat yang di deklarasikan pada *uluru*, peta akan menampilkan koordinat tepat di bagian tengah dan marker digunakan untuk memberi penanda dimana lokasi berada.

Berdasarkan hasil perancangan aplikasi yang dibuat, dilakukan pengujian menggunakan *black box testing*, untuk melihat apakah semua fungsi sudah dapat berjalan sebagaimana mestinya. Hasil pengujian *black box* seperti pada Tabel 1.

Tabel 1 Hasil Pengujian *Black Box*

Fungsi yang diuji	Kondisi	Hasil yang diharapkan	Hasil yang diperoleh	Hasil
<i>Login administrator</i>	<i>User & password</i> benar	Masuk menu administrator	Masuk menu administrator	Sesuai
	<i>User & password</i> salah	Tidak masuk menu administrator	Tidak masuk menu administrator	Sesuai
<i>Insert data peserta</i>	Data peserta sesuai (<i>name, gender, birth, phone number, status, program, batch, year</i>)	Data peserta berhasil ditambahkan	Data peserta berhasil ditambahkan	Sesuai
	Data peserta tidak sesuai (<i>name, gender, birth, phone number, status, program, batch, year</i>)	Data peserta gagal ditambahkan	Data gagal ditambahkan	Sesuai
<i>Insert data payment</i>	Data <i>payment</i> sesuai (<i>payment, date, cost, id peserta</i>)	Data <i>payment</i> berhasil ditambahkan	Data <i>payment</i> berhasil ditambahkan	Sesuai
	Data <i>payment</i> tidak sesuai (<i>payment, date, cost, id peserta</i>)	Data <i>payment</i> gagal ditambahkan	Data <i>payment</i> gagal ditambahkan	Sesuai
<i>Insert data program kursus</i>	Data program kursus sesuai (<i>nama program, durasi, pelaksanaan, hari, waktu, biaya</i>)	Data program kursus berhasil ditambahkan	Data program kursus berhasil ditambahkan	Sesuai
	Data <i>teacher</i> tidak sesuai (<i>nama program, durasi, pelaksanaan, hari, waktu, biaya</i>)	Data program kursus gagal ditambahkan	Data program kursus gagal ditambahkan	Sesuai
<i>SMS Gateway</i>	Data sesuai (<i>Name, phone number, message</i>)	Pesan berhasil dikirimkan	Pesan berhasil dikirimkan	Sesuai
	Data tidak	Pesan gagal	Pesan gagal	Sesuai

sesuai (*Name, phone number, message*) dikirimkan dikirimkan

Berdasarkan Tabel 1, dapat dilihat pengujian menunjukkan bahwa semua fungsi dapat berjalan dengan semestinya dapat disimpulkan bahwa aplikasi ini sudah dapat digunakan dengan baik.

Selain itu dilakukan wawancara kepada pengguna, dalam hal ini administrator yang nantinya akan mengoperasikan aplikasi ini yaitu *staff* keuangan, *receptionist*, *manager*. Hasil wawancara dengan administrator dapat dilihat pada Tabel 2.

Tabel 2 Hasil Wawancara

Pertanyaan	<i>Staff</i> Keuangan	<i>Receptionist</i>	<i>Manager</i>
Dari skala 1 – 10, berapa nilai untuk tampilan <i>website</i> ? bagaimana pendapat anda mengenai tampilan?	8, sangat ringkas. <i>Background</i> sebaiknya menggunakan gambar.	6, <i>Map</i> sangat bagus. menu pada panel <i>admin</i> mudah di akses.	7, di beri penjelasan yang lebih jelas mengenai <i>form</i> pendaftaran. menu pada panel <i>admin</i> mudah di akses.
Saran apa saja yang dapat diberikan untuk pengembangan tampilan <i>website</i> kedepannya?	Tampilan <i>user</i> dibuat lebih baik lagi dengan menambahkan gambar sebagai <i>background</i> ,	tulisan kontak di ganti dengan nomor telepon.	Untuk <i>admin</i> sudah bagus, untuk tampilan <i>user</i> diberikan penjelasan untuk setiap bagian, seperti jadwal, <i>form</i> pendaftaran dan cek pembayaran.
Apakah menu - menu pada <i>website</i> mudah untuk di akses?	Ya, menu – menu pada <i>website</i> mudah di akses.	Ya, menu – menu mudah untuk di gunakan.	Ya, menu – menu pada <i>website</i> mudah di akses.
Apakah fungsi - fungsi dalam program sudah dapat dijalankan dengan baik?	Ya	Ya	Ya
Apakah dengan adanya <i>website</i> ini dapat bermanfaat dan memudahkan proses manajemen data peserta?	Ya, bermanfaat untuk proses manajemen data keuangan.	Ya, mempermudah untuk proses manajemen data peserta.	Ya, mempermudah penyimpanan data peserta
Dari skala 1 – 10, berapa nilai untuk <i>website</i> ini secara keseluruhan? Apa saja tanggapan anda mengenai keseluruhan <i>website</i> ini?	8, Karena <i>website</i> mudah digunakan dan fungsi – fungsi pada <i>website</i> sudah berjalan dengan baik.	8, fungsi mudah di gunakan dan tampilan sangat menarik.	7, sudah baik di gunakan, tampilan <i>user</i> di buat lebih menarik
Saran apa saja yang dapat anda berikan untuk pengembangan <i>website</i> ini kedepannya?	Menambah menu yang bisa melihat nilai dan <i>presensi</i> peserta.	Menambah fitur <i>sort by level</i> , sehingga peserta dapat dibedakan berdasarkan level.	Fitur untuk peserta ditambah seperti pemberitahuan ke peserta batas pembayaran secara otomatis

Berdasarkan wawancara yang telah dilakukan diperoleh hasil bahwa aplikasi mudah digunakan, menu – menu pada aplikasi mudah diakses dan semua fungsi sudah berjalan dengan baik. Dengan adanya aplikasi ini juga mempermudah administrator dalam proses manajemen data peserta, data tagihan dan mempermudah memberikan informasi ke peserta. Akan tetapi memiliki kekurangan dari segi pengolahan data peserta yang belum bisa di bedakan berdasarkan level untuk setiap program.

5. Simpulan

Pembuatan aplikasi ini menggunakan *framework* Laravel dapat dengan mudah dilakukan karena dalam pengembangannya menggunakan pendekatan *Model-View-Controller*, dimana *source code* fungsi hubungan antara *database* dan logika terpisah dengan *source code* tampilan sehingga penulisan kode program teratur dengan baik dan terstruktur

Berbagai fitur *framework* Laravel yang digunakan dalam pengembangan aplikasi ini dapat mempermudah proses pengembangan aplikasi, fitur lengkap bahkan Laravel mempunyai banyak fitur yang belum dimiliki oleh *framework* lainnya seperti *blade templating*, *migration*, *eloquent ORM* dan lain lain dengan *coding* yang lebih *simple* dan mudah dimengerti memungkinkan pengembangan yang cepat dan efisien.

Adanya aplikasi pendaftaran dapat mempermudah peserta untuk memperoleh informasi mengenai pendaftaran, jadwal, dan tagihan, serta membantu administrator dalam mengelola data peserta, data pembayaran, dan mempermudah dalam memberikan informasi mengenai jadwal kepada peserta dan calon peserta.



6. Daftar Pustaka

- [1] Devita; F, Dico; Kartini. 2013. “Sistem Informasi Manajemen Pemasaran Produk pada PT. Autochem Industry berbasis Web”. Palembang: Jurusan Sistem Informasi, STIMIK PalComTech.
- [2] Ariefin, Muhammad Rizki Samsul; Ilhamsyah. 2015. “Membangun *Website* Sistem Informasi Sekolah dengan menggunakan *Framework PHP* Laravel pada SMA Negeri 4 Pontianak”. Pontianak: Bidang Teknologi Informasi dan Multi Disiplin Universitas Tanjungpura.
- [3] Hidayatullah, Ahmad; Ariyus, Dony. 2015. “Analisi dan Perancangan toko online Geraihana berbasis SEO menggunakan Laravel”. Yogyakarta: Jurusan Teknik Informatika, Sekolah Tinggi Manajemen Informatika dan Komputer, Amikom Yogyakarta.
- [4] Satana, Jaka; Bahtiar, Nurdin; Wibawa, Helmie Arief. 2013. “Sistem Informasi Akademik Lembaga Kursus dan Pelatihan Berbasis *Web*”. Saemarang: Jurusan Matematika, Fakultas Sains dan Matematika, Universitas Diponegoro.
- [5] Steviana, Grace. 2014. “Perancangan *Website* Gereja Pekabaran Injil ‘Jalan Suci’ Ambon”. Salatiga: Program Studi Teknik Informatika, Universitas Kristen Satya Wacana.
- [6] Aminudin. 2015 “Cara Efektif Belajar *Framework Laravel*”. Yogyakarta: CV. Lokomedia.
- [7] Nanlohy, Max William. 2012 “Perancangan Sistem Absensi Siswa berbasis *SMS Gateway*”. Salatiga: Program Studi Sistem Informasi, Universitas Kristen Satya Wacana.
- [8] Priago, Ricad Neton. 2013 “Pengelolaan Lembaga Kursus (LPK) Chinju dalam Menyelenggarakan Kursus Bahasa Korea di Kota Bengkulu”. Bengkulu: Program Studi Pendidikan Luar Sekolah, Jurusan Ilmu Pendidikan, Universitas Bengkulu.