BAB I

PENDAHULUAN

A. Latar Belakang

Seiring dengan perkembangan zaman, komputer bukan lagi menjadi kebutuhan yang biasa, maka kebutuhan untuk memaksimalkan penggunaan komputer semakin meningkat. Hampir dari skala kecil sampai dengan skala besar mulai menggunakan komputer untuk menunjang kegiatan operasionalnya. Namun, seringkali mengabaikan atau kurang memaksimalkan pemakaian komputer di dalam kegiatan operasional.

Pertumbuhan ekonomi yang menuju kearah yang lebih baik diikuti oleh pertumbuhan pembangunan yang signifikan. Hal ini membutuhkan pengawasan dari pihak pemerintah untuk menjaga ketertiban dan kenyamanan masyarakat disekitarnya. Dalam pengelolaan pengawasan tersebut maka pemerintah daerah mewajibkan masyarakatnya untuk mendapatkan perizinan dan perizinan dalam berbagai bidang pada Penyelenggaraan Pelayanan Perizinan Terpadu Satu Pintu. Penyelenggaraan Pelayanan Perizinan Terpadu Satu Pintu adalah kegiatan penyelenggaraan pelayanan perizinan dan non perizinan yang dilakukan masyarakat pada satu tempat. Tujuan dari penyelenggaraan Pelayanan Perizinan dan Non Perizinan adalah meningkatkan kualitas publik, memberikan akses yang lebih luas kepada masyarakat untuk memperoleh pelayanan publik.

Berdasarkan latar belakang diatas, maka dari itu penulis akan mengangkat judul "**Perancangan Aplikasi Perizinan Lembaga Pelatihan Kerja**".

B. Rumusan Masalah

Berdasarkan latar belakang diatas maka dirumuskan masalah yaitu Rancang Perancangan Aplikasi Perizinan Lembaga Pelatihan Kerja Kota Parepare?

C. Batasan Masalah

Adapun batasan masalah dalam penelitian ini adalah :

1. Studi kasus penelitian dilakukan kantor perzinan kota Parepare
2. Aplikasi ini menggunakan Bahasa pemograman PHP

D. Tujuan Penelitian

Tujuan dari penelitian ini adalah Aplikasi Perizinan Lembaga Pelatihan Kerja Kota Parepare Dalam penyusunan Tugas Akhir ini penulis mengadakan penelitian dengan mengumpulkan data-data yang berhubungan dengan masalah formular pendaftaran pemohon, informasi persyaratan dalam mengurus perizinan dan non perizinan, laporan data pemohon dan penyimpanan data pemohon perizinan dan non perizinan pada Dinas Penanaman Modal dan Pelayanan Terpadu pada kota parepare

E. Manfaat Penelitian

Adapun manfaat penelitian yaitu sebagai berikut :

1. Aspek Akademis

a. Sebagai bahan masukan bagi para pengkaji ilmu komunikasi di bidang *Public Relations* yang berminat untuk meneliti masalah yang sama.
b. Sebagai langkah awal bagi peneliti untuk dapat mengembangkan teori-teori yang didapat selama ini dan menambah Ilmu khususnya tentang Aplikasi perizinan *online* berbasis *Website*.

2. Aspek Praktis
   a. Sebagai masukan dan bahan informasi bagi dunia Pendidikan Formal yang berada di kota Parepare.
   b. Bagi peneliti diharapkan dapat memberikan pengalaman dan menerapkan ilmu pengetahuan yang diterima selama mengikuti perkuliahan.
   c. Sebagai tugas akhir untuk mendapatkan gelar sarjana strata 1 (S.1) pada Fakultas Teknik Informatika

BAB II

TINJAUAN PUSTAKA

A. Tinjuan Teori

1. Pengertian Perancangan

Perancangan merupakan salah satu hal yang penting dalam membuat program. Adapun 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. Menurut *Pre*ssman perancangan atau rancang merupakan serangkaian prosedur untuk menterjemahkan hasil analisa dan sebuah sistem ke dalam bahasa pemrograman untuk mendeskripsikan dengan detail bagaimana komponen-komponen sistem di implementasikan.

2. Aplikasi

Aplikasi berasal dari kata *application* yaitu bentuk benda dari kata kerja *to apply* yang dalam Bahasa Indonesia berarti pengolah. Secara istilah, aplikasi komputer adalah suatu sub kelas perangkat lunak komputer yang menggunakan komputer langsung untuk melakukan suatu tugas yang

diinginkan pemakai. Menurut Kamus Besar Bahasa Indonesia (KBBI, 2015) Aplikasi adalah penerapan dari rancang sistem untuk mengolah data yang menggunakan aturan atau ketentuan bahasa pemrograman tertentu. Aplikasi adalah suatu program komputer yang dibuat untuk mengerjakan dan melaksanakan tugas khusus dari pengguna. Sementara menurut Hartono (1999) aplikasi adalah penggunaan dalam suatu komputer, insturksi (*instruction*) atau pernyataan (*statement*) yang disusun sedemikian rupa sehingga komputer dapat memproses input menjadi output.

3. Monitoring

Menurut Dr. Harry Hikmat (2010), *Monitoring* adalah proses pengumpulan dan analisis informasi berdasarkan indikator yang ditetapkan secara sistematis dan berkelanjutan tentang kegiatan/program sehingga dapat dilakukan tindakan koreksi untuk penyempurnaan program/kegiatan itu selanjutnya. *Monitoring* adalah pemantauan yang dapat dijelaskan sebagai kesadaran tentang apa yang ingin diketahui, pemantauan berkadar tingkat tinggi dilakukan agar dapat membuat pengukuran melalui waktu yang menunjukkan pergerakan ke arah tujuan atau menjauh dari itu. *Monitoring* akan memberikan informasi tentang status dan kecenderungan bahwa pengukuran dan evaluasi yang diselesaikan berulang dari waktu kewaktu, pemantauan umumnya dilakukan untuk tujuan tertentu, untuk memeriksa terhadap proses berikut objek atau untuk mengevaluasi kondisi atau kemajuan menuju tujuan hasil manajemen atas efek tindakan dari beberapa jenis antara lain tindakan untuk mempertahankan manajemen yang sedangberjalan. *Monitoring* adalah proses rutin pengumpulan

data dan pengukuran kemajuan atas objektif program. Memantau perubahan yang fokus pada proses dan keluaran

B. Sekilas Tentang Aplikasi Yang Digunakan

1. Web

Web merupakan media informasi berbasis jaringan komputer yang dapat diakses dimana saja dengan biaya relatif murah. *Web* merupakan bentuk implementasi dari bahasa pemrograman *Web (web programing).* Sejarah perkembangan bahasa pemrograman *Web* diawali dengan munculnya HTML *(Hypertaxt Markup Language)*, yang kemudian dikembangkan dengan munculnya CSS (*Cascading Style Sheet*) yang bertujuan untuk memperindah tampilan *website*.

Bahasa pemrograman saat ini sudah sangat berkembang dengan berbagai kemudahan dalam penyajian dan *interface* yang lebih *friendly used*. Penyajian yang baik dari suatu Bahasa pemrograman akan menghasilkan sebuah *Web* yang dinamis sehingga pengunjung akan lebih mudah mendapatkan informasi yang dibutuhkan. Beberapa bahasa pemrograman *Web* berbasis *server* (*Server Side*) mulai muncul dan dikembangkan oleh beberapa perusahaan perangkat lunak, seperti : ASP (*Active Server Pages*) oleh *Microsoft*, JPS (*Java Server Pages*) oleh *Sun Microsystem*, dan CGI (*Common getway Interface*) oleh perusahaan UNIX yang sekarang diadopsi oleh NCSA (*National Central For Supercomputing Aplication*) *Web server*. (Alexander F. K, Sibero. 2011)

2. Internet (*Interconneted Network*)

Internet adalah sekumpulan jaringan komputer yang saling terhubung secara fisik dan memiliki kemampuan untuk membaca dan menguraikan protokol komunikasi tertentu yang disebut *Internet Protocol (IP)* dan *Transmission Control Protocol (TCP)*. *Protocol* adalah *Spesifikasi* sederhana mengenai bagaiman komputer saling bertukar informasi. (Alan Nur Aditya, 2011)

3. WWW (*World Wide Web*)

World Wide Web atau WWW atau juga dikenal dengan *WEB* adalah suatu ruang informasi yang dipakai oleh pengenal global yang disebut pengidentifikasi sumber serangam untuk mengenal pasti sumber daya berguna. WWW sering dianggap sama dengan internet.

WWW merupakan kempulan *Web* dari seluruh dunia yang mempunyai kegunaan untuk menyediakan data dan informasi untuk dapat digunakan bersama. WWW adalah bagian yang paling menarik dari internet. Melalui *Web*, para pengguna dapat mengakses informasi- informasi yang tidak hanya berupa teks tetapi bisa juga berupa gambar, suara, video dan animasi.WWW adalah suatu program yang ditemukan oleh Tim burners-Lee pada tahun 1991. Awalnya Berners Lee hanya ingin menemukan cara untuk menyusun arsip-arsip risetnya. Dengan program itu menciptakan jaringan yang manautkan berbagai arsip sehingga memudahkan pencairan informasi yang dibutuhkan. Inilah yang kelak menjadi dasar sebuah perkembangan pesat yang dikenal sebagai WWW. (Ramadhan Arif. 2006)

4. HTTP (*Hypertext Tranfer Protocol*)

HTTP *(Hypertext Tranfer Protocol)* adalah *Protocol* yang dipergunakan untuk mentransfer dokumen dalam *World Wide Web* (WWW). *protocol* ini adalah *protocol* ringan, tidak berstatus dan generik yang dapat dipergunakan berbagai macam tipe dokumen. Pengembangan HTTP dikoordinasikan oleh Konsorsium *World Wide Web* (W3C) dan grup bekerja *Internet Engineering Task Force* (IETF), bekerja dalam publikasi satu seri RFC, yang paling terkenal RFC yang menjelaskan HTTP/1,1, versi HTTP yang digunakan umum sekarang ini. HTTP adalah sebuah *protocol* meminta / menjawab antara *client* dan *server*.

Sebuah client HTTP seperti *Web browser*, biasanya melalui permintaan dengan membuat hubungan TPC/IP ke *port* tertentu. Sebuah *server* HTTP yang mendengarkan di*port* tersebut menunggu *client* mengirim kode permintaan di ikuti dengan pesan MIME yang memiliki beberapa informasi data tertentu. (Deka M Wildan, 2014).

5. URL (*Uniform Resource Locator*)

URL *(Uniform Resource Locator)* merupakan suatu *pathname* untuk mengidentifikasikan sebuah dokumen di *Web*. Dibandingkan dengan *logocal pathname* yang membuat nama direktori dan subdirektori yang menunjuk ke satu *file* di sebuah komputer, cakupan URL lebih luas lagi. Didalam URL dimuat lokasi atau nama mesin komputer yang diminta, nama *file* dari dokumen itu (termasuk *logical pathnamanya*), serta protoko untuk mengaksesnya. (Santoso,Harip. 2010)

6. HTML (*Hypertext Markup Language*)

HTML (*HyperText Markup Language*) Adalah sebuah bahasa markah yang digunakan untuk membuat sebuah halaman *Web*, menampilkan berbagai informasi di dalam sebuah penjelajah *Web* internet dan penformatan hiperteks sederhana yang ditulis dalam berkas format ASCII agar dapat menghasilkan tampilan wujud yang terintegrasi. Dengan kata lain, berkas yang dibuat dalam perangkat lunak pengolah kata dan disimpan dalam format ASCII normal sehingga menjadi halaman *Web* dengan perintah-perintah HTML. Bermula dari sebuah bahasa yang sebelumnya banyak yang digunakan di dunia penerbitan dan pencetakan yang di sebut dengan SGML (*Standard Generalized Markup Language*), HTML adalah sebuah standar yang digunakan secara luas untuk menampilkan halaman *Web*. HTML saat ini merupakan standar internet yang didefenisikan dan dikendalikan penggunanya oleh *World Wide Web Concortium* (W3C).HTML dibuat oleh kolaborasi Caillau TIM dengan Berners-Lee Robert ketika mereka bekerja di CERN pada tahun 1989 (CERN adalah lembaga penelitian fisika energi tinggi di Jenewa). (Santoso,Harip. 2010)

HTML juga bisa berupa dasar atau tulang punggung dalam pembuatan *Web*. Dalam setiap *file* HTML akan selalu di dahului dengan tag (HTML). Tag ini akan membuat *Web* browser mencoba untuk membaca *file* yang ada didalam tag tersebut untuk menampilkan dalam *Web browser*. (Santoso,Harip. 2010)

7. PHP (*Presonal Home Page*)

PHP *(Presonal Home Page)* merupakan bahasa pemrograman berbasis *Web* yang memiliki kemampuan untuk memproses dan mengolah data secara dinamis. PHP dapat dikatakan sebagai sebuah *server-side embedded script language*, artinya semua sintaks dan perintah program yang ditulis akan sepenuhnya dijalankan oleh *server*, tetapi dapat disertakan pada halaman HTML biasa. Pada umumnya, semua aplikasi yang dibangun menggunakan PHP akan memberikan hasil pada *Web browser*, tetapi prosesnya secara keseluruhan dijalankan pada *server*. Pada prinsipnya, *server* akan bekerja apabila ada permintaan dari client. Dalam hal ini, client menggunakan kode-kode PHP akan mengirim perintah ke *server* ketika menggunakan PHP sebagai *serverside embedded script language.* (Alan Nur Aditya, 2011)

8. MySQL

(Sutarman, 2007) *My*SQL merupakan salah satu program yang dapat digunakan sebagai *database*, dan merupakan salah satu *software* untuk *database server* yang banyak digunakan. *MySQL* bersifat *Open Source* dan menggunakan SQL. *MySQL* bisa dijalankan diberbagai flatform misalnya *Windows, linux* dan lain sebagainya.

*My*SQL merupakan salah satu perangkat lunak sistem manajemen basis data (*database management system*) atau DBMS yang menggunakan perintah standar SQL (*Structured Query Language*). Dimana *My*SQL mampu untuk melakukan banyak eksekusi perintah *query* dalam suatu permintaan (*multithread),* baik itu mengirim dan menerima data.*My*SQL juga *multi-user*

dalam arti dapat dipergunakan oleh banyak pengguna dalam waktu bersamaan. Dengan sekitar enam juta Instalasi diseluruh dunia. *My*SQL AB membuat *My*SQL tersedia dalam perangkat lunak gratis dibawah lisensi *GNU General Public Lisence* (GPL) dan juga menjual dalam lisensi komersial untuk ke*Perl*uan jika penggunanya tidak cocok menggunakan lisensi GPL. (Nugroho, Bunafit. 2014)

9. *Adobe Dreamweaver CS6*

Adobe Dreamweaver adalah sebuah HTML *(Hyper Text Markup Language)* editor profesional untuk mendesain *Web* secara visual dan mengelola situs atau halaman *Web*. Saat ini terdapat *software* dari kelompok *Adobe* yang belakangan banyak digunakan untuk mendesain suatu situs *Web*. Versi terbaru dari *adobe dreamweaver* saat ini adalah *Dreamweaver CS6.* Pada *Dreamweaver CS6*, terdapat beberapa kemampuan bukan hanya sebagai *software* untuk mendesain *Web* saja tetapi juga untuk menyunting kode serta pembuatan aplikasi *Web* dengan menggunakan berbagai bahasa pemrograman *Web*, antara lain HTML, PHP, CSS dan sebagainya.

Dreamweaver merupakan *software* utama yang digunakan oleh *Web* desainer maupun *Web* pemrograman dalam mengembangkan suatu situs *Web*. Hal ini disebabkan ruang kerja, fasilitas dan kemampuan *dreamweaver* yang mampu meningkatkan produktivitas dan efektivitas dalam desain maupun membangun suatu situs *Web***.** (*Madcoms Madiun*, 2012)

10. *PhpMyAdmin*

PhpMyAdmin adalah perangkat lunak yang ditulis dalam bahasa pemrograman *PHP* digunakan untuk menangani *Admin*istrasi *MySQL* dengan menggunakan protokol WWW secara langsung dari *browser*. *PhpMyAdmin* mendukung berbagai operasi *MySQL*, diantaranya mengelola basis data, tabel-tabel, bidang *fields*, relasi *relations*, indeks, pengguna *users*, perijinan *permissions*, dan lain-lain. Dengan menggunakan *phpMyAdmin* diharapkan akan memudahkan dalam memanajemen *database MySQL*. (*Madcoms Madiun*, 2012)

11. XAMPP (*Cross-Platfrom (x), Apache, MySQL dan Perl*)

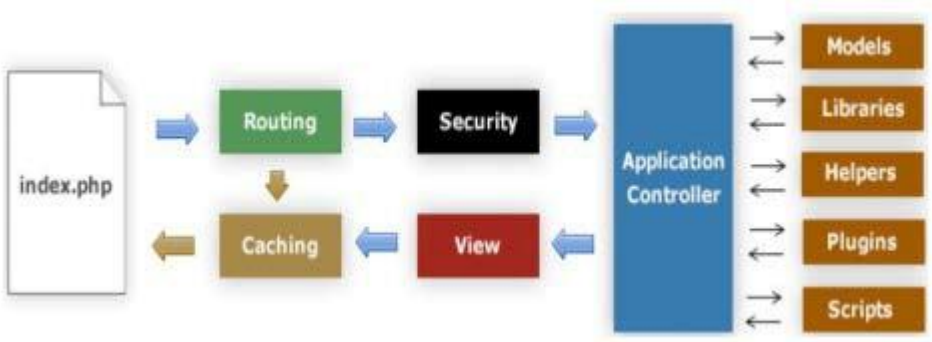
XAMPP adalah perangkat lunak besar, yang mendukung banyak sistem operasi, merupakan kompilasi dari beberapa program. Fungsinya adalah sabagai *server* yang berdiri sendiri (*localhost)*, yang terdiri atas program *Apache* HTTP *server*, *My*Sql *database* dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan *Perl*. Nama XAMPP merupakan singkatan dari X (tempat sistem operasi apapun), *Apache, My*Sql, PHP dan *Perl*. Program ini tersediah dalam GNU *general Public license* dan bebas, merupakan *Web server* yang mudah digunakan yang dapat melayani tampilan halaman *Web* yang dinamis. (*Madcoms Madiun*, 2012)

12. *Framework Codeigniter*

Menurut (*Madcoms Madiun*, 2012) "*CodeIgniter* adalah sebuah *framework* PHP yang dapat membantu mempercepat *developer* dalam pengembangan aplikasi *Web* berbasis PHP dibanding jika menulis semua kode

program dari awal". Didalam CI ini terdapat beberapa macam kelas yang berbentuk *library* dan *helper* yang berfungsi untuk membantu pemrograman dalam mengembangkan aplikasi.*File* dokumentasi ini disertakan secara langsung pada saat Pengunduhan paket *framework CI*.

CodeIgniter menggunakan konsep MVC (*Model View Controller*). Menurut Akhmad Sofwan (2003), konsep MVC adalah konsep pemisahan antara logika dengan tampilan dan *database*. Manfaat dari konsep ini adalah membuat pengodean logika lebih *simple*, karena sudah dipisahkan dengan kode untuk tampilan dan membuat *programmer* dapat bekerja secara terpisah dengan desainer. *Programmer* mengerjakan logika sedangkan desainer berkutat dengan desain dan tampilan. Model (M) berisi kode penghubung *database,View* (V) berisi kode desain tampilan, dan *Controller* (C) berisi kode logika.



Gambar 2.1 *CodeIgniter*

C. Perancangan Basis Data

Menurut (Jogiyanto Hartono, 2005:700). Basis data (*Database*) merupakan kumpulan dari data yang saling berhubungan satu dengan yang lainnya, tersimpan di simpan luar komputer dan digunakan perangkat lunak tertentu

untuk memanipulasi. Sedangkan menurut penadapat saya Basis Data adalah data data yang tersimpan dalan komputer dan masih harus di proses agar saling berhubungan satu sama lain. *Database* merupakan salah satu komponen yang penting di sistem informasi, karena berfungsi sebagai basis penyedia informasi bagi para pemakai. Sistem basis data ini adalah suatu sistem informasi yang mengintegrasikan kumpulan dari data yang saling berhubungan satu dengan lainnya dan membuatnya tersedia untuk beberapa aplikasi yang bermacam-macam di dalam suatu organisasi.

1. Entity Relationship Diagram (ERD)

Menurut Andi, ERD berguna untuk mengorganisasi data-data (*hanya data*) dalam hubungannya dengan data-data yang lain. Sedangkan menurut saya ERD berfungsi untuk mengelompokan data- data agar tejadi keselarasan hubungan antar data lainnya. ERD mempunyai koleksi objek-objek dasar yang dinamakan entitas serta hubungan (*relationship*) antara entitas –entitas itu.

Entitas adalah sesuatu atau objek pada dunia nyata yang dapat dibedakan satu terhadap yang lainnya, yang bermanfaat bagi aplikasi yang sedang kita kembangkan. Sedangkan *Relationship* adalah hubungan antara beberapa entitas.

2. Normalisasi

Menurut Andi, Normalisasi sebagai proses untuk mengubah suatu relasi yang memiliki masalah tertentu ke dalam dua buah relasi atau lebih yang tak

memiliki masalah tersebut. Sedangkan Normalisasi menurut pendapat saya adalah proses pengolahan data yang masih salah dan memiliki data ganda untuk kemudian di perbaiki sehingga menghasilkan relasi-relasi yang lebih baik dari sebelumnya. Normalisasi dapat dipahami sebagai tahapan-tahapan yang masing-masing berhubungan dengan bentuk normal. Bentuk normal adalah keadaan relasi yang dihasilkan dengan menerapkan aturan sederhana yang bekaitan dengan konsep kebergantungan fungsional pada relasi yang besangkutan sebagai berikut.

a. Bentuk normal pertama.

   Bentuk normal pertama adalah suatu bentuk relasi dimana atribut bernilai banyak (*Multivalues attribute*) telah dihilangkan sehingga kita akan menjumpai nilai tunggal (*mungkin saja nilai null*) pada perpotongan setiap baris dan kolom.

b. Bentuk normal kedua.

   Suatu relasi adalah dalam bentuk normalkedu*a* jika dia berada dalam bentuk normal pertama dan setiap atribut bukan kunci bergantung penuh pada kunci primer.

c. Bentuk normal ketiga.

   Relasi adalah dalam bentuk normal ke tiga adalah jika berada dalam bentuk normal kedua dan tidak dijumpai kebergantungan *transitif*. Kebergantungan *transitif* dalam suatu relasi adalah kebergantungan fungsional antara 2 atau lebih atribut bukan kunci.Ketika relasi memilliki lebih dari 1 kunci kandidat, anomali-anomali seperti kita bahas di atas masih mungkin

dijumpai meskipun relasi yang bersangkutan sudah dalam bentuk normal ke tiga.

3. Tabel Relasi

Tabel relasi adalah tabel hubungan antara suatu himpunan entitas dengan entitas yang lainnya.

4. Kunci

Kunci (*key*) merupakan suatu atribut yang unik yang dapat digunakan untuk membedakan suatu entitas dengan entitas yang lainnya dalam suatu himpunan entitas.

D. Tahap-tahap Perancangan Sistem

1. *Use Case*

(Sukamto, Rosa Ariani. 2009) *Use case diagram* adalah abstraksi dari interaksi antara sistem dan *actor*. *Use Case* bekerja mendeskripsikan tipe interaksi antara *user* sebuah sistem dengan sistemnya sendiri melalui sebuah cerita bagaimana sebuah sistem dipakai.

2. Kamus Data

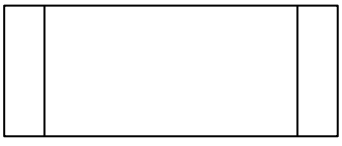
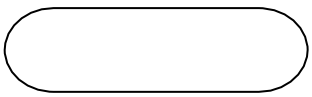
Kamus data (*data dictionary*) adalah katalog fakta tentang data dan kebutuhan informasi yang mengalir pada sistem dengan lengkap dari suatu sistem informasi. Kamus data di buat berdasarkan arus data yang mengalir pada *Data Flow Diagram* (DFD). Kamus data merupakan tempat penyimpanan dari aliran-aliran data, *file-file* dan proses-proses dalam sebuah sistem. Bagian ini menjelaskan

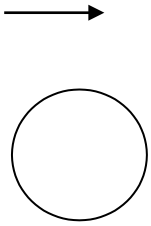
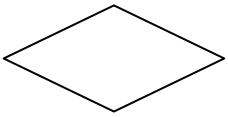
secara detail proses-proses yang terjadi disetiap proses, *file* dan struktur data mengenai model sistem yang digambarkan. Kamus data digunakan untuk menjelaskan semua data yang mengalir atau digunakan dalam sistem, yaitu mengenai arus data yang masuk ke dalam sistem dan tentang informasi yang dibutuhkan oleh pemakai sistem.

3. Bagan Alir Program

Bagan alir program (*program flowchart*) merupakan bagan yang menjelaskan rinci langkah-langkah dari proses program. Bagan alir program dibuat dari derifikasi bagan alir sistem (Bagan alir program dibuat dengan menggunakan symbol-simbol sebagai berikut :

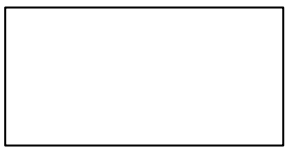
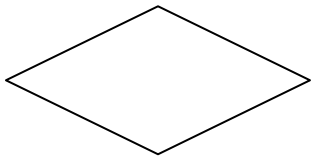
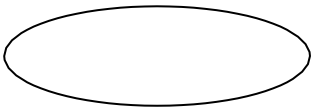
Table 2.1 Simbol *Flowchart*

Simbol	Keterangan
	Simbol *input/output* digunakan untuk mewakili data *input/output*
	Simbol proses, digunakan untuk mewakili suatu proses
	Simbol proses terdefinisi, digunakan untuk menunjukkan suatu operasi yang rinciannya ditunjukkan ditempat lain
	Simbol titik terminal, digunakan untuk awal dan akhir suatu proses

Simbol	Keterangan
	Simbol penghubung, digunakan untuk menunjukkan sambungan dari alir yang terputus dihalaman yang masih sama
	Simbol garis alir, menunjukkan arus dari proses
	Simbol keputusan, digunakan unuk suatu penyelesaian kondisi didalam program

4. Entity Relationship Diagram

Table 2.2 Simbol *Entity Relationship* Diagram

NOTAI	KETERANGAN
	Entitas, yaitu kumpulan dari objek yang di identifikasi secara unik
	Relasi, yaitu hubungan yang terjadi antara satu atau lebih entitas. Jenis hubungan antara lain : satu ke satu, satu ke banyak dan banyak ke banyak
	Atribut, yaitu karakteristik dari *Entity* atau relasi yang merupakan penjalasan detail tentang entitas.

	Garis, hubungan antara *Entity* dengan atributnya dan himpunan entitas dengan himpunan relasi.
	*Input/Output* data, yaitu proses *input/output* data, parameter, informasi.

Entity Relationship Diagram (ERD) adalah adalah suatu pemodelan konseptual yang didesain secara khusus untuk mengidentifikasikan entitas yang menjelaskan data dan hubungan antar data.ERD merupakan suatu model untuk menjelaskan hubungan antardata dalam basis data berdasarkan objek-objek dasar data yang mempunyai hubungan antar relasi. ERD untuk memodelkan struktur data dan hubungan antardata, untuk menggambarkannya digunakan beberapa notasi dan simbol. Pada dasarnya ada tiga komponen yang digunakan, yaitu :

a. *Entity*

Entity merupakan objek yang mewakili sesuatu yang nyata dan dapat dibedakan dari sesuatu yang lain. Simbol dari entiti ini biasanya digambarkan dengan persegi panjang.

b. Atribut

Setiap entitas pasti mempunyai elemen yang disebut atribut yang berfungsi untuk mendeskripsikan karakteristik dari entitas tersebut. Isi dari atribut mempunyai sesuatu yang dapat mengidentifikasikan isi elemen satu dengan yang lain. Gambar atribut diwakili oleh simbol *elips*.

c. Hubungan / Relasi

Hubungan antara sejumlah entitas yang berasal dari himpunan entitas yang berbeda. Relasi dapat digambarkan sebagai berikut :

Relasi yang terjadi diantara dua himpunan entitas (misalnya A dan B) dalam satu basis data yaitu:

1) Satu ke satu (*One to one*)

   Hubungan relasi satu ke satu yaitu setiap entitas pada himpunan entitas A berhubungan paling banyak dengan satu entitas pada himpunan entitas B.

2) Satu ke banyak (*One to many*)

   Setiap entitas pada himpunan entitas A dapat berhubungan dengan banyak entitas pada himpunan entitas B, tetapi setiap entitas pada entitas B dapat berhubungan dengan satu entitas pada himpunan entitas A.

3) Banyak ke banyak (*Many to many*)

   Setiap entitas pada himpunan entitas A dapat berhubungan dengan banyak entitas pada himpunan entitas B.

5. UML(*Unified Modelling Language*)

UML (*Unified Modelling Language*) menurut menurut Adi nugroho (2010:6), "*Unified Modelling Language* adalah bahasa pemodelan untuk sistem atau perangkat lunak yang berpradigma berorientasi objek". Pemodelan (*modeling*) sesungguhnya digunakan untuk penyederhanaan permasalahan-permasalahan yang kompleks sedemikian rupa sehingga lebih mudah dipelajari dan dipahami.

Metode *Unified Modelling Language* (UML) menggunakan tiga bangunan dasar untuk mendeskripsikan sistem atau perangkat lunak yang akan

dikembangkan, yaitu:

a. Sesuatu (*things*)

Ada empat *things* dalam *Unified Modelling Language* (UML):

1. *Structural things*, bagian yang relatif statis dapat berupa elemen-elemen yang bersifat fisik maupun konseptual.
2. *Behavorial things,* bagian dinamis biasanya merupakan kata kerja dari model UML yang mencerminkan perilaku sepanjang waktu.
3. *Grouping things,* bagian pengorganisasian dalam UML. Dalam penggambaran model UML yang rumit di*Perl*ukan penggambaran paket yang menyederhanakan model. Paket-paket ini kemudian dapat didekomposisi lebih lanjut. Paket berguna bagi pengelompokan sesuatu, misalnya model-model serta subsitem-subsistem.
4. *An notational things,* merupakan bagian yang meperjelas model UML. Dapat berisi komentar yang menjelaskan fungsi serta ciri-ciri tiap element dalam model UML.

b. Relasi (*relationship*)

Ada empat *relationship* (hubungan) dalam *Unified Modelling Language* (UML):

1. Ketergantungan (d*ependency*) adalah hubungan dimana perubahan yang terjadi pada suatu elemn independent akan mempengaruhi elemen yang bergantung padanya.

2. Asosiasi adalah apa dan bagaimana yang mengubungkan antara objek satu dengan yang lainnya. Suatu bentuk asosiasi adalah agregasi yang menampilkan hubungan suatu objek dengan bagian-bagiannya.
3. Generalisasi adalah hubungan dimana objek anak berbagai perilaku dan struktur data dari objek yang ada diatasnya (objek induk). Arah dari objek induk ke objek anak dinamakan spesialisasi sedangkan arah sebaliknya dinamakan generalisasi..
4. Realisasi adalah operasi yang benar-benar dilakukan oleh suatu objek.

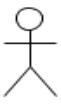
c. Diagram, *Unified Modelling Language* (UML) menyediakan Sembilan jenis diagram yang dapat dikelompokkan berdasarkan sifatnya (statis dan dinamis).

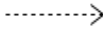
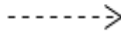
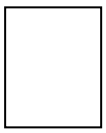
d. diagram *use case*

Diagram *use case* menyajikan interaksi antara *use case* dan aktor, dimana aktor dapat berupa orang, peralatan, atau sistem lain yang berinteraksi dengan sistem yang sedang di bangu. *Use case* menggambarkan fungsionalitas sistem atau persyaratan-persyaratan yang harus dipenuhi sistem dari pandangan pemakai

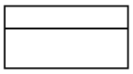
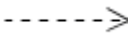
Adapun simbol-simbol *Use Case* Diangram antara lain :

Tabel 2.1. Simbol *Use Case* Diagram

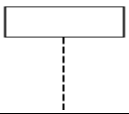
NO	GAMBAR	NAMA	KETERANGAN
1		*Actor*	Men*Spesifikasi*kan himpuan peran yang pengguna mainkan ketika berinteraksi dengan *use case*.

NO	GAMBAR	NAMA	KETERANGAN
2		*Dependency*	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri *(independent)* akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (*independent*).
3		*Generalization*	Hubungan dimana objek anak (*descendent*) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (*ancestor*).
4		*Include*	Men*Spesifikasi*kan bahwa *use case* sumber secara *eksplisit*.
5		*Extend*	Men*Spesifikasi*kan bahwa *use case* target memPerluas perilaku dari *use case* sumber pada suatu titik yang diberikan.
6		*Association*	Apa yang menghubungkan antara objek satu dengan objek lainnya.
7		*System*	Men*Spesifikasi*kan paket yang menampilkan sistem secara terbatas.
8		*Use Case*	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor
9		*Collaboration*	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan prilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi).
10		*Note*	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi

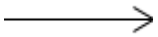
Tabel 2.2. Simbol *Class* Diagram

NO	GAMBAR	NAMA	KETERANGAN
1		*Generalization*	Hubungan dimana objek anak *(descendent)* berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (*ancestor*).
2		*Nary Association*	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
3		*Class*	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
4		*Collaboration*	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor
5		*Realization*	Operasi yang benar-benar dilakukan oleh suatu objek.
6		*Dependency*	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri *(independent)* akan mempegaruhi elemen yang bergantung padanya elemen yang tidak mandiri
7		*Association*	Apa yang menghubungkan antara objek satu dengan objek lainnya

Tabel 2.3. Simbol *Sequence* Diagram

NO	GAMBAR	NAMA	KETERANGAN
1		*LifeLine*	Objek *entity*, antarmuka yang saling berinteraksi.
2		*Message*	*Spesifikasi* dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi
3		*Message*	*Spesifikasi* dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi

Tabel 2.4. Simbol StateChart Diagram

NO	GAMBAR	NAMA	KETERANGAN
1		*State*	Nilai atribut dan nilai link pada suatu waktu tertentu, yang dimiliki oleh suatu objek.
2		*Initial Pseudo State*	Bagaimana objek dibentuk atau diawali
3		*Final State*	Bagaimana objek dibentuk dan dihancurkan
4		*Transition*	Sebuah kejadian yang memicu sebuah state objek dengan cara memperbaharui satu atau lebih nilai atributnya
5		*Association*	Apa yang menghubungkan antara objek satu dengan objek lainnya.
6		*Node*	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi.

Tabel 2.5. Simbol *Actifity* Diagram

NO	GAMBAR	NAMA	KETERANGAN
1		*Actifity*	Mem*Perl*ihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain
2		*Action*	State dari sistem yang mencerminkan eksekusi dari suatu aksi
3		*Initial Node*	Bagaimana objek dibentuk atau diawali.
4		*Actifity Final Node*	Bagaimana objek dibentuk dan dihancurkan
5		*Fork Node*	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran

E. Penelitian Terdahulu

Berbagai penelitian sebelumnya merupakan hal yang sangat Perlu dan dapat dijadikan sebagai data pendukung. Salah satu data pendukung yang menurut peneliti Perlu dijadikan bagian tersendiri adalah penelitian terdahulu yang relevan dengan permasalahan yang sedang dibahas dalam penelitian ini. Dalam hal ini, fokus penelitian terdahulu yang dijadikan acuan adalah terkait dengan masalah teknologi informasi. Oleh karena itu, peneliti melakukan kajian terhadap beberapa hasil penelitian berupa tugas akhir/skripsi atau jurnal melalui internet.

1. Ahmad Sidqi (2020) program studi teknik informatika – program sarjana fakultas teknologi industry Universitas islam indonesia, Melakukan Penelitian Yang Berjudul **"Sistem Monitoring Pekerja pada ninas tenaga kerja"** Proses pembelajaran tahfiz di Madrasah Tsanawiyah Sunan Pandanaran (MTs SPA), pembukuan hasil perkembangan hafalan santri dilakukan secara manual oleh guru. Belum adanya sistem pencatatan yang baik menyebabkan guru terhambat dalam memperoleh informasi santri. Guru hendaknya dapat mengelola informasi seperti *Pre*sensi santri dan berapa jumlah hafalan santri. Selain itu, tidak adanya laporan terhadap perkembangan santri membuat kepala sekolah kesulitan dalam mengetahui hasil pembelajaran santri secara menyeluruh. Pembuatan sistem ini menggunakan beberapa tahapan. Tahapan awal yaitu analisis proses bisnis yang meliputi pengumpulan data, kemudian tahapan kedua yaitu analisis kebutuhan sistem yang meliputi analisis kebutuhan *input*, analisis kebutuhan proses, dan analisis kebutuhan *otput.* Tahapan ketiga pengimplementasian sistem menggunakan *Framework* CI,

tahapan keempat yaitu pengujian yang telah dilakukan menggunakan pengujian kuesioner dan *User Acceptance Test* (UAT).

2. Ridwan Ridwan, Cut Mutia (2021), Fakultas Ilmu Komputer, Universitas Mercu Buana, dalam penelitiannya yang berjudul **"Pengembangan Aplikasi Web dan Mobile Pada dinas tenaga kerja "** Kabupaten Pidie merupakan salah satu lembaga pendidikan yang memiliki program tahfidz. Saat ini, pencatatan hafalan Al-Qur'an setiap siswa dilakukan dalam satu lembar kertas. Keberadaan teknologi informasi menuntut lembaga pendidikan untuk mengimplementasikan aplikasi TI dengan baik. Penelitian ini bertujuan untuk membangun sistem *Monitoring* dan evaluasi tahfidz Al-Qur'an yang baik dalam hal penyajian data kepada MUQ Kabupaten Pidie dan orang tua. Informasi data pada sistem aplikasi *Web* dan mobile yang diusulkan berupa tabel, teks, grafik, dan laporan capaian hafalan. Metode penelitian yang digunakan adalah pendekatan penelitian dan pengembangan dengan model pengembangan aplikasi (*waterfall*). Aplikasi yang digunakan adalah konsep layanan *Web* yang menghubungkan antara aplikasi *Web* dan mobile. Hasil penelitian menunjukkan bahwa pengembangan sistem *Monitoring* dan evaluasi berupa aplikasi *Web* dan mobile untuk pengelolaan data perkembangan tahfidz dengan penyajian data dalam bentuk teks, grafik, dan tabel. Hal ini secara efektif memberikan fungsi *Monitoring* dan evaluasi yang lebih mudah, cepat, dan efisien. Hasil uji efektivitas aplikasi menunjukkan nilai rata-rata persentase dari 19 pertanyaan adalah 96%. Berdasarkan hasil

kuesioner, nilai rata-rata di atas 70%. Secara umum, aplikasi ini sangat membantu orang tua dalam memantau perkembangan hafalan anak mereka.

3. Fulka Juhdan Hanief (2020), Fakultas Teknik dan Ilmu Komputer unikom dalam penelitiannya yang berjudul **"Rancang Bangun Sistem Informasi Monitoring Perizinan Lapak Pada Pelaku Usaha"** Ponpes Uswatun Hasanah saat ini melakukan *Monitoring* kegiatan harian para santri dengan melakukan rapat evaluasi bulanan rutin yang dihadiri oleh seluruh QST, IST, para guru(pembimbing) dan kepala sekolah. Yang dibahas pada rapat evaluasi bulanan ini salah satunya adalah perkembangan akademik santri, baik itu perkembangan santri yang ber*Pres*tasi maupun yang perkembangan akademiknya bermasalah. Informasi perkembangan akademik santri ini meliputi kehadiran santri disetiap kegiatan, kemampuan bacaan Al-Quran, hapalan Qur'an, kemampuan bahasa arab/inggris dan nilai raport santri. Informasi tersebut dituliskan secara *deskriptif* pada kartu evaluasi bulanan yang dikumpulkan oleh semua guru kepada kepala sekolah. Kemudian kepala sekolah melaporkan perkembangan akademik santri yang diperoleh tersebut kepada ketua yayasan. Dalam Proses *Monitoring* yang berjalan saat ini ditemukan beberapa kendala dan keterbatasan, diantaranya yaitu belum adanya proses perhitungan yang mengeloladata nilai-nilai disetiap aspek kegiatan pondok dan belum adanya system yang terintegrasi yang memudahkan para guru dalam mengelola data-data santri sehingga guru kesulitan dalam *Monitoring* kemampuan setiap santrinya. Informasi *Monitoring* perkembangan santri yang didapatkan kepala sekolah melalui

kartu evaluasi bulanan hanya berisi tentang informasi umum perkembangan santri sehingga menyulitkan kepala sekolah untuk membuat laporan *Monitoring* perkembangan santri. Hal ini menyebabkan ketua yayasan pun tidak bisa mengetahui bagaimana informasi *Monitoring* perkembangan santri karena belum ada sistem pelaporan yang jelas

BAB III

METODOLOGI PENELITIAN

A. Waktu Dan Lokasi Penelitian

Pelaksanaan penelitian dilakukan pada bulan Maret 2023 dan lokasi Dinas Tenaga Kerja Kota Parepare.

B. Jenis Penelitian

Penulis dalam hal ini menggunakan metode penelitian *Deskriptif* yaitu kegiatan yang dilakukan dengan cara mengumpulkan data secara langsung dari objek penelitian melalui wawancara, pengamatan langsung dan pengumpulan dokumen.

C. Metode Pengumpulan Data

Di penelitian ini, penulis memperoleh data dengan metode sebagai berikut:

1. Observasi

Dilakukan dengan mengadakan penelitian langsung dengan kantor yang terkait untuk mengumpulkan informasi dan data yang dibutuhkan dalam menunjang permasalahan.

2. Wawancara

Wawancara adalah bentuk komunikasi langsung antara peneliti dengan *responden*. Komunikasi ini berlangsung dalam bentuk tanya jawab dalam hubungan tata muka guna memperoleh data langsung

3. Studi Pustaka

Studi pustaka adalah suatu metode pengumpulan data dengan menggunakan buku-buku, laporan-laporan atau bacaan lain sebagai bahan referensi dalam penulisan laporan dan pembuatan sistem.

D. Alat Dan Bahan Penelitian

Dalam melakukan penelitian, penulis memerlukan alat dan bahan penelitian yang mendukung kegiatan tersebut. Alat dan bahan yang di*Perl*ukan antara lain:

1. Perangkat keras

Perangkat keras yang digunakan dalam pembuatan Perancangan Aplikasi Perizinan Lembaga Pelatihan Kerja Kota Parepare adalah sebuah laptop dengan *Spesifikasi* minimal yang dapat dilihat pada tabel 4 berikut :

Tabel 3.1. *Spesifikasi* Minimal Perangkat Keras

Jenis	**Spesifikasi**
Laptop	Asus
Processor	Core-i3-3217U 1,8 GHz
Memory	4 GB
Hardisk	500 GB

2. Perangkat Lunak

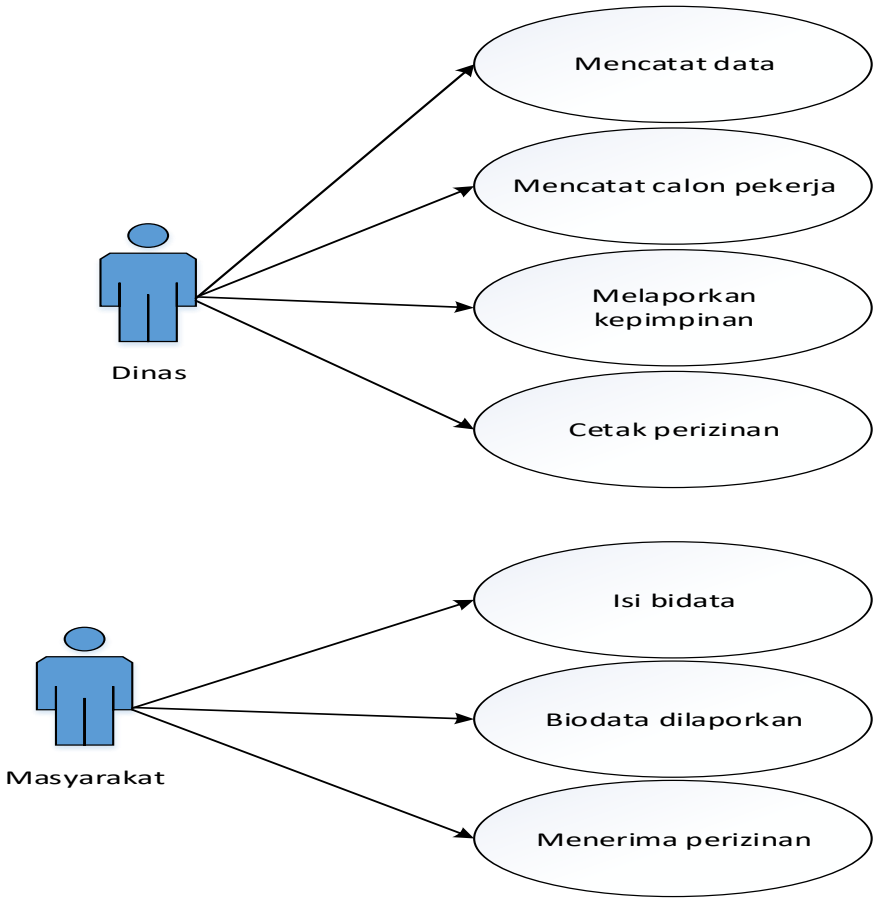
Perangkat lunak yang digunakan untuk membangun aplikasi ini dapat dilihat pada Tabel 5 berikut :

Tabel 3.2 : *Spesifikasi* Minimal Perangkat Lunak

Jenis	Spesifikasi
Sistem Operasi	*Windows 11*
Bahasa Pemrograman	*PHP*
Database	*MySql*
Tools	*Notepad++, Netbeans, XAMPP*

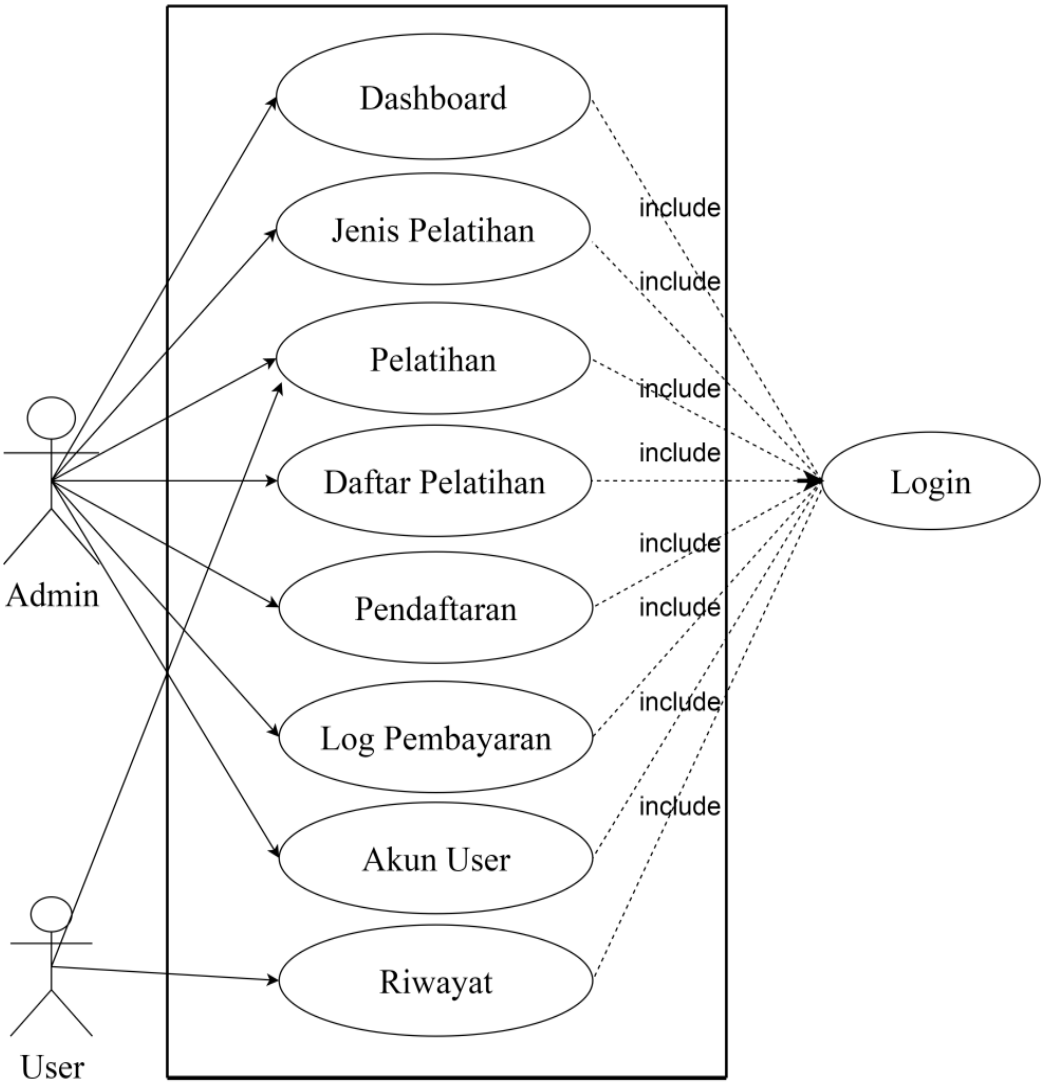
E. Desain Sistem

1. Sistem yang berjalan



Gambar 3.1 Sistem Berjalan

2. Sistem yang diusulkan



Gambar 3.2 Sistem yang diusulkan

BAB IV

HASIL PENELITIAN DAN PEMBAHASAN

A. Hasil Penelitian

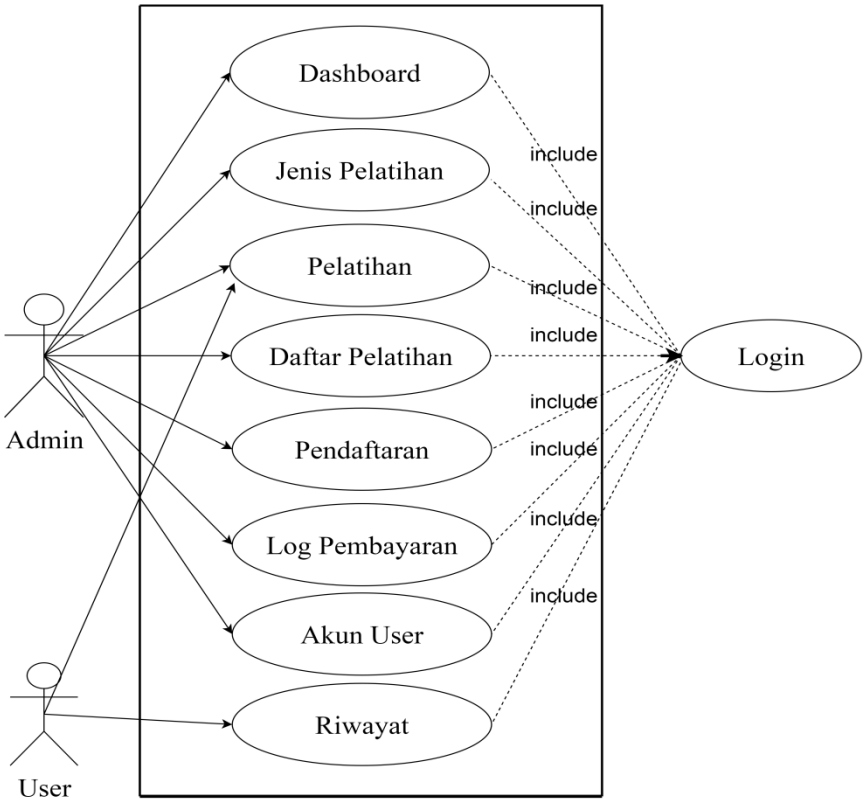
Aplikasi Perizinan Lembaga Pelatihan Kerja Kota Parepare, Menggunakan pemrograman *Javascript,* sedangkan untuk pengelolaan basis data menggunakan *MySQL*. Metode yang digunakan dalam perancangan aplikasi ini adalah UML (*Unified Modelling Langguage*).

B. Pembahasan

1. Analisis Aliran Data UML

a. *Use Case Diagram*

Use Case Diagram atau diagram *use case* merupakan pemodelan untuk memodelkan kelakuan (*behavior*) sistem informasi yang dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi. *Use case* juga dapat digunakan untuk mere*Pre*sentasikan interaksi yang terjadi antara aktor dengan proses sistem yang dibuat. Pada perancangan *use case* juga terdapat skenario, yaitu langkah yang menerangkan urutan kejadian antara pengguna dengan sistem. Diagram *use case* pada Aplikasi ini digambarkan seperti berikut ini:



Gambar 4 1. Use Case Diagram.

b. Tabel *Use Case*

Penjabaran *Use Case* diagram di atas didefinisikan atas definisi aktor dan definisi *Use Case* pada sistem yang diusulkan. Definisi aktor sebagai berikut :

Tabel 4.1 Definisi Aktor dalam Sistem.

No	Aktor	Keterangan
1	*User*	Pengguna *User* merupakan aktor yang berperan sebagai pengguna aplikasi Perizinan Lembaga Pelatihan Kerja Kota Parepare.
2	*Admin*	*Admin* merupakan aktor yang dapat ,mengelola *User*.

Sedangkan untuk penjelasan definisi setiap *Use Case* pada sistem yang diusulkan sebagai berikut :

Tabel 4.2 Definisi Use Case Sistem

No	*Use Case*	Keterangan
1	*Login*	Menampilkan halaman *login* kemudian masukkan *username* dan *password*
2	*Dashboard*	Menampilkan halaman utama aplikasi
3	Jenis Pelatihan	Menampilkan halaman Jenis Pelatihan
4	Pelatihan	Menampilkan halaman Pelatihan
5	Daftar Pelatihan	Menampilkan halaman Daftar Pelatihan
6	Pendaftaran	Manampilkan halaman Pendaftaran
7	Log Pembayaran	Manampilkan halaman Log Pembayaran
8	Akun *User*	Manampilkan halaman Akun *User*
9	Riwayat	Menampilkan halaman Riwayat

Dalam *Use Case* yang telah didefinisikan di atas, terdapat beberapa *Use Case* yang memiliki alur proses yang hampir sama, sehingga dalam penjabaran selanjutnya akan didefinisikan secara ringkas untuk memudahkan pemahaman proses tiap *Use Case*. Berikut adalah skenario *Use Case* yang telah didefinisikan sebagai berikut :

1) *Use Case Login*

Aktor : *Admin* & *User*

Deskripsi : Sebuah kegiatan untuk masuk kedalam aplikasi.

Pre-condition : *Admin*, *User* berada di halaman *Login*.

Post-condition : -

Tabel 4 3. Skenario *Use Case Login*

Aksi Aktor	Reaksi Sistem
1. Masuk kehalaman *Login* dan Memasukkan *Username* dan Password.	
	2. Masuk ke halaman utama aplikasi

2) *Use Case Home*.

Aktor : *Admin*, *User*

Deskripsi : Sebuah kegiatan untuk menampilkan menu-menu aplikasi.

Pre-condition : *Admin*, *User* berada di halaman utama.

Post-condition : -

Tabel 4 4. Skenario *Use Case Home*.

Aksi Aktor	Reaksi Sistem
1. Masuk kehalaman utama aplikasi	
	2. Menampilkan menu-menu aplikasi.

3) *Use Case* Jenis Pelatihan

Aktor : *Admin*

Deskripsi : Sebuah kegiatan untuk menampilkan Jenis Pelatihan.

Pre-condition : *Admin* berada di halaman Jenis Pelatihan.

Post-condition : Data dalam sistem akan diperbarui.

Tabel 4 5. Skenario *Use Case* Jenis Pelatihan

Aksi Aktor	Reaksi Sistem
1. Memilih menu Jenis Pelatihan	
	2. Menampilkan form Jenis Pelatihan

4) Skenario *Use Case* Pelatihan

Aktor : *Admin* dan *User*

Deskripsi : Sebuah kegiatan untuk melihat Pelatihan

Pre-condition : *Admin* dan *User* berada di halaman Pelatihan

Post-condition : Data dalam sistem akan diperbarui.

Tabel 4 6. Skenario *Use Case* Pelatihan

Aksi Aktor	Reaksi Sistem
1. Menekan menu Pelatihan	
	2. Menampilkan data Pelatihan

5) *Use Case* Daftar Pelatihan

Aktor : *Admin*

Deskripsi : Sebuah kegiatan untuk melihat Daftar Pelatihan.

Pre-condition : *Admin* berada di halaman Daftar Pelatihan.

Post-condition : Data dalam sistem akan diperbarui.

Tabel 4 7. Skenario *Use Case* Daftar Pelatihan

Aksi Aktor	Reaksi Sistem
1. Menekan menu Daftar Pelatihan	
	2.Menampilkan menu Daftar Pelatihan

6) *Use Case* Pendaftaran

Aktor : *Admin*

Deskripsi : Sebuah kegiatan untuk melihat BukPendaftaran.

Pre-condition : *Admin* berada di halaman BukPendaftaran

Post-condition : Data dalam sistem akan diperbarui.

Tabel 4 8. Skenario *Use Case* Pendaftaran

Aksi Aktor	Reaksi Sistem
1.Menekan menu Pendaftaran	
	2.Menampilkan menu Pendaftaran

7) *Use Case* Log Pembayaran

Aktor : *Admin*

Deskripsi : Sebuah kegiatan untuk melihat Log Pembayaran

Pre-condition : *Admin* berada di halamanLog Pembayaran.

Post-condition : Data dalam sistem akan diperbarui.

Tabel 4 9. Skenario *Use Case* Log Pembayaran

Aksi Aktor	Reaksi Sistem
1.Menekan menu Log Pembayaran	
	2.Menampilkan menu Log Pembayaran

8) *Use Case* Akun *User*

Aktor : *Admin*

Deskripsi : Sebuah kegiatan untuk melihat LapAkun *User*

Pre-condition : *Admin* berada di halaman Akun *User*

Post-condition : Data dalam sistem akan diperbarui.

Tabel 4 10. Skenario Use Case Akun User

Aksi Aktor	Reaksi Sistem
1.Menekan menu Akun *User*	
	2. Menampilkan menu Akun *User*

9) *Use Case* Riwayat

Aktor : *User*

Deskripsi : Sebuah kegiatan untuk melihat Riwayat

Pre-condition : *User* berada di halaman Riwayat.

Post-condition : Data dalam sistem akan diperbarui.

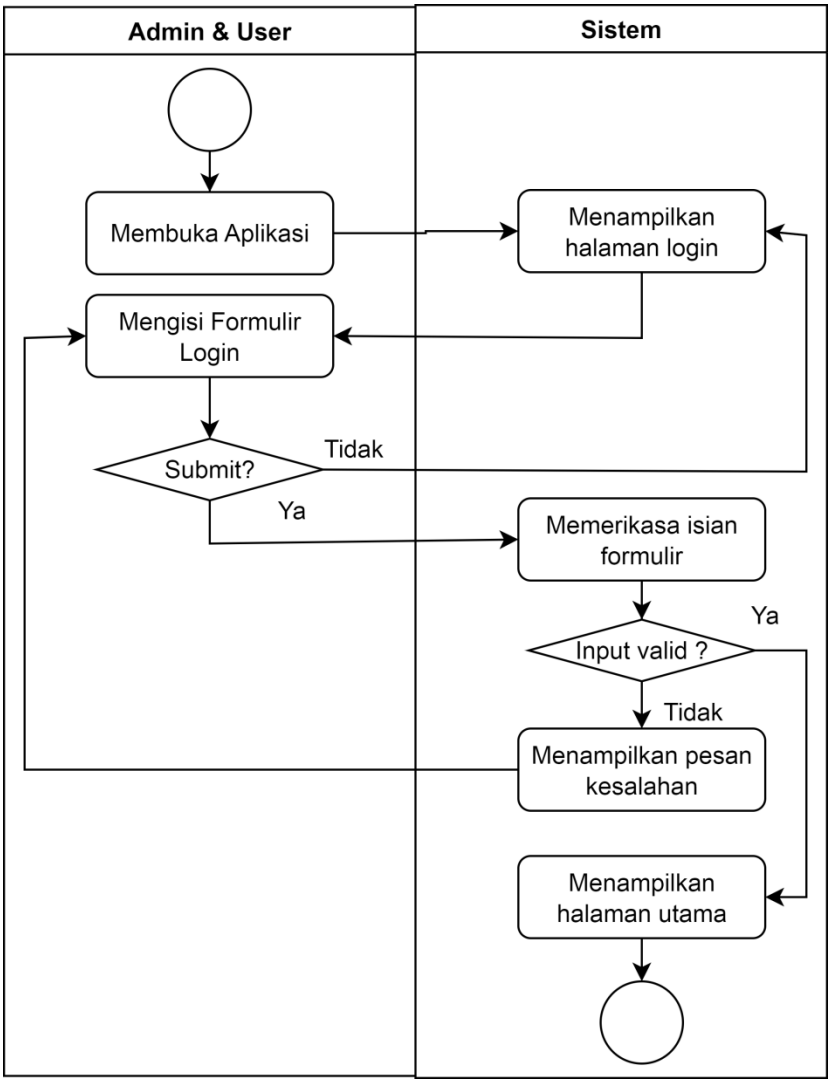
Tabel 4 11. Skenario Use Case Riwayat

Aksi Aktor	Reaksi Sistem
1.Menekan menu Riwayat	
	2. Menampilkan menu Riwayat

2. Activity Diagram

Activity diagram adalah bahasa *Spesifikasi* standar yang dipergunakan untuk mendokumentasikan, men*Spesifikasi*kan dan membanngun perangkat lunak *Activity diagram* ini menjelaskan tentang aktifitas-aktifitas yang terjadi dalam sebuah aliran proses pada sebuah system.

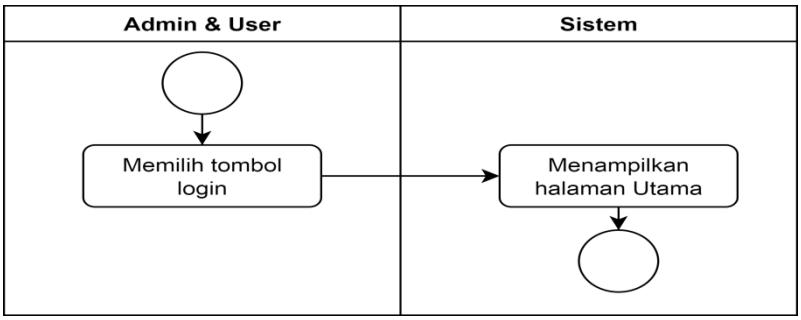
a. ***Activity Diagram Login***



Gambar 4 2. ***Activity Diagram Login***

Diagram aktivitas di atas menerangkan alur proses untuk aktivitas *Login*. Proses dimulai dengan *Admin* dan *User* memilih tombol *Login*, kemudian sistem menampilkan form isian yang kemudian *Admin* atau *User* akan mengisi form isian. Ketika sudah disimpan, sistem akan memperbarui data di *database* yang akan mengecek data yang dikirim, jika terdapat kesalahan, akan menampilkan pesan kesalahan, jika tidak, sistem menampilkan halaman utama.

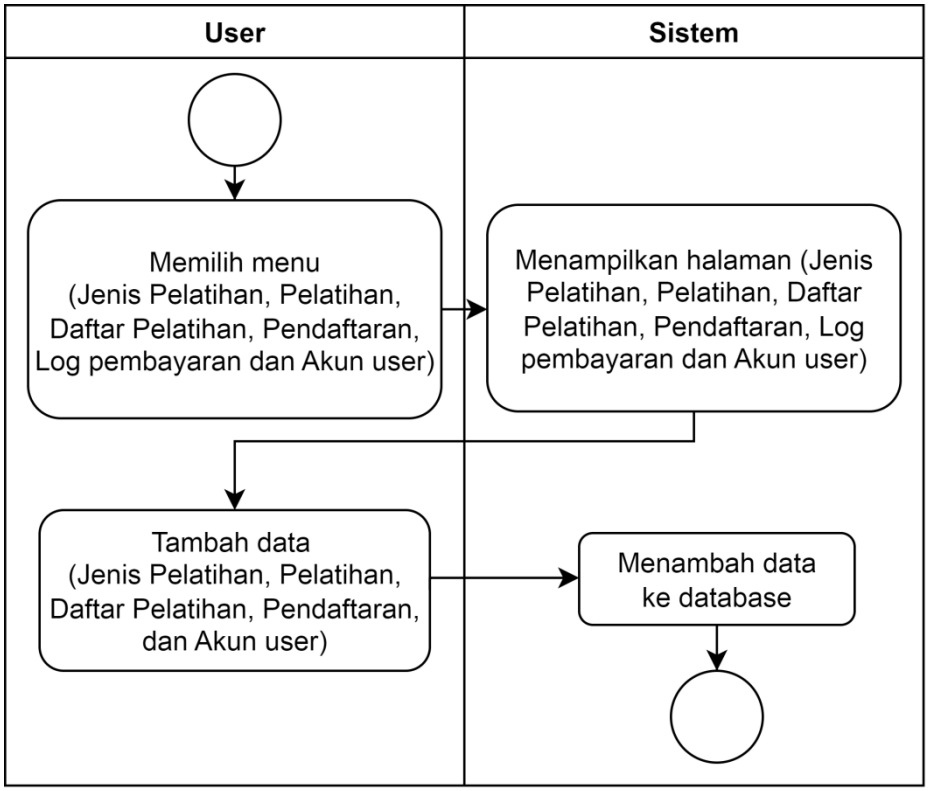
b. ***Activity Diagram Home***



Gambar 4 3. ***Activity Diagram Home***

Diagram aktivitas di atas menerangkan alur proses untuk aktivitas *Home*. Proses dimulai dengan *Admin* dan *User* memilih tombol *Login*, kemudian system berhasil menampilkan halaman utama aplikasi.

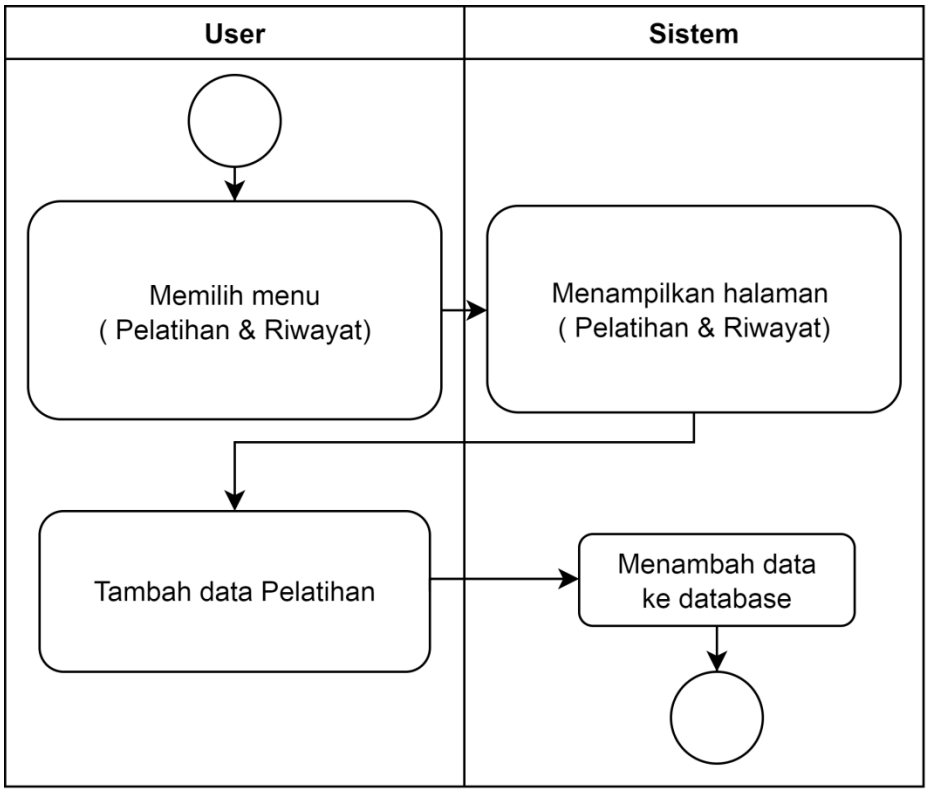
c. ***Activity Diagram Admin***



Gambar 4 4. ***Activity Diagram Admin***

Diagram aktivitas berikut menerangkan alur proses untuk aktivitas *Admin* Proses dimulai dengan Pengguna admmin memilih salah satu mmenu di atas, kemudian sistem menampilkan halaman yang dipilih. Jika ingin tambah data, maka dapat memilih tambah data sistem akan menambahkan data tersebut ke *database*.

d. *Activity Diagram User*



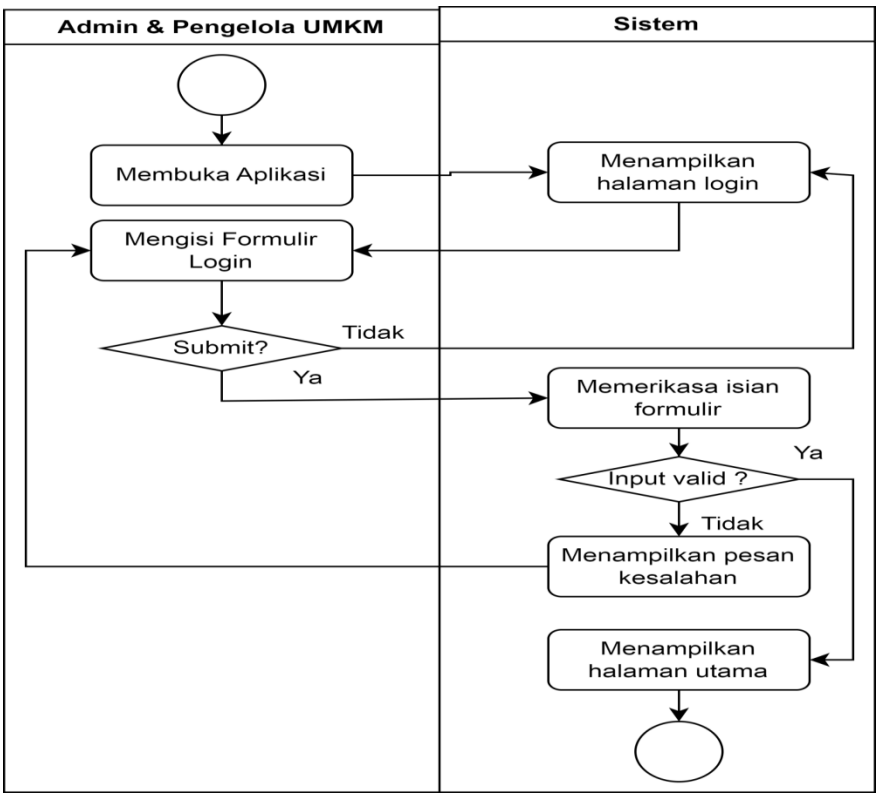
Gambar 4 5. *Activity* Diagram Jurnal Penyesuaian

Diagram aktivitas berikut menerangkan alur proses untuk aktivitas *User* Proses dimulai dengan *User* memilih mmmenu yang ada diatas, kemudian sistem menampilkan halaman yang dipilih. Jika ingin tambah data, maka dapat memilih tambah data sistem akan menambahkan data tersebut ke *database*.

3. *Sequence* Diagram

Sequence diagram atau diagram sekuen menggambarkan kelakuan objek pada *Use Case* dengan mendeskripisikan waktu hidup objek dan *message* yang dikirim dan diterima antar objek. Proses menggambarkan diagram sekuen maka harus diketahui objek-objek yang terlibat dalam sebuah *Use Case* beserta *method* yang dimiliki kelas yang diinisialisasi menjadi objek yang sudah tergambar dalam class diagram. Berdasarkan desain *Use Case*, terdapat beberapa *Use Case* yang prosesnya hampir sama satu sama lain. Untuk mempermudah pembaha-san proses dalam pembuatan diagram sekuen, berikut ini ringkasan diagram sekuen pada Aplikasi Perizinan Lembaga Pelatihan Kerja Kota Parepare

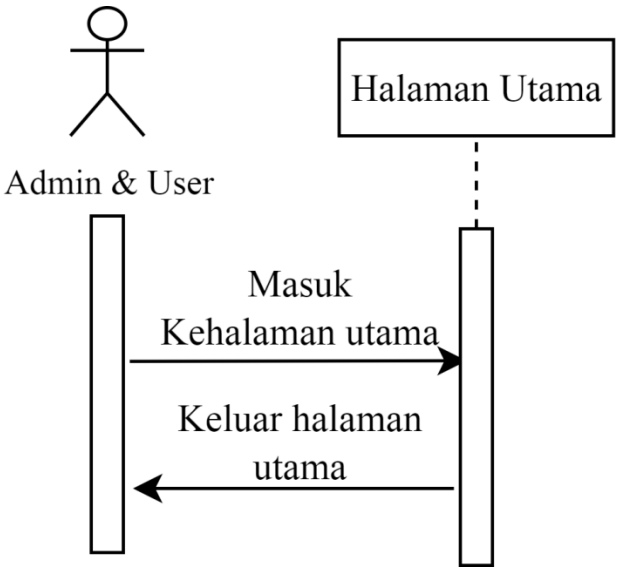
a. *Sequence Diagram*: *Login*



Gambar 4 5. *Sequence Diagram login*

Diagram *Sequence* diagram *Login Admin* dan *User* merupakan penggambaran proses *Login* yang dilakukan *Admin* dan *User*. Bila proses *Login* benar *Admin* dan *User* bisa masuk ke dalam halaman utama sebaliknya bila proses *Login* gagal akan kembali ke menu *login*.

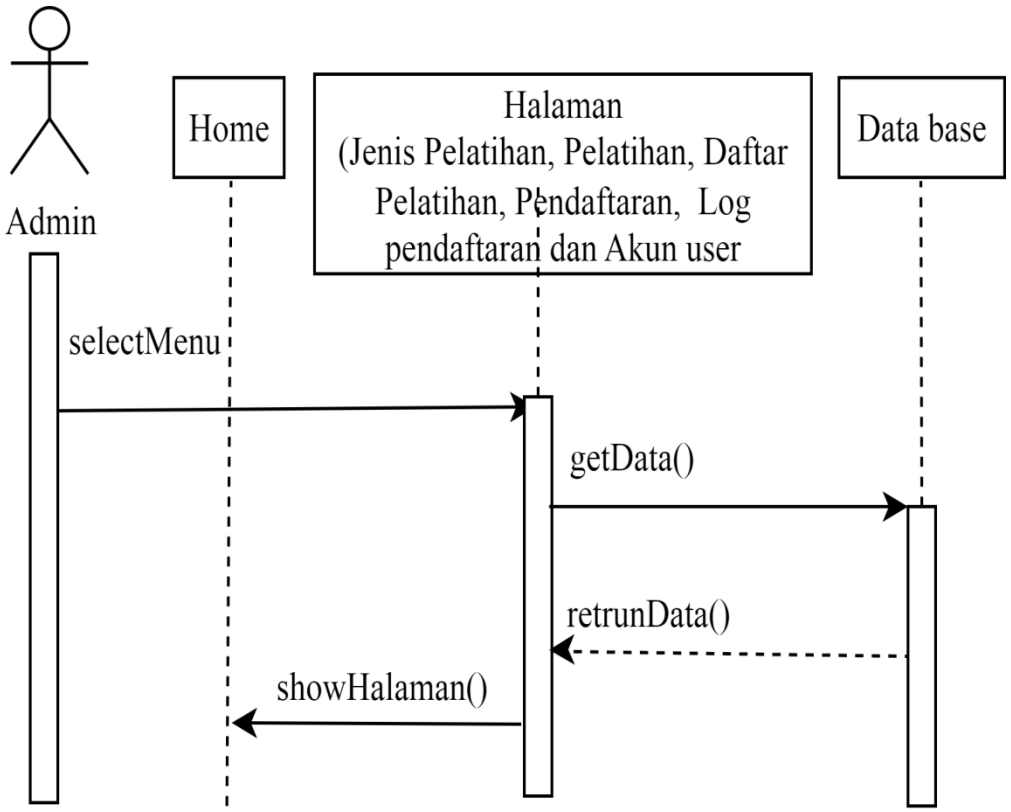
b. *Sequence Diagram*: *Home*



Gambar 4 6. *Sequence Diagram Home*

Diagram *Sequence* diagram *Home Admin* dan *User* merupakan penggambaran proses masuk ke halaman utama. Bila proses *Login* benar *Admin* dan *User* bisa masuk ke dalam halaman utama dan menampilkan menu-menu yang ada pada aplikasi.

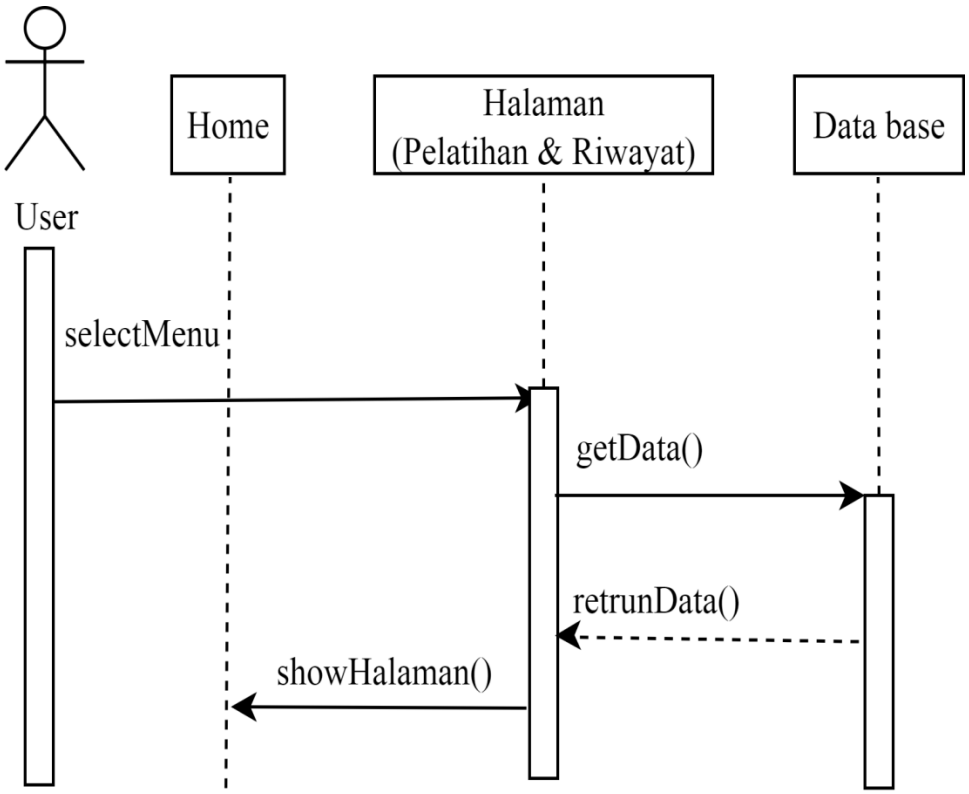
c. *Sequence Diagram Admin*



Gambar 4 7. *Sequence Diagram Admin*

Diagram sekuen di atas menerangkan alur proses untuk aktivitas *Admin*. *Admin* memilih data yang akan ditampilkan pada halaman *Admin*. kemudian *method* get Data digunakan untuk mengambil data yang akan ditampilkan grafiknya ke *database*. *Database* akan mengembalikan data, kemudian *method show* Halaman akan menampilkan halaman yang ingin ditampilkan.

d. *Sequence Diagram User*



Gambar 4 8. *Sequence Diagram User*

Diagram sekuen di atas menerangkan alur proses untuk aktivitas *User*. *User* memilih data yang akan ditampilkan pada halaman *User*. kemudian *method* get Data digunakan untuk mengambil data yang akan ditampilkan grafiknya ke *database*. *Database* akan mengembalikan data, kemudian *method show* Halaman akan menampilkan halaman yang ingin ditampilkan.

4. Kamus Data

Kamus data merupakan kumpulan data yang digunakan serta dihasilkan oleh perangkat lunak. Berikut ini daftar seluruh data yang akan digunakan dan dihasilkan oleh perangkat lunak sistem laporan keuangan UMKM.

Tabel 4.14. *User*s
Primary key
Foreign key

No	Nama *Field*	Tipe Data	Ukuran
1	id	*bigint*	20
2	*User*s_id	*bigint*	20
3	*phone*	*Varchar*	16
4	*name*	*Varchar*	255
5	nomor_hp	*Varchar*	255

Tabel 4. 15 Jenis Pelatihan
Primary key
Foreign key

No	Nama *Field*	Tipe Data	Ukuran
1	id	*bigint*	20
2	*name*_pelatihan	*Varchar*	255
3	status	enum(‘active,’inactive’	-
4	address	*Varchar*	255

Tabel 4.16. Pelatihan

Primary key
Foreign key

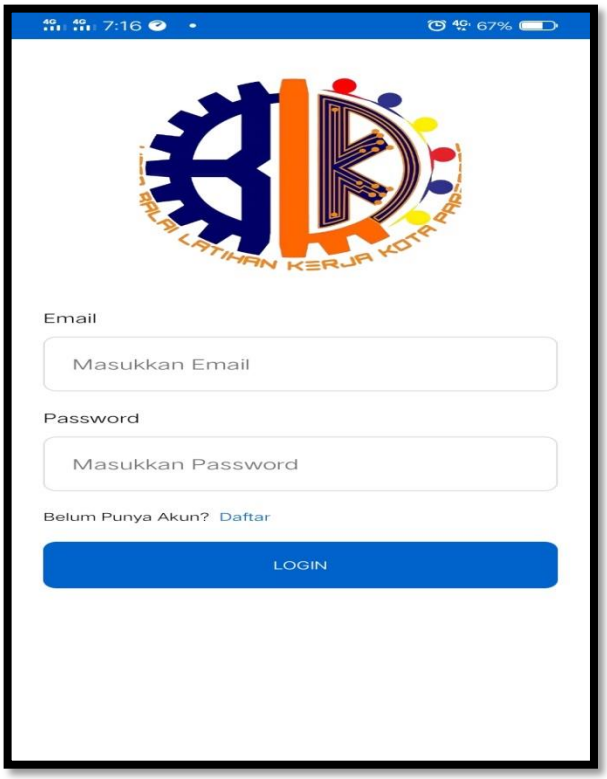
No	Nama *Field*	Tipe Data	Ukuran
1	id	*bigint*	20
2	pelatihan_id	*bigint*	20
3	*deleted*_at	*timestamp*	-
4	*created*_at	*timestamp*	-
5	*updated*_at	*timestamp*	-

5. Perancangan Sistem

a. Halaman *Admin*

1) Halaman *Login*

Berikut adalah tampilan *Login* pada Aplikasi Perizinan Lembaga Pelatihan Kerja Kota Parepare. Halaman menu *Login* merupakan tampilan yang ada pada sistem, yang digunakan untuk masuk ke halaman selanjutnya dengan memasukkan *username* dan *password*.



Gambar 4 15. Halaman *Login*

2) Halaman *Dashboard*

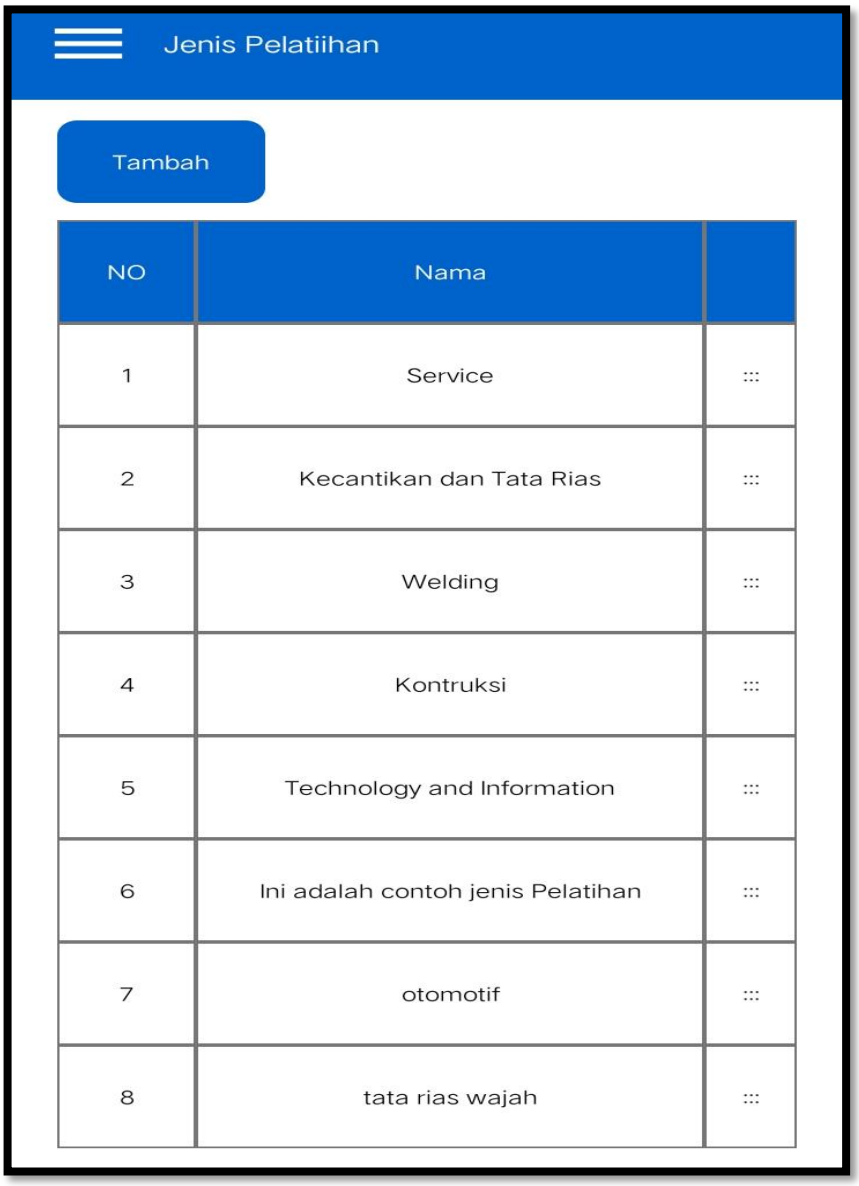
Berikut adalah tampilan *dashboard* pada Aplikasi Perizinan Lembaga Pelatihan Kerja Kota Parepare. Halaman *dashboard* dalam aplikasi ini berfungsi sebagai pusat informasi utama dan titik navigasi bagi *Admin*.



Gambar 4 16. Halaman *Dashboard*

3) Halaman Jenis Pelatihan

Berikut adalah tampilan Jenis Pelatihan pada Aplikasi Perizinan Lembaga Pelatihan Kerja Kota Parepare.. Halaman Jenis Pelatihan dalam aplikasi ini berfungsi sebagai pusat informasi utama pada menu Jenis Pelatihan.



Gambar 4 17. Halaman Jenis Pelatihan

4) Halaman Pelatihan

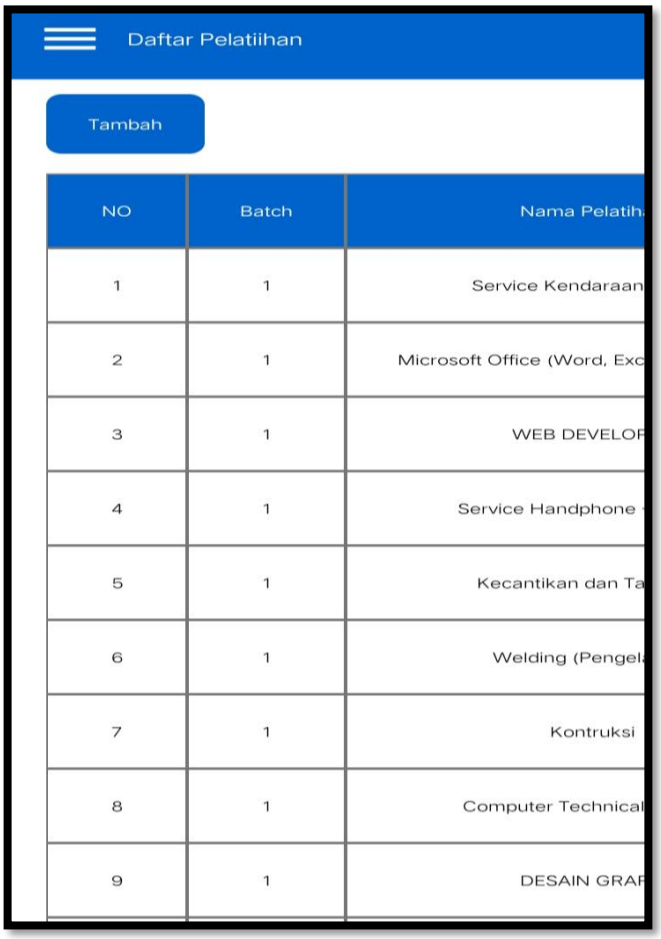
Berikut adalah tampilan Pelatihan pada Aplikasi Perizinan Lembaga Pelatihan Kerja Kota Parepare.. Halaman Pelatihan dalam aplikasi ini berfungsi sebagai pusat informasi utama pada menu Pelatihan.



Gambar 4 18. Halaman Pelatihan

5) Halaman Daftar Pelatihan

Berikut adalah tampilan Daftar Pelatihan pada Aplikasi Perizinan Lembaga Pelatihan Kerja Kota Parepare.. Halaman Daftar Pelatihan dalam aplikasi ini berfungsi sebagai pusat informasi utama pada menu Daftar Pelatihan.



Gambar 4 19. Halaman Daftar Pelatihan

6) Halaman Log Pembayaran

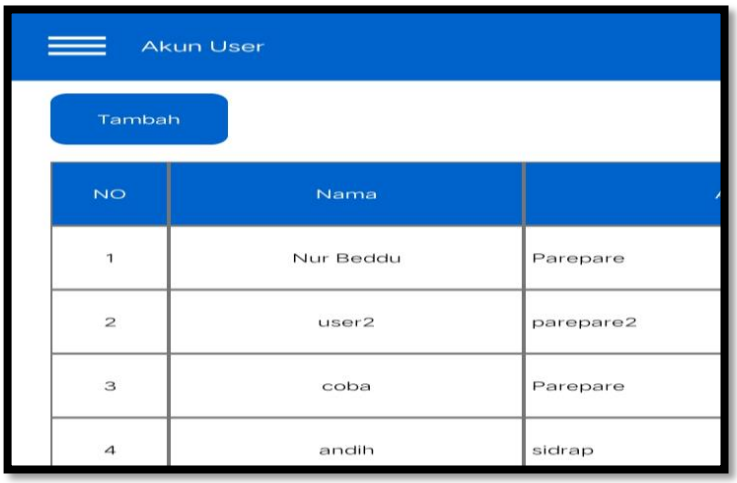
Berikut adalah tampilan Log Pembayaran pada Aplikasi Perizinan Lembaga Pelatihan Kerja Kota Parepare.. Halaman Log Pembayaran dalam aplikasi ini berfungsi sebagai pusat informasi utama pada menu Log Pembayaran.



Gambar 4 20. Halaman Log Pemmbayaran

7) Halaman Akun *User*

Berikut adalah tampilan akun *User* pada Aplikasi Perizinan Lembaga Pelatihan Kerja Kota Parepare.. Halaman akun *User* dalam aplikasi ini berfungsi sebagai pusat informasi utama pada menu akun *User*.

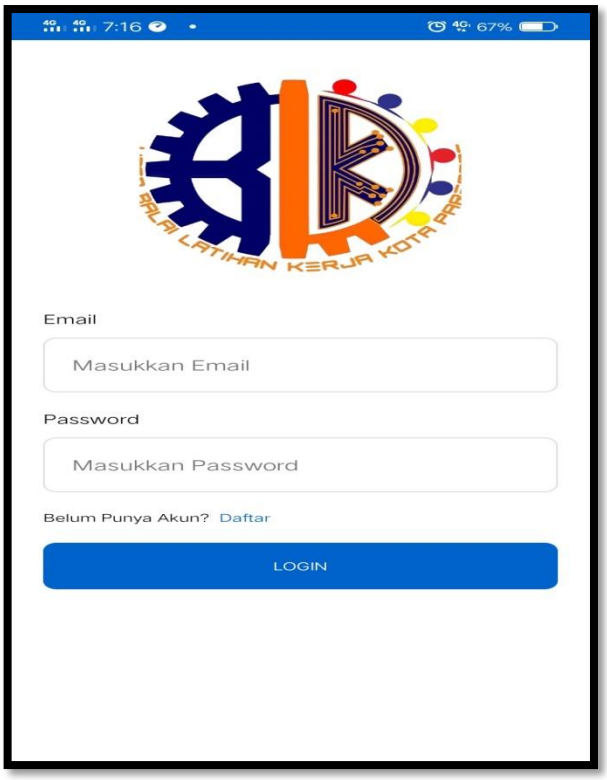


Gambar 4 21. Halaman Akun *User*

b. Halaman *User*

1) Halaman *Login*

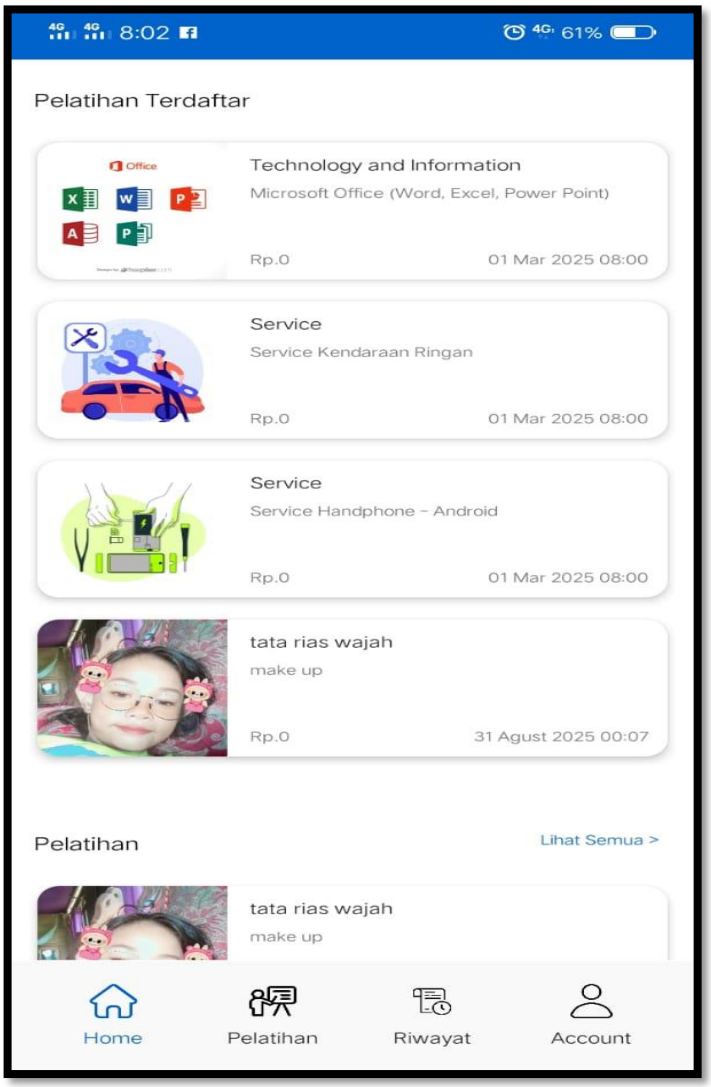
Berikut adalah tampilan *Login* pada Aplikasi Perizinan Lembaga Pelatihan Kerja Kota Parepare. Halaman menu *Login* merupakan tampilan yang ada pada sistem, yang digunakan untuk masuk ke halaman selanjutnya dengan memasukkan *username* dan *password*.



Gambar 4 22. Halaman *Login*

2) Halaman *Dashboard*

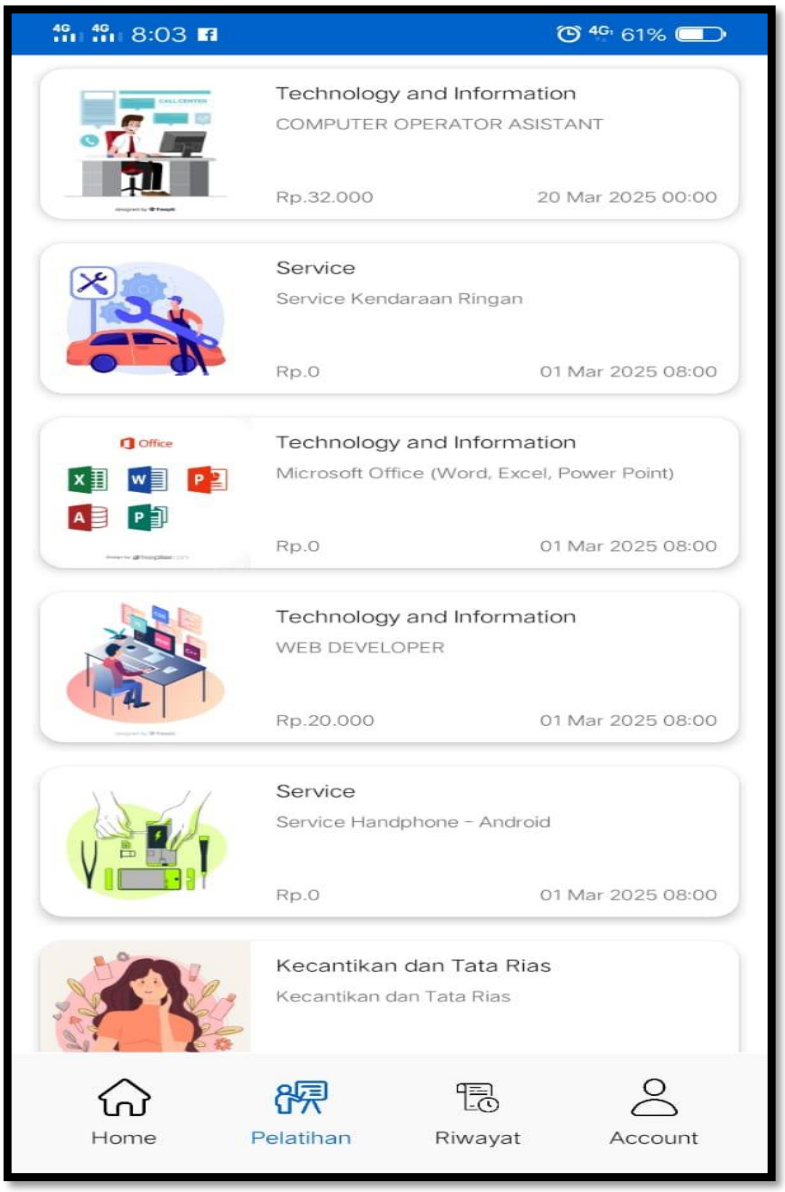
Berikut adalah tampilan *dashboard* pada Aplikasi Perizinan Lembaga Pelatihan Kerja Kota Parepare. Halaman *dashboard* dalam aplikasi ini berfungsi sebagai pusat informasi utama dan titik navigasi bagi *User*.



Gambar 4 23. Halaman *Dashboard*

3) Halaman Pelatihan

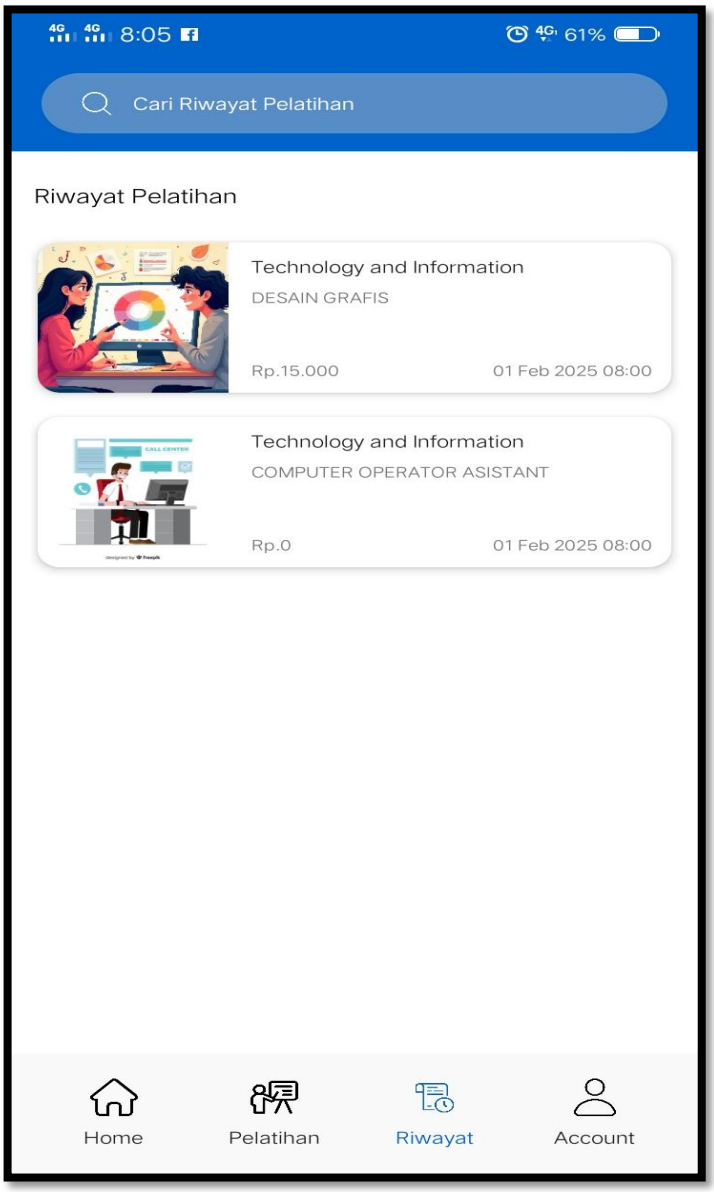
Berikut adalah tampilan Pelatihan pada Aplikasi Perizinan Lembaga Pelatihan Kerja Kota Parepare.. Halaman Pelatihan dalam aplikasi ini berfungsi sebagai pusat informasi utama pada menu Pelatihan.



Gambar 4 24. Halaman Pelatihan

4) Halaman Riwayat

Berikut adalah tampilan riwayat pada Aplikasi Perizinan Lembaga Pelatihan Kerja Kota Parepare.. Halaman riwayat dalam aplikasi ini berfungsi sebagai pusat informasi utama pada menu riwayat.



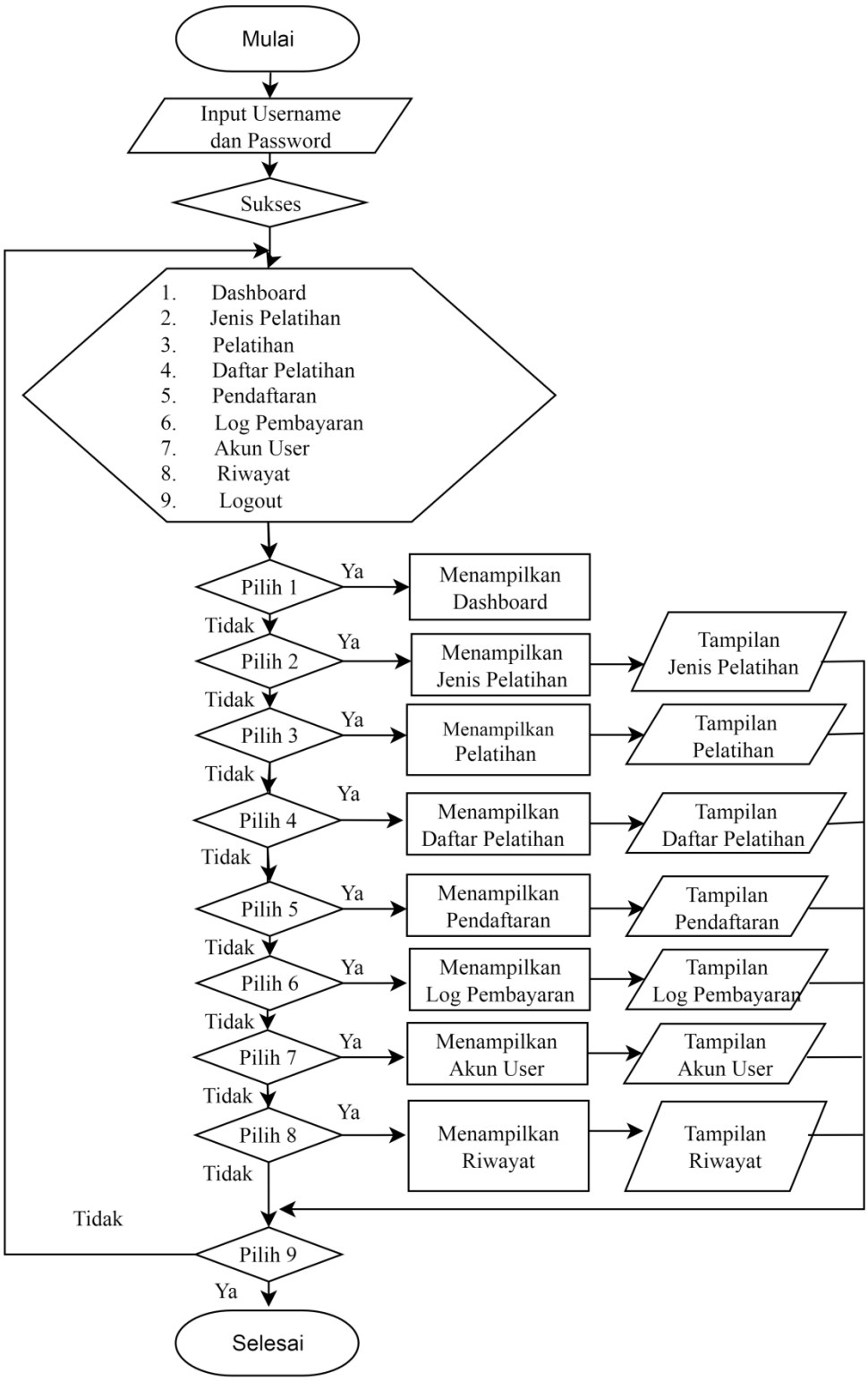
Gambar 4 25. Halaman Riwayat

6. Pengujian Sistem

a. *WhiteBox*

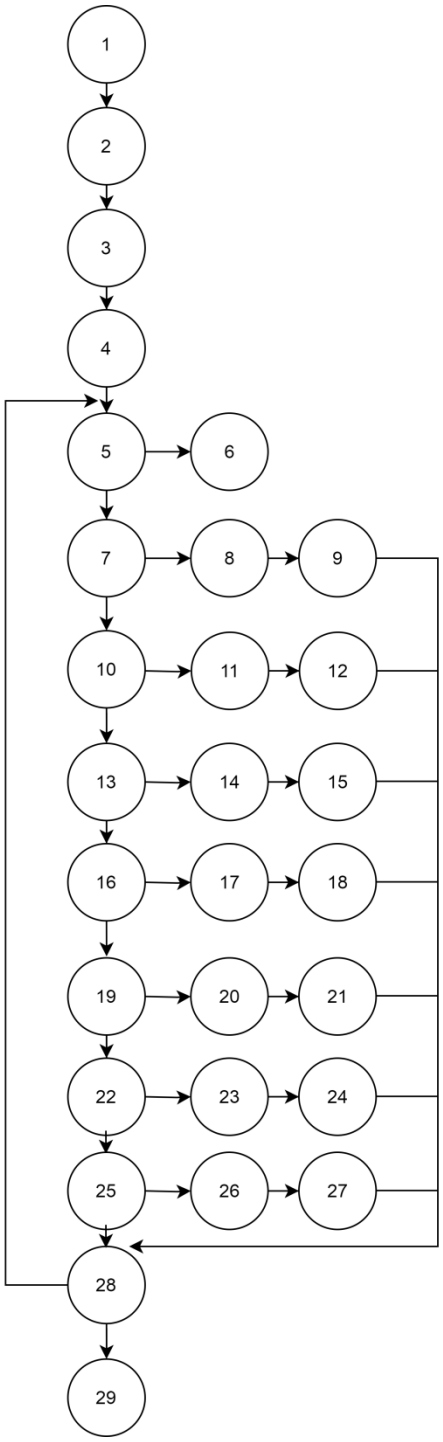
Pengujian aplikasi dilakukan dengan cara pengujian *WhiteBox*. *WhiteBox* adalah metode pengujian perangkat lunak yang menguji struktur internal atau cara kerja aplikasi.

1) *Flowchart* dan *Flowgraph* Aktivitas *Admin* dan *User*



Gambar 4 26. *Flowchart* Aktivitas *Admin* dan *User*

Dari *Flowchart* yang digunakan untuk pengujian perangkat lunak, maka ditentukan *Flowgraph* sebagai berikut:



Gambar 4. 27. *Flowgraph* Aktivitas *Admin* dan *User*

Dari *Flowgraph* aktivitas *admin* dan *User* diatas dapat dilakukan proses perhitungan sebagai berikut:

a) Menghitung *Cyclomatic Complexcity* V(G) dari *Egde* dan *Node*:

Diketahui :

N = 22 E = 29 R = 2

Penyelasaian :

V(G) = (E- N) + 2

= (29 – 22) + 2 = 9

Independent Path :

Path 1 = 1 – 2 – 3 – 4 – 5 –6

Path 2 = 1 – 2 – 3 – 5 – 7 – 8 –9 – 29

Path 3 = 1 – 2 – 3 – 5 – 7 – 10 –11 –12– 29

Path 4 = 1 – 2 – 3 – 5 – 7 – 10 –13 –14 – 15 –29

Path 5 = 1 – 2 – 3 – 5 – 7 – 10 –13 –16 – 17– 18 –29

Path 6 = 1 – 2 – 3 – 5 – 7 – 10 –13 –16 – 19– 20 –21–29

Path 7 = 1 – 2 – 3 – 5 – 7 – 10 –13 –16 – 19– 22 –23–24–29

Path 8 = 1 – 2 – 3 – 5 – 7 – 10 –13 –16 – 19– 22 –25–26–27–29

Path 9 = 1 – 2 – 3 – 5 – 7 – 10 –13 –16 – 19– 22–25–28–29

Tabel 4 1. *White Box* Perhitungan menu *Admin* dan *User*

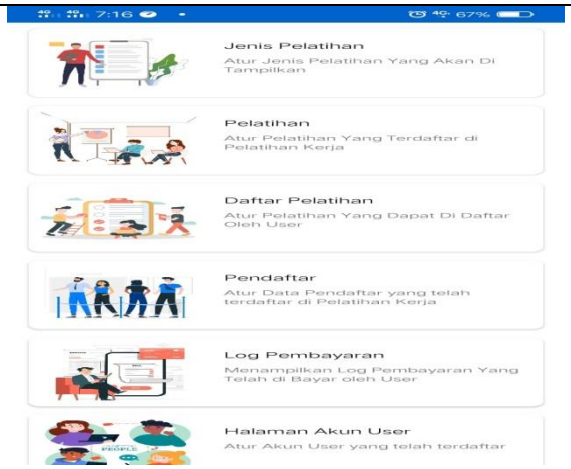
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	E-1
1		1																			1-1=0
2			1																		1-1=0
3				1																	1-1=0
4					1																1-1=0

5						1	1														2-1=1
6																					0
7								1		1											2-1=1
8									1												1-1=0
9																					1-1=0
10											1		1								2-1=1
11												1									1-1=0
12																					1-1=0
13														1		1					2-1=1
14															1						1-1=0
15																					1-1=0
16																	1		1		2-1=1
17																		1			1-1=0
18																					1-1=0
19																				1	2-1=1
20																					0
SUM(E+1)																					8+1=9

b. *Black box*

Pengujian *BlackBox* digunakan untuk menguji sistem. *BlackBox* adalah metode pengujian perangkat lunak yang memeriksa fungsionalitas aplikasi tanpa mempelajari struktur atau cara kerja internalnya. Pendekatan pengujian ini berlaku untuk hampir semua tingkat pengujian perangkat lunak, termasuk unit, integrasi, sistem, dan penerimaan.

Tabel 4 19. *Black Box* Halaman Utama

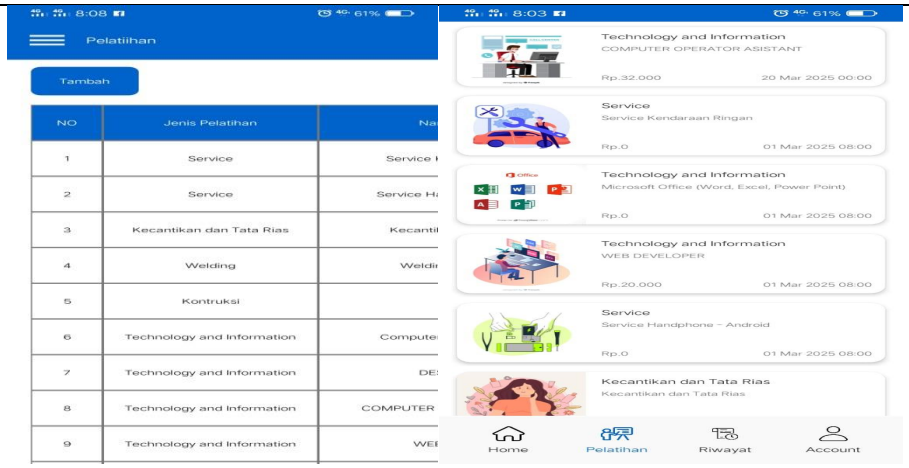
Test Faktor	Hasil	Kesimpulan
Jika memasukkan *Username* dan password dengan benar.	✓	Informasi, tampil halaman utama *Admin* dan *User*.
Screen Shot		
		

Tabel 4 2. *Black Box* Form Jenis Pelatihan

Test Faktor	Hasil	Kesimpulan
Jika memilih menu jenis pelatihan pada halaman utama.	✓	Informasi, tampil form Jenis Pelatihan
Screen Shot		
		

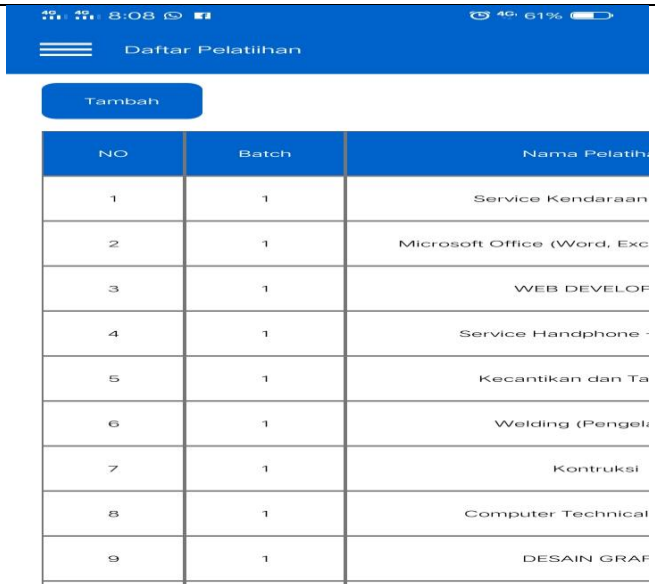
Tabel 4 21. *Black Box* Pelatihan

Test Faktor	Hasil	Kesimpulan
Jika memilih menu pelatihan pada halaman utama.	✓	Informasi, tampil form Pelatihan untuk *Admin* dan *User*
Screen Shot		



Tabel 4 3. *Black Box* Form Daftar Pelatihan

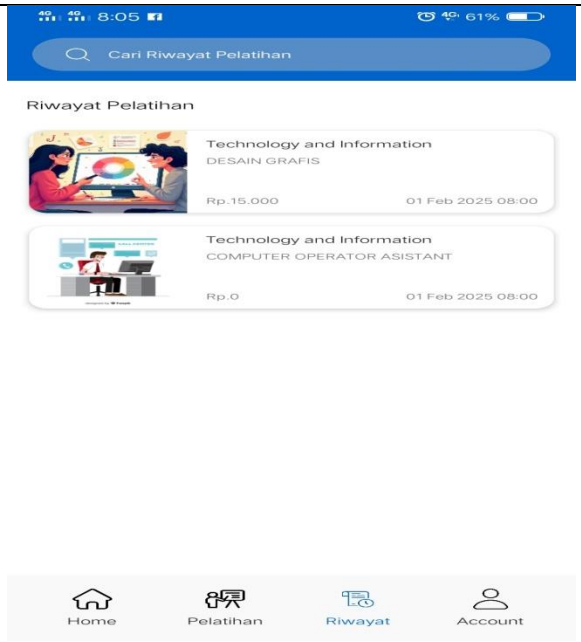
Test Faktor	Hasil	Kesimpulan
Jika memilih menu Daftar Pelatihan pada halaman utama.	✓	Informasi, tampil form Daftar Pelatihan
Screen Shot		



Tabel 4 23. *Black Box* Pendaftaran

Test Faktor	Hasil	Kesimpulan
Jika memilihmenu Pendaftaran pada halaman utama *Admin*	✓	Informasi, tampil form Pendaftaran
Screen Shot		
		

Tabel 4 4. *Black Box* Form Riwayat

Test Faktor	Hasil	Kesimpulan
Jika memilih menu riwayat pada halaman utama *User*	✓	Informasi, tampil form riwayat
Screen Shot		
		

BAB V

KESIMPULAN DAN SARAN

A. Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan serta proses pengembangan aplikasi, maka dapat disimpulkan bahwa:

1. Aplikasi Perizinan Lembaga Pelatihan Kerja Kota Parepare yang dirancang mampu memfasilitasi proses perizinan secara digital di Dinas Penanaman Modal dan Pelayanan Terpadu Satu Pintu (DPMPTSP) Kota Parepare, khususnya dalam hal pengelolaan data pemohon izin dan non-izin lembaga pelatihan kerja.
2. Aplikasi ini menyediakan fitur-fitur utama seperti formulir pendaftaran pemohon, informasi persyaratan perizinan dan non-perizinan, laporan data pemohon, serta penyimpanan dan pengelolaan data secara terpusat. Hal ini membantu mempermudah pelayanan kepada masyarakat dan meningkatkan efisiensi kerja di lingkup dinas terkait.
3. Dengan adanya sistem ini, proses pengarsipan data menjadi lebih tertata, akurat, dan mudah diakses oleh petugas dinas. Selain itu, aplikasi ini juga memberikan kemudahan kepada pemohon dalam mengakses informasi secara mandiri tanpa harus datang langsung ke kantor.

B. Saran

Untuk pengembangan lebih lanjut, berikut adalah beberapa saran yang dapat dipertimbangkan:

1. Untuk Masyarakat/Pemohon Izin: Diharapkan masyarakat dapat memanfaatkan aplikasi ini sebagai sarana dalam mengakses informasi serta melakukan proses permohonan perizinan dengan lebih mudah, cepat, dan efisien, tanpa harus melakukan proses secara manual.
2. Untuk Peneliti Selanjutnya: Penelitian ini masih memiliki keterbatasan, khususnya dalam pengembangan fitur-fitur yang lebih kompleks seperti integrasi tanda tangan digital atau notifikasi otomatis. Oleh karena itu, diharapkan peneliti selanjutnya dapat mengembangkan aplikasi ini dengan teknologi yang lebih mutakhir serta memPerluas cakupan sistem agar lebih kom*Pre*hensif.

DAFTAR PUSTAKA

Abdul Kahfi Assidiq, (2020) dalam kamus Biologi, laboratorium, Jakatra: MediaKompitindo

Adi nugroho (2010), "*Unified Modelling Language* Jakatra: MediaKompitindo

Ahmad Sidqi (2020) program studi teknik informatika – program sarjana fakultas teknologi industry Universitas islam indonesia, Melakukan Penelitian Yang Berjudul "Sistem *Monitoring* Hafalan Al-Qur'an Santri Mts Pondok Pesantren Sunan Pandanaran"

Alan Nur Aditya, 2011, Internet *Protocol* (IP) dan Transmission Control *Protocol* (TCP*)*

Alexander F. K, Sibero. 2011. Kitab Suci *Web* Progreming. MediaKom.Yogyakarta. Bahra, Al. 2013. Analisis dan Desain Sistem Informasi. Yogyakarta: Graha Ilmu.

Fulka Juhdan Hanief (2020), Fakultas Teknik dan Ilmu Komputer unikom dalam penelitiannya yang berjudul "Rancang Bangun Sistem Informasi *Monitoring* Kegiatan Santri Pondok Pesantren Islam Uswatun Hasanah Purwakarta"

Jogiyanto, H.M. 2005. Analisis dan Desain Sistem Informasi: Pendekatan Terstruktur teori dan Praktik Aplikasi Bisnis. Andi. Yogyakarta.

Madcoms Madiun, 2012, HTML *(Hyper Text Markup Language)* editor profesional, Jakarta: Raja GrafindoPersada,

Nugroho, Bunafit. 2014. Pemrograman *Web*:Membuat Sistem Informasi Akademik Sekolah dengan PHP-*MySQL* dan *Dreamweaver*. Yogyakarta: Gava Media.

Ramadhan Arif. 2006. Pemrograman *Web Database* dengan PHP dan *MySQL*. Jakarta: PT Elex Media Komputindo.

Ridwan Ridwan, Cut Mutia (2021), Fakultas Ilmu Komputer, Universitas Mercu Buana, dalam penelitiannya yang berjudul "Pengembangan Aplikasi *Web* dan Mobile Pada *Monitoring* dan Evaluasi Program Tahfidz Quran "

*Pres*man, Roger.S. 2012. Rekayasa Perangkat Lunak. Andi. Yogyakarta.

Santoso, Harip. 2010. *Aplikasi Web/asp.net + cd*. Jakart: Elex MediaKompitindo.

Sukamto, Rosa Ariani. 2009. *Analisa dan Desain Sistem Informasi*.Bandung: Informatika.