

BAB I

PENDAHULUAN

A. Latar Belakang

Data *Tracer* Alumni merupakan salah satu contoh dari suatu sistem informasi yang di rancang untuk dapat membantu pekerjaan dari suatu instansi pendidikan khususnya diperguruan tinggi dalam mengolah data sampai memberikan data secara lengkap lewat tersedianya layanan informasi. Data-data tersebut meliputi nama, angkatan, tempat tinggal dan beberapa tambahan data lainnya. *Tracer* Alumni juga memiliki peranan penting untuk pengembangan institusi pendidikan. Karena *Tracer* Alumni bisa berperan sebagai cermin kualitas sebuah institusi pendidikan diranah publik. Dengan adanya informasi tentang *Tracer* Alumni yang baik secara tidak langsung dapat meningkatkan citra baik universitas terhadap masyarakat umum. Sampai saat ini masih banyak universitas yang belum memanfaatkan penggunaan sistem informasi dalam pendataan dan penelusuran *Tracer* Alumni, yang diharapkan mempermudah dalam hal

pengolahan dan pengelolaan data *Tracer* Alumni tersebut. Untuk mempermudah memperoleh informasi data *Tracer* Alumni Universitas Muhammadiyah Parepare, diperlukan suatu sistem informasi yang dapat menginformasikan data *Tracer* Alumni.

Sistem informasi ini berfungsi untuk melihat data *Tracer* Alumni Universitas Muhammadiyah Parepare berdasarkan data yang diinput oleh masing-masing alumnus Universitas Muhammadiyah Parepare.

B. Rumusan Masalah

Dari latar belakang masalah yang ditemukan diatas maka pokok permasalahan yang terdapat dalam penelitian ini adalah Sebagai Berikut:

1. Bagaimana membuat suatu aplikasi data *Tracer* Alumni ?
2. Bagaimana cara mengetahui data *Tracer* Alumni khusus Universitas Muhammadiyah Parepare?

C. Batasan Masalah

Batasan masalah dalam penelitian tugas akhir ini adalah sebagai berikut :

1. Aplikasi yang dirancang pada aplikasi Website.
2. Data *Tracer* Alumni hanya di Universitas Muhammadiyah Parepare

D. Tujuan Penelitian

Adapun tujuan yang ingin dicapai pada proposal penelitian ini yaitu merancang dan mengimplementasikan suatu Sistem Informasi yang dapat melakukan proses manajemen data *Tracer* Alumni yang terdapat di Universitas Muhammadiyah Parepare.

E. Manfaat Penelitian

1. Manfaat bagi mahasiswa

Sebagai penerapan teori yang telah di dapatkan dari bangku kuliah dan diharapkan dapat menambah pengetahuan dan wawasan tentang bagaimana cara membuat aplikasi pada website.

2. Manfaat bagi penulis

Menambah pengetahuan, pengalaman, serta wawasan tentang metode perkembangan dan melatih diri dalam menyelesaikan masalah untuk maju dan berkembang.

F. Sistematika Penelitian

Secara garis besar sistematika penulisan yang dapat disajikan sebagai berikut:

BAB I PENDAHULUAN

Bab ini berisi latar belakang masalah, rumusan masalah, tujuan dan manfaat, serta sistematika penulis.

BAB II TINJAUAN PUSTAKA

Bab ini menguraikan tentang teori-teori dasar dan komponen-komponen yang digunakan dalam penelitian ini.

BAB III METODE PENELITIAN

Bab ini diuraikan mengenai jenis penelitian, lokasi dan waktu penelitian, parameter pengujian, tahap-tahap metode penelitian, dan bagan alir penelitian.

BAB IV HASIL PENGUJIAN

Bab ini membahas tentang pengujian alat yang terhubung pada Mikrokontroller dan pengambilan data.

BAB V PENUTUP

Bab ini membahas tentang kesimpulan hasil dan pengujian yang telah dilakukan, serta saran bagi penulis guna untuk memperbaiki kesalahan terhadap perencanaan yang telah dilakukan.

BAB II

TINJAUAN PUSTAKA

A. Tinjauan Penelitian Sebelumnya

Penelitian relevan berisikan penelitian yang terdahulu kemudian dicari bahan penelitian yang ada dan relevan dengan penelitian ini karena sangat berguna untuk mendukung kajian teori yang dikemukakan, sehingga dapat digunakan sebagai landasan pada penyusunan kerangka pikir. Beberapa hasil penelitian yang relevan dengan penelitian ini adalah penelitian yang dilakukan oleh:

1. Peneliti Takwir Satria (2013) dengan judul penelitian “Aplikasi data mahasiswa berbasis Visual Basic 6.0”. Metode yang digunakan adalah dengan pembuatan program menggunakan Visual Basic.net 2006, dan Microsoft Access yang berguna untuk mengolah berbagai jenis data dan pengoperasian yang mudah
2. Leonardi dan Hermawan (2019) dalam penelitiannya yang berjudul "Penerapan *Augmented Reality* Pada Brosur Interior Berbasis *Multy Markerless* Pada *Android*" manfaat dari penerapan *Augmented Reality* pada brosur interior adalah pelanggan dapat melihat dengan jelas barang-barang yang ingin di beli karena gambar yang ada dalam brosur bisa dilihat dalam bentuk 3 dimensi. Dalam penelitian ini, pengembangan aplikasi menggunakan metode *Rational Unified Process (RUP)*. Hasil yang didapat dari penelitian ini adalah dapat membuat aplikasi *Android* untuk brosur

interior yang dapat menampilkan barang-barang interior seperti meja, kursi dan lain-lain. Pada pembuatan gambar 3 dimensi peneliti tidak membuat objek 3 dimensinya karena objek yang berupa barang-barang interior telah dibuat oleh CV Wijaya Lesmana Sejahtera, sehingga penulis tidak harus membuat model 3D lagi tetapi tinggal di *export* dari *archicad*.

3. Nadira, Sujaini dan Pratiwi (2016) dalam penelitiannya yang berjudul “Implementasi *Augmented Reality* Pada Brosur Teknik Informatika Universitas Tanjungpura Menggunakan Metode *Marker* ” saat ini brosur Teknik Informatika Universitas Tanjungpura masih berisi gambar-gambar dari kampus itu sendiri, tetapi gambar-gambar tersebut masih kurang memberikan informasi yang jelas sehingga calon mahasiswa sulit untuk membayangkan gambaran dari kampus Teknik Informatika Universitas Tanjungpura dan akhirnya calon mahasiswa tidak terlalu tertarik untuk memilih Teknik Informatika Universitas Tanjungpura.
4. Manfaat dari penerapan *android* pada brosur Teknik Informatika Universitas Tanjungpura yaitu dapat memberikan gambaran yang jelas karena dilengkapi dengan gambar bangunan kampus dalam bentuk 3 dimensi. Dalam penelitian ini menggunakan metode *Marker Based Tracking*. Hasil yang didapat dari penelitian ini adalah peneliti menguji aplikasi yang dibuat dan hasilnya berjalan lancar pada *handphone android* dan pengguna menyatakan setuju bahwa aplikasi *Augmented Relaity* brosur dapat memperjelas informasi pada brosur Teknik Informatika Universitas Tanjungpura.

Hermawan dan Hariadi (2015) dalam penelitiannya yang berjudul "Pemanfaatan data alumni Sebagai Media Informasi Kampus Menggunakan Brosur" Manfaat dari penerapan *Augmented Reality* pada brosur kampus yaitu pembaca dapat melihat bagian brosur *Augmented Reality* dengan detail serta lingkungan pada brosur juga akan terasa lebih hidup dengan adanya animasi pendukung seperti gedung, pohon, dan lain sebagainya. Metode penelitian yang akan digunakan dalam pembuatan brosur berbasis *Augmented Reality* yaitu terdiri dari empat proses yaitu perencanaan, pengumpulan data, analisis dan dokumentasi hasil yang telah dibuat. Hasil yang didapat dari penelitian ini adalah aplikasi berjalan lancar dan telah diuji menggunakan komputer dan dapat menampilkan objek 3 dimensi pada brosur yang dapat memberikan informasi yang belum ada pada brosur sebelumnya.

B. Tracer Study

Tracer Study atau yang sering disebut sebagai survei alumni atau survei "follow up" adalah studi mengenai lulusan lembaga penyelenggara pendidikan tinggi. Studi ini mampu menyediakan berbagai informasi yang bermanfaat bagi kepentingan evaluasi hasil pendidikan tinggi dan selanjutnya dapat digunakan untuk penyempurnaan dan penjaminan kualitas lembaga pendidikan tinggi yang bersangkutan. *Tracer Study* juga bermanfaat dalam menyediakan informasi penting mengenai hubungan antara pendidikan tinggi dan dunia kerja profesional, menilai relevansi pendidikan tinggi, informasi bagi pemangku kepentingan (*stakeholders*), dan kelengkapan persyaratan bagi akreditasi pendidikan tinggi.

Perguruan tinggi perlu melaksanakan *Tracer Study* karena membutuhkan umpan balik dari alumni dalam usahanya untuk perbaikan sistem dan pengelolaan pendidikan. Perguruan tinggi di awal tahun ajaran menentukan arah kebijakan pendidikan tinggi dari masukan berupa kondisi, pengalaman, dan motivasi mahasiswa baru yang masuk ke perguruan tinggi tersebut. Masukan mengenai kondisi, pengalaman dan motivasi ini menentukan pula perguruan tinggi dalam menerapkan sistem dan pengelolaan pendidikan dalam hal pola/proses pengajaran dan pembelajaran, penelitian, praktikum, *workshop*, laboratorium, studio ataupun riset. Penerapan sistem pengajaran dan pembelajaran inipun akan dipengaruhi pula oleh kebijakan pendidikan yang ditetapkan oleh perguruan tinggi.

Hasil dari masukan berupa kondisi, pengalaman dan motivasi mahasiswa, sistem dan kebijakan pendidikan di perguruan tinggi, dan proses pengajaran dan pembelajaran di perguruan tinggi akan membantu dalam membentuk karakter/kompetensi dari lulusan perguruan tinggi itu sendiri. Lulusan/alumni dari perguruan tinggi umumnya akan memiliki pengetahuan, kemampuan, motivasi dan kompetensi yang dibutuhkan untuk memasuki dunia kerja.

Hasil dari pendidikan tinggi adalah pengetahuan, kemampuan dan kompetensi alumni perguruan tinggi yang dibutuhkan untuk memasuki dunia kerja. Hasil-hasil ini beserta kondisi saat alumni menjalani pekerjaan di awal karir mereka merupakan hal-hal yang dibutuhkan bagi perguruan tinggi untuk perbaikan sistem dan pengelolaan pendidikan. Kebutuhan untuk mengetahui

rekam jejak alumni serta hubungan pendidikan tinggi dengan pekerjaan inilah yang menjadi konsep dasar dalam penelitian Tracer Study.

Pada tahun 2016 Politeknik Negeri Bandung telah melakukan *Tracer Study* untuk mahasiswa yang lulus pada tahun 2014. *Tracer study* ini dilakukan secara online. Penyebaran informasi kepada alumni tentang alamat tautan untuk pengisian data dilakukan melalui email dan sms. Selanjutnya, alumni membuka alamat tautan tersebut, mengisi data yang diminta dan mengirimkan kuesioner yang telah diisi. Kuesioner berisi pertanyaan tentang data pribadi, penelusuran pekerjaan pertama dan saat ini, serta pertanyaan-pertanyaan seputar proses belajar mengajar semasa kuliah.

C. Pengujian Black Box

Metode pengujian adalah salah satu cara atau metode untuk menguji perangkat lunak dan data kemungkinan terjadi kesalahan. Menurut Yakub (2012: 150) pengujian *Black Box* merupakan pendekatan komplementer dari teknik *White Box testing*, karena pengujian *Black Box* mampu mengungkap kesalahan yang lebih luas. Pengujian *Black Box* berfokus pada pengujian persyaratan fungsional perangkat lunak, karena untuk mendapat serangkaian kondisi *input* yang sesuai dengan persyaratan fungsional suatu program.

Menurut Ronger S.Pressman (2007:551) dalam buku rekayasa perangkat lunak yang menyatakan "pengujian *Black Box* berfokus pada persyaratan fungsional perangkat lunak." Dengan demikian, pengujian *Black Box* memungkinkan perekayasa perangkat lunak mendapatkan serangkaian kondisi input yang sepenuhnya menggunakan semua persyaratan fungsional untuk suatu

program. Pengujian *Black Box* bukan merupakan alternative dari teknik *white box*. tetapi merupakan pendekatan komplementer yang kemungkinan besar mampu mengungkap *class* kesalahan daripada metode *white box*. Pengujian *Black Box* berusaha menemukan kesalahan dalam kategori sebagai berikut :

1. Fungsi-fungsi yang tidak benar atau hilang
2. Kesalahan *interface*
3. Kesalahan dalam struktur data atau akses data / *wave eksternal*
4. Kesalahan kinerja
5. Inisialisasi dan kesalahan terminasi.

D. Pengujian *White Box*

Untuk menguji sistem yang telah dibuat, penulis menggunakan metode pengujian *white box testing* dan *test case*. Metode ini bertujuan untuk mendapatkan ukuran kekompleksan logikal dari perancangan prosedural program. Untuk menghitung tingkat kompleksitas logika program maka digunakan metode *Cyclomatic Complexity* (CC).

Untuk menguji suatu sistem maka bagan alir program (*flowchart*) yang didesain sebelumnya dipetakan ke dalam bentuk bagan alir kontrol (*flowgraph*) yang nantinya memudahkan untuk penentuan jumlah *region*, *Cyclomatic Complexity* (CC) dan *Independent Path*, jika jumlah *region*, *Cyclomatic Complexity* (CC) dan *Independent Path* sama besar maka sistem dinyatakan benar, tetapi jika sebaliknya maka sistem masih memiliki kesalahan, mungkin dari segi logika maupun dari sisi lainnya.

Cyclomatic Complexity (CC) dapat dihitung dengan menggunakan

rumus :

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

Dimana:

E = jumlah *edge* pada *flowgraph*

N = Jumlah *node* pada *flowgraph*

Adapun istilah-istilah penting yang perlu diperhatikan pada saat pembuatan *flowgraph* yaitu:

1. *Node*

Yaitu lingkaran pada *flowgraph* yang menggambarkan satu atau lebih perintah prosedural. Urutan proses dan keputusan dapat dipetakan dalam satu *node*.

2. *Edge*

Yaitu tanda panah yang menggambarkan aliran kontrol. Setiap *node* harus mempunyai tujuan *node*.

3. *Region*

Yaitu daerah yang dibatasi oleh *node* dan *edge*. Untuk menghitung *region* daerah diluar *flowgraph* juga harus dihitung.

4. *Predicate Node*

Yaitu kondisi yang terdapat pada *node* dan mempunyai karakteristik dua atau lebih *edge* darinya.

5. *Cyclomatic Complexcity*

Yaitu *metric* perangkat lunak yang menyajikan ukuran kuantitatif dari kekompleksan logikal suatu program. *Cyclomatic Complexcity* digunakan untuk mencari jumlah *path* dalam suatu *flowgraph*.

6. *Independent Path*

Yaitu jalur yang melintasi atau melalui dalam program dimana sekurang-kurangnya terdapat proses perintah yang baru atau kondisi yang baru. Dalam istilah *flowgraph*, *independent path* harus bergerak sekurang-kurangnya pada satu *edge*, dimana *edge* yang akan dilewati tidak dapat dilewati *edge* sebelum jalur tersebut didefinisikan.

Adapun manfaat yang dapat diperoleh dengan menggunakan teknik pengujian *white box* adalah antara lain sebagai berikut:

1. Dapat menjamin seluruh independent path di dalam modul yang dikerjakan sekurang-kurangnya satu kali.
2. Dapat mengerjakan seluruh keputusan logikal.
3. Dapat mengerjakan seluruh perulangan yang sesuai dengan batasannya.
4. Dapat mengerjakan seluruh struktur data internal yang menjamin validilitas.

E. Internet (*Interconnected Network*)

Internet adalah kelompok atau kumpulan dari jutaan komputer. Penggunaan internet memungkinkan kita untuk mendapatkan informasi dari komputer yang ada di dalam kelompok tersebut dengan asumsi bahwa pemilik komputer memberikan izin akses. (Janner Simarmata, 2009:47).

F. HTTP (*Hyper Text Transfer Protocol*)

Definisi dari HTTP (*HyperText Transfer Protocol*) yaitu protokol komunikasi *stateless* yang berbasiskan pada TCP yang awalnya digunakan untuk mengambil kembali file-file HTML versi 0.9, adalah protokol yang sederhana untuk komunikasi antara client dan server. (Janner Simarmata, 2009:52).

G. URL (*UniForm Resource Locator*)

URL adalah string yang dapat dibaca oleh manusia dengan struktur yang sederhana. URL berperan untuk menjelaskan bagaimana dokumen dapat diakses. (Janner Simarmata, 2009:68-69)

H. WWW (*World Wide Web*)

WWW atau yang sering disebut sebagai “web” saja merupakan aplikasi internet yang paling populer.

Sejarah web dikembangkan pertama kali oleh Sir Timothy Jhon Tim Berners-Lee, hanya saja pada saat itu web masih berjalan tanpa terhubung jaringan internet, yaitu pada akhir tahun 80-an. Saat itu laboratorium *CERN* berlokasi di kota Geneva, swiss menyatakan bahwa web bisa diakses melalui jaringan dan dimiliki oleh siapa saja.

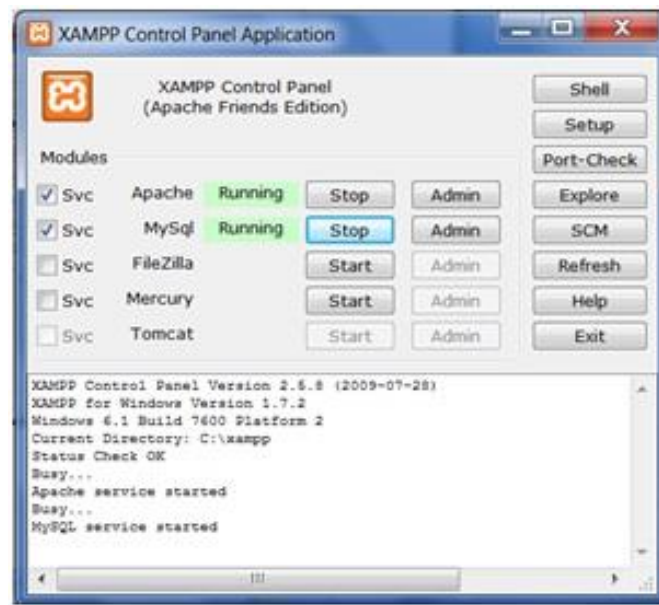
Sejarah web juga berkaitan dengan sejarah perkembangan teknologi Komputer, karena pada awalnya tampilan web masih sangatlah sederhana, hanya menampilkan teks, lalu untuk *hypertext (link)* pada saat itu masih menggunakan tampilan nomor yang berhubungan antara satu halaman ke halaman lain. Pada saat itu pun, teknologi web dikembangkan dan berjalan pada sistem operasi *Unix*, masih sangat jarang yang menggunakan teknologi *windows*. Andaiapun ada, teknologi *windows* masih sangat sederhana.

I. PHP (*Hypertext Preprocessor*)

PHP adalah akronim dari *Hypertext Preprocessor*, yaitu pengembangan untuk menempelkan kode di dalam HTML dengan menggunakan bahasa yang sama, seperti *perl* dan *UNIX shells*. Objek sumber tersusun sebagai halaman

HTML, tetapi generasi konten dinamis yang *programmatic* (Janner Simarmata, 2009:148).

J. XAMPP



Gambar 2.1 Tampilan Aplikasi XAMPP

XAMPP (/icon'zæmp/ atau /'ɛks.æmp/[1]) adalah perangkat lunak bebas, yang mendukung banyak sistem operasi, merupakan kompilasi dari beberapa program.

Fungsinya adalah sebagai server yang berdiri sendiri (*localhost*), yang terdiri atas program Apache HTTP Server, MySQL *database*, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan Perl. Nama XAMPP merupakan singkatan dari X (empat sistem operasi apapun), Apache, MySQL, PHP dan Perl. Program ini tersedia dalam GNU *General Public License* dan bebas, merupakan web server yang mudah digunakan yang dapat melayani tampilan halaman web yang dinamis. Untuk mendapatkannya dapat mengunduh langsung dari web resminya.(Ali Ibrahim,2008)

K. MySQL

MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL (bahasa Inggris: *database management system*) atau DBMS yang *multithread*, *multi-user*, dengan sekitar 6 juta instalasi di seluruh dunia. MySQL AB membuat MySQL tersedia sebagai perangkat lunak gratis di bawah lisensi GNU *General Public License* (GPL), tetapi mereka juga menjual di bawah lisensi komersial untuk kasus-kasus di mana penggunaannya tidak cocok dengan penggunaan GPL.

Tidak sama dengan proyek-proyek seperti Apache, di mana perangkat lunak dikembangkan oleh komunitas umum, dan hak cipta untuk kode sumber dimiliki oleh penulisnya masing-masing, MySQL dimiliki dan disponsori oleh sebuah perusahaan komersial Swedia MySQL AB, di mana memegang hak cipta hampir atas semua kodenya. Kedua orang Swedia dan satu orang Finlandia yang mendirikan MySQL AB adalah: David Axmark, Allan Larsson, dan Michael "Monty" Widenius. (Andrea Adelheid dan Khairil Nst,2013)

L. Hosting

Setelah selesai mendesain dan membuat website secara *offline*, langkah selanjutnya adalah mengunggah website tersebut ke *account* hostingan yang telah didapatkan. Yang dinamakan dengan mengunggah adalah proses memindahkan file – file dari komputer ke server hostingan sehingga nantinya bisa diakses oleh banyak orang di seluruh dunia. Untuk teknik mengunggah website ada beberapa cara, ada 2 cara yang umum dalam melakukan unggah file – file website ke server hosting :

1. Unggah menggunakan file manager

Keuntungan mengunggah file menggunakan file manager adalah tidak perlu menginstal apapun server, cukup menggunakan browser dan mengakses Cpanel saja. Sedangkan kelemahannya adalah folder tidak bisa ikut diunggah, sehingga membuat folder secara manual. Selain itu, bisa dibilang bahwa proses unggah dilakukan satu – persatu. *File manager* hanya bisa mengunggah 15 file sekaligus dan berada di folder yang sama.

2. Unggah menggunakan software FTP

Unggah menggunakan *software* FTP cukup mudah dan menyenangkan. Selain bisa mengunggah banyak file sekaligus (termasuk folder – foldernya), ketika koneksi terputus dan proses unggah terhenti, itu bisa di lanjutkan sewaktu – waktu. Selain itu, Unggah menggunakan FTP lebih bisa diandalkan ketika harus mengunggah file – file dalam ukuran besar. *Software* FTP yang banyak digunakan antara lain adalah fileZilla dan CuteFTP. (Andy Krisianto,2012)

M. 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. *Behaviorial 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 diperlukan penggambaran paket yang menyederhanakan model. Paket-paket ini kemudian dapat didekomposisi lebih lanjut. Paket berguna bagi pengelompokan sesuatu, misalnya model-model serta subsistem-subsistem.
4. *An notational things*, merupakan bagian yang memperjelas 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 (*dependency*) adalah hubungan dimana perubahan yang terjadi pada suatu elemen independent akan mempengaruhi elemen yang bergantung padanya.

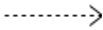
2. Asosiasi adalah apa dan bagaimana yang menghubungkan antara objek satu dengan yang lainnya. Suatu bentuk asosiasi adalah agregasi yang menampilkan hubungan suatu objek dengan bagian-bagiannya.
 3. Generalisasi adalah hubungan dimana objek anak berbagai perilaku dan struktur data dari objek yang ada di atasnya (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 bangun. *Use case* menggambarkan fungsionalitas sistem atau persyaratan-persyaratan yang harus dipenuhi sistem dari pandangan pemakai

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

Tabel 2.1 Simbol *Use Case Diagram*

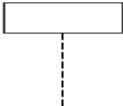
NO	GAMBAR	NAMA	KETERANGAN
1		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .

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

Tabel 2.2 Simbol *Class Diagram*

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

Tabel 2.3 Simbol *Sequence Diagram*

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

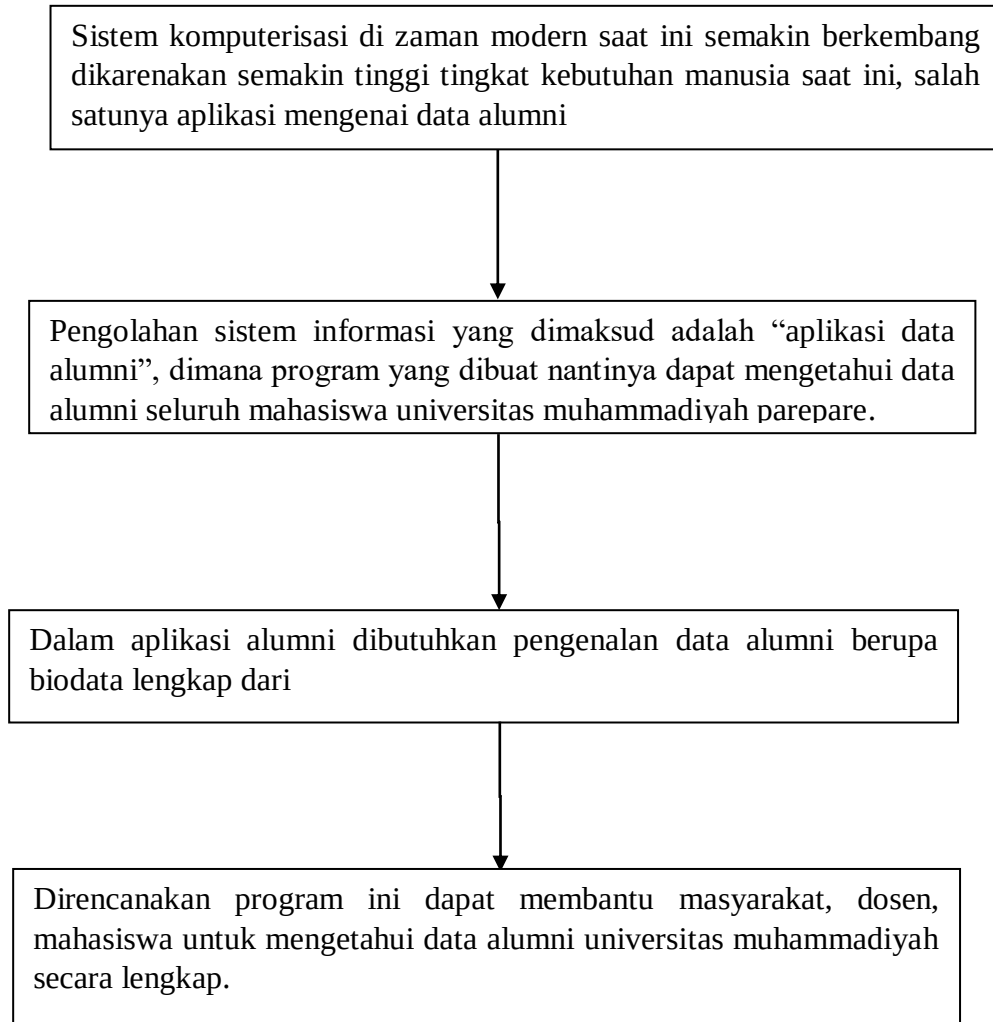
Tabel 2.4 Simbol *State Chart Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		<i>State</i>	Nilai atribut dan nilai <i>link</i> pada suatu waktu tertentu, yang dimiliki oleh suatu objek.
2		<i>Initial Pseudo State</i>	Bagaimana objek dibentuk atau diawali
3		<i>Final State</i>	Bagaimana objek dibentuk dan dihancurkan
4		<i>Transition</i>	Sebuah kejadian yang memicu sebuah <i>state</i> objek dengan cara memperbaharui satu atau lebih nilai atributnya
5		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.

Tabel 2.5 Simbol *Activity Diagram*

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

N. Kerangka Pikir



BAB III

METODE PENELITIAN

A. Jenis Penelitian

Jenis penelitian yang digunakan adalah penelitian deskriptif dimana memberikan gambaran mengenai apa yang sesungguhnya terjadi. Dalam pembuatan Skripsi ini digunakan metode deskripsif yang menggambarkan fakta-fakta dan informasi secara sistematis, faktual dan akurat.

Penelitian ini dilakukan melalui internet yang dapat memberikan sumber data dan pengetahuan mengenai sistem yang diteliti, kemudian mencocokkan dengan kemungkinan yang terjadi dalam usaha penyelesaian masalah.

B. Waktu dan Tempat Penelitian

Adapun waktu dan tempat penelitian yang dilaksanakan pada bulan Oktober tahun 2022, di Universitas Muhammadiyah Parepare, dimana proses pengambilan data alumni yang akan dimasukkan dalam aplikasi nantinya

NO	Uraian Kegiatan	Bulan 2022				
		Oktober	Oktober	November	November	Desember
1	Studi Literatur					
2	Perancangan aplikasi					
3	Pembuatan aplikasi					
4	Pengujian					

C. Metode Pengumpulan Data

Untuk memperoleh data-data yang dibutuhkan dalam rangka melakukan penelitian, maka penulis mengumpulkan data melalui beberapa cara yaitu :

1. Analisis Data

Menganalisa data-data yang sebelumnya telah dikumpulkan.

2. Perancangan Program

Sebagai pedoman dalam penulisan program atau kode-kode agar berjalan sesuai rencana.

3. Uji Coba Program

Pengujian program dilakukan untuk memastikan bahwa program yang dibuat dapat berjalan dengan baik.

4. Evaluasi

Sistem yang telah selesai dibangun perlu adanya evaluasi untuk menemukan kelemahan yang terdapat pada program yang telah dibangun tadi, yang nantinya bisa digunakan sebagai acuan untuk memperbaiki program sehingga lebih sempurna.

D. Jenis Data

Data yang digunakan dalam penelitian ini berupa data-data yang telah dikumpulkan melalui Penelitian Pustaka (*Library Research*). Adapun jenis data primer dan data sekunder yang relevan dengan masalah yang akan dibahas.

a. Data primer

Data Primer adalah data yang berasal atau data yang diperoleh langsung dari sumber data dan pengetahuan.

b. Data Sekunder

Data sekunder adalah data yang diperoleh tidak secara langsung dari objek penelitian. Peneliti mendapatkan data yang sudah jadi dari internet, website dan jurnal.

E. Tahapan Penelitian

Tahapan penelitian yang dimaksud dalam penelitian ini ada beberapa tahapan yaitu persiapan penelitian, pengumpulan data, analisis perancangan, pengujian dan implementasi. Adapun Uraian dari tahapan tersebut adalah sebagai berikut :

1. Persiapan Penelitian

Pada tahapan ini peneliti melakukan persiapan penelitian. Persiapan penelitian yang dimaksud adalah menyiapkan buku-buku, artikel-artikel tentang topik penelitian serta *software* yang digunakan selama penelitian.

2. Pengumpulan Data

Pada tahap ini peneliti melakukan observasi dengan peninjauan, pencatatan dan pengamatan langsung di tempat penelitian.

3. Analisis

Pada tahap analisis, peneliti melakukan analisa terhadap sistem yang di terapkan sekarang berdasarkan kemudian merumuskan masalah yang menjadi pokok penelitian sehingga dapat dibuat alternatif pemecahan masalah.

4. Perancangan

Peneliti kemudian merancang aplikasi yang ingin dibuat berdasarkan alternatif pemecahan masalah.

5. Pengujian

Setelah melakukan perancangan, peneliti kemudian menguji hasil perancangan yang telah dibuat. Jika hasil perancangan terdapat kekurangan atau kelemahan maka kembali ke tahap analisis.

6. Implementasi

Setelah pada perancangan tidak terdapat kekurangan maka aplikasi siap untuk digunakan oleh *user*.

F. Metode Pengujian Aplikasi

Beberapa *test-case* harus dilaksanakan dengan beberapa perbedaan strategi, *query*, atau jalur navigasi yang mewakili penggunaan sistem yang *tipical*, kritis atau abnormal. Isu kunci pada pengembangan sistem adalah pemilihan sekelompok *test-case* yang cocok, sekecil dan secepat mungkin, untuk meyakinkan perilaku sistem secara detail. Pengujian harus mencakup *unit testing*, yang mengecek validasi dari prosedur dan fungsi-fungsi secara independen dari komponen sistem yang lain. Kemudian modul testing harus menyusul dilakukan untuk mengetahui apakah penggabungan beberapa unit dalam satu modul sudah berjalan dengan baik, termasuk eksekusi dari beberapa modul yang saling berelasi, apakah sudah berjalan sesuai karakteristik sistem yang diinginkan.

Jika struktur kendali antar modul sudah terbukti bagus, maka pengujian yang tak kalah pentingnya adalah pengujian unit. Pengujian unit digunakan untuk menguji setiap modul untuk menjamin setiap modul menjalankan fungsinya dengan baik. Ada 2 metode untuk melakukan *unit testing*, yaitu :

1. White Box Testing

Uji coba *white box testing* merupakan metode perancangan *testcase* yang menggunakan struktural untuk mendapatkan *testcase*, test ini digunakan untuk meramal cara kerja perangkat lunak secara rinci kepada *logic path* (jalur logika), perangkat lunak di tes dengan kondisi dan perulangan secara fisik.

Contoh pengujian *white box testing* ini merupakan peringatan ketika user memasukkan *password user* yang salah, untuk kesalahan semacam ini akan memberikan suatu informasi kepada *user* mengenai kesalahan yang di lakukan.

2. *Black Box*

Berdasarkan hasil uji coba yang dilakukan, seluruh navigasi dan tombol fasilitas program lainnya serta proses yang di jalankan tidak terjadi kesalahan, tetapi aplikasi mempunyai aturan-aturan yang sudah di tetapkan dan harus di ikuti karena apabila di hiraukan maka sistem akan menolak perintah yang tidak sesuai seperti kesalahan ketika *user* belum menginput data yang harusnya di input sesuai ketentuan sistem yang di jalankan dan sistem memberikan informasi kepada *user* karena data yang ingin diproses belum lengkap atau tidak memenuhi ketentuan untuk proses selanjutnya.

G. Metode Pengujian

Metode pengujian yang dilakukan pada penelitian ini menggunakan *blackbox* yang dipakai untuk mengetahui kinerja aplikasi pada saat melakukan proses *Tour Guide* dan mengetahui berfungsinya semua aplikasi yang telah dibuat, *blackbox* dalam metode pengujian perangkat lunak yang tes fungsionalitas dari aplikasi yang bertentangan dengan struktur internal atau kerja.

BAB IV

ANALISIS DAN PERANCANGAN

A. Analisis Sistem

Analisis dan perancangan sistem ialah tahapan pengumpulan kebutuhan data dan informasi dan kemudian diolah sehingga dapat di definisikan kebutuhan yang harus dipenuhi oleh sistem yang akan dibangun. Jenis data yang dikumpulkan selama proses pengambilan data berupa data sekunder. Data sekunder merupakan berupa data yang di peroleh langsung oleh *user* yang mendaftar pada aplikasi kami dengan cara memperkenalkan aplikasi kami pada grup grup diskusi yang secara online, baik grup *whatshap*, *facebook* dan lain lain.

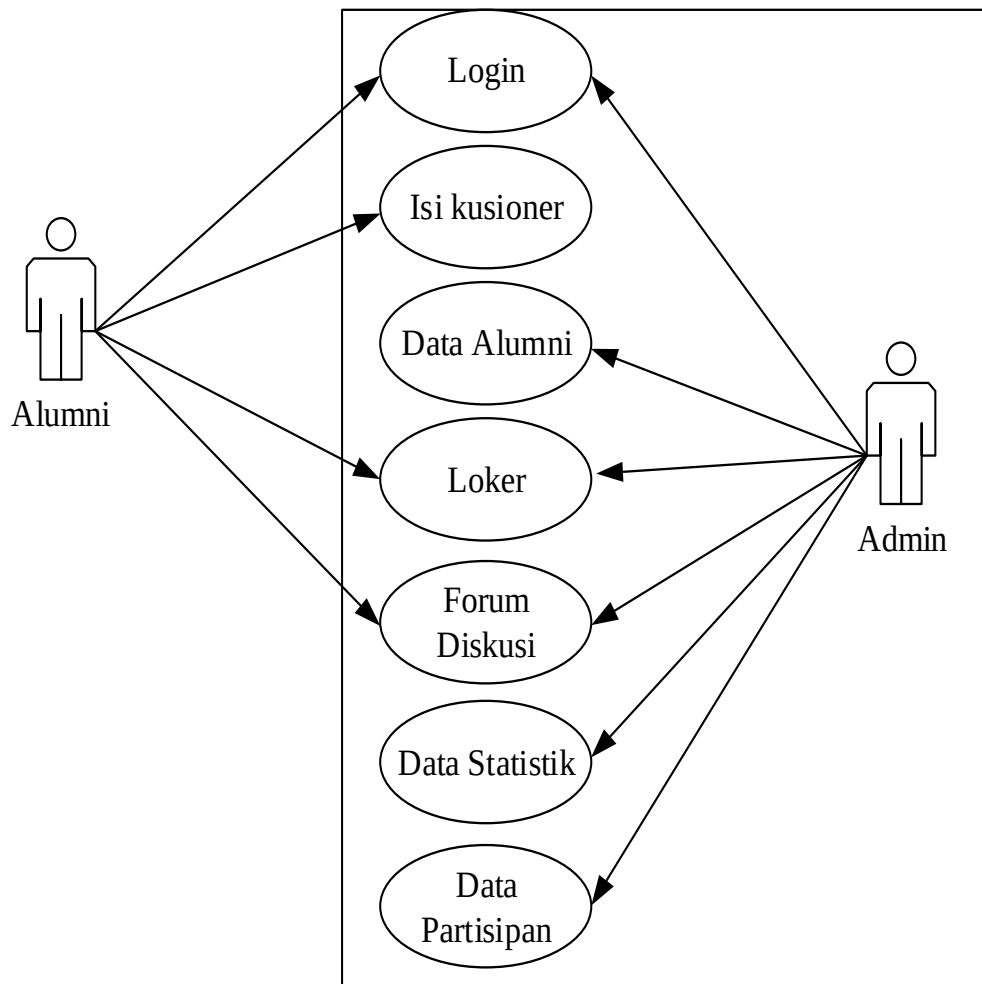
B. Perancangan Sistem

Perancangan Sistem adalah pendefinisian dari kebutuhan-kebutuhan fungsional dan persiapan untuk rancang bangun implementasi yang menggambarkan bagaimana suatu sistem dibentuk. Adapun kegiatan dalam tahapan ini antara lain :

1. *Use case diagram*

Use Case Diagram adalah suatu bentuk diagram yang menggambarkan fungsionalitas yang diharapkan dari sebuah sistem dilihat dari perspektif pengguna di luar sistem. Sebuah *use case diagram* merepresentasikan interaksi yang terjadi antara *user* dengan proses atau sistem yang dibuat. *Use Case diagram* menggambarkan kegiatan yang dilakukan pada *user* pada sistem yang akan

dibangun sehingga *use case diagram* dari sistem yang dibangun dapat dilihat pada gambar berikut ini :



Gambar 4.1 *Use Case Diagram*

Pada gambar 4.1, dapat dijelaskan bahwa *User* dapat melakukan *Login* diri di aplikasi dan mengisi kusioner yang telah disiapkan, setelah itu admin dapat melakukan mengecek data alumni dan data statistik.

2. Tabel *Use Case*

Penjelasan *Use Case Diagram* di atas :

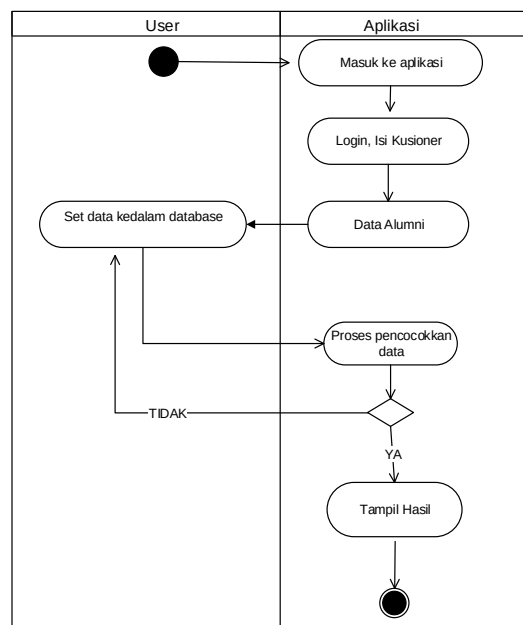
- a. Pengguna

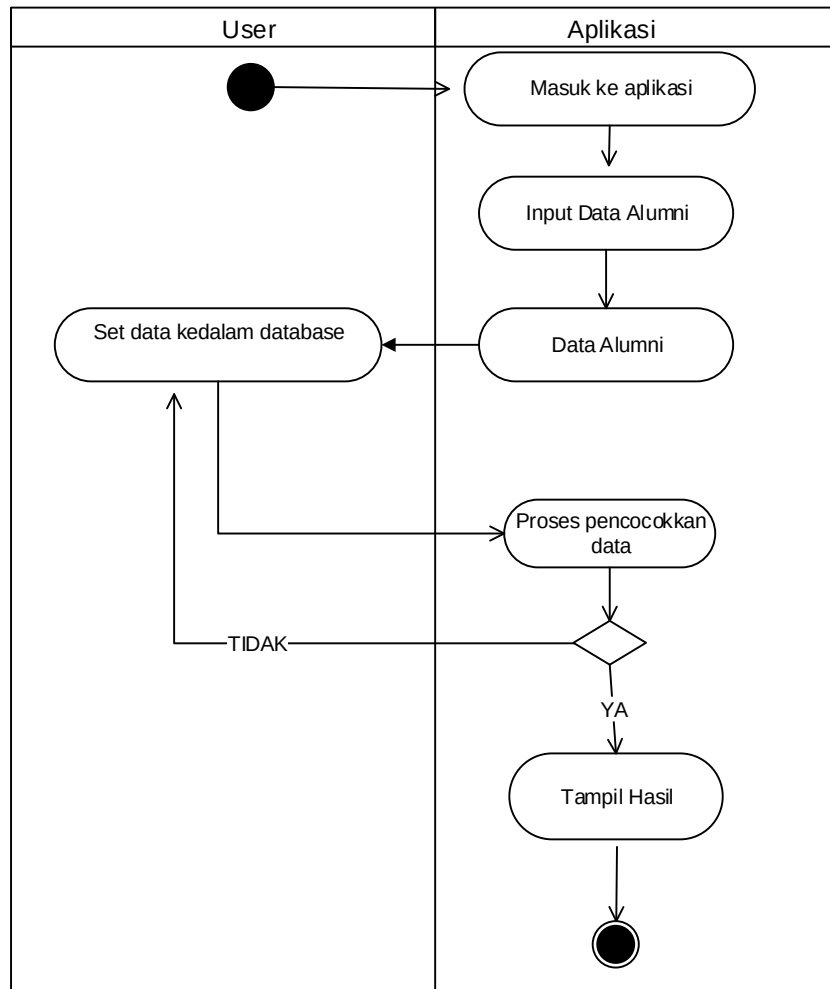
Tabel 4.1 Tabel *Use Case*

Nama <i>Use Case</i>	Deskripsi <i>Use Case</i>
Login	<i>Use Case</i> ini menjelaskan bahwa setelah <i>user</i> mengaktifkan aplikasi, akan terbuka tampilan awal sistem.
Isi Kusioner	<i>Use Case</i> ini menjelaskan bahwa setelah <i>user</i> mengaktifkan aplikasi, akan terbuka <i>form</i> mengisi kusioner yang disiapkan aplikasi.
Data Alumni	<i>Use Case</i> ini menjelaskan bahwa setelah <i>user</i> mengisi kusioner maka admin dapat mengecek data alumni tersebut didalam aplikasi.
Data Statistik	<i>Use Case</i> ini menjelaskan bahwa setelah <i>user</i> mengisi kusioner maka admin dapat mengecek data Statistik tersebut didalam aplikasi.
Data Partisipan	<i>Use Case</i> ini menjelaskan bahwa setelah <i>user</i> mengisi kusioner maka admin dapat mengecek data Partisipan tersebut di dalam aplikasi.

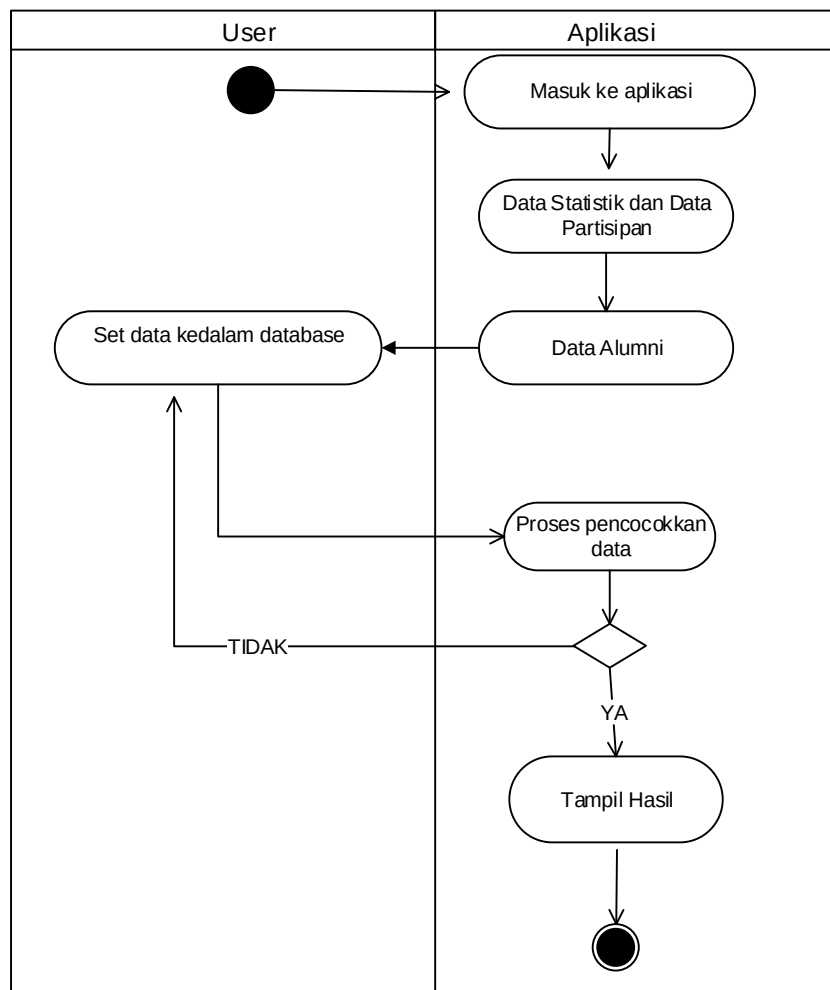
1. Activity Diagram

a. Activity Diagram User

**Gambar 4.2** Activity Diagram user

b. *Activity Diagram Input data alumni***Gambar 4.3** *Activity Diagram data alumni*

c. *Activity Diagram* data Statistik dan partisipan

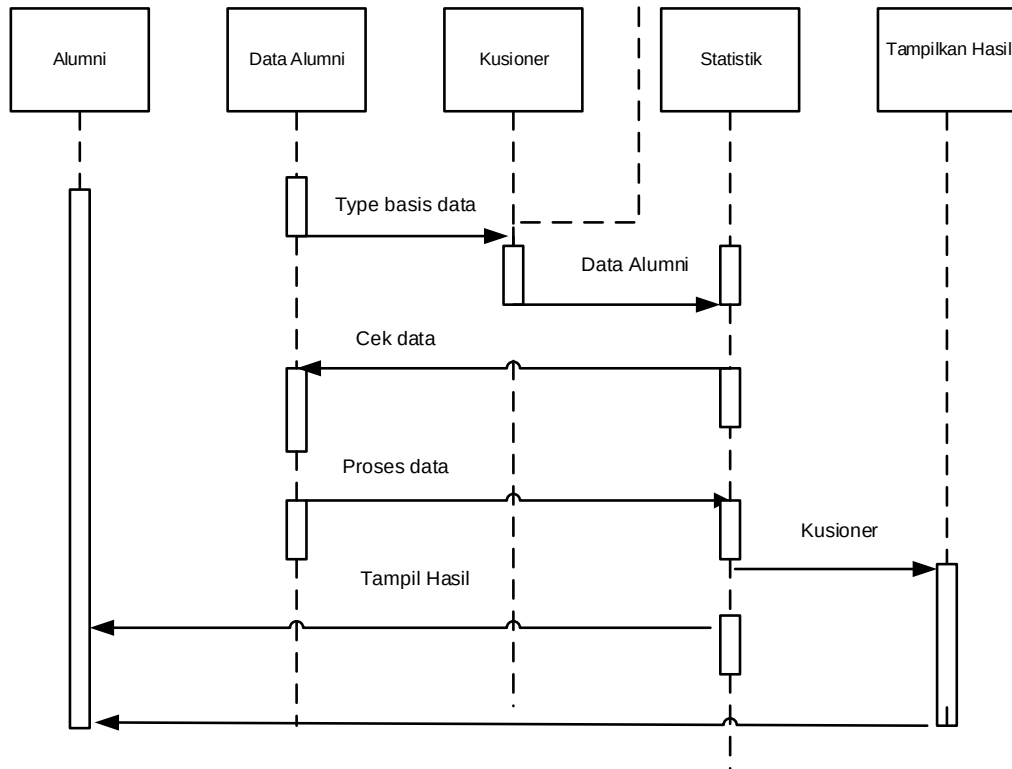


Gambar 4.4 *Activity Diagram* Statistik dan partisipan

Activity diagram di atas menjelaskan bahwa ketika *user* mulai masuk ke aplikasi kemudian menginput data alumni dan melakukan proses pengisian kusioner dan dapat melihat hasil statistik dan partisipan.

2. Sequence Diagram

a. Sequence Diagram Aplikasi

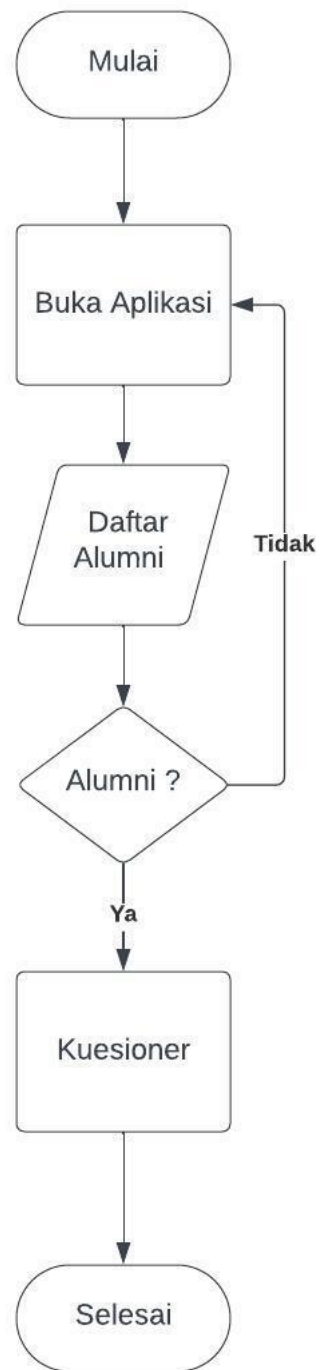


Gambar 4.5 Sequence Diagram

Sequence diagram di atas menjelaskan bahwa ketika alumni mulai masuk ke aplikasi kemudian melakukan login dan melakukan pengisian kusioner pada sistem tersebut kemudian admin dapat melihat data data alumni dan statistik tersebut.

3. Rancangan Flowchart

Flowchart merupakan bagan yang menunjukkan alur kerja atau apa yang sedang dikerjakan di dalam sistem secara keseluruhan dan menjelaskan urutan dari prosedur-prosedur yang ada di dalam alur penelitian. Pada flowchart di bawah ini akan menjelaskan mengenai cara aplikasi.



Gambar 4.6 *Flowchart*

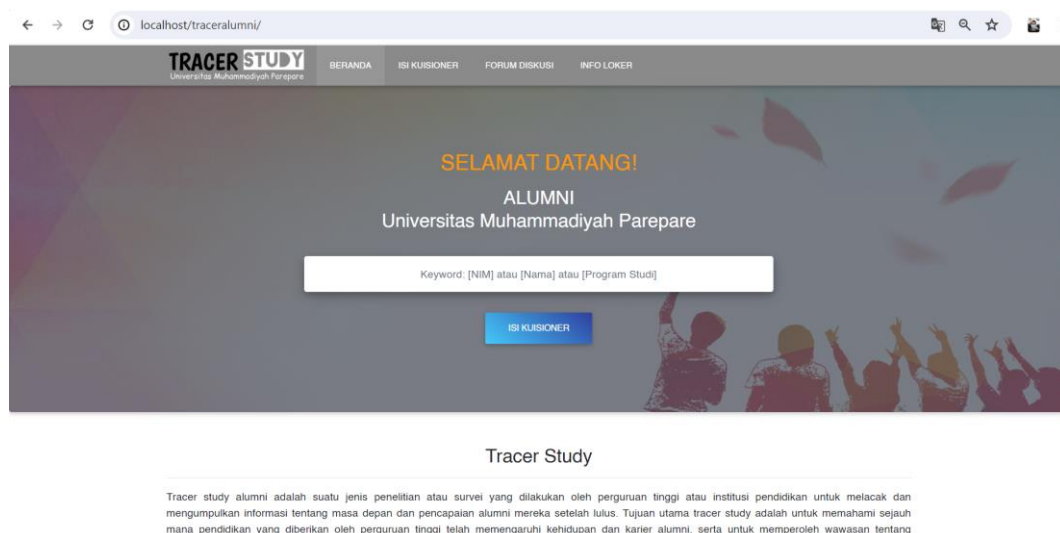
C. Implementasi

Program ini adalah aplikasi yang mempunyai fungsi untuk mempermudah seseorang untuk memperoleh data diri dari hasil aplikasi.

Implementasi aplikasi ini dilakukan setelah proses pendaftaran diri alumni diaplikasi selesai dilakukan. Untuk menjalankan aplikasi ini maka dibutuhkan sebuah perangkat *smartphone* atau *android*.

1. Tampilan Menu utama

Ini adalah tampilan menu utama pada aplikasi sebagai berikut;



Gambar 4.7 Tampilan Menu Utama

Pada Menu Utama menampilkan beberapa menu yaitu ; Beranda, Isi Kuesioner, Forum Diskusi dan juga Info Loker.

2. Tampilan Isi Kusioner

Setelah menekan Tombol isi kusioner di bawah maka *user* akan diarahkan ke *form* pengisian kusioner.

← → ↻ localhost/traceralumni/isi_kuisiener

TRACER STUDY
Universitas Muhammadiyah Parepare

BERANDA ISI KUISIENER FORUM DISKUSI INFO LOKER

Isi Kuisiener

1. Kami ingin memahami lebih baik tentang pengalaman Anda setelah lulus. Mohon bantu kami dengan menjawab beberapa pertanyaan.
2. Kuisiener ini bertujuan untuk membantu kami memahami sejauh mana pendidikan yang Anda terima di institusi kami telah memengaruhi karier Anda.
3. Silakan berikan pandangan Anda tentang pekerjaan, pendidikan lanjutan, dan kontribusi Anda setelah lulus.

Masukkan NIM Masukkan Nama Masukkan Tanggal Lahir

-- Pilih Fakultas -- -- Pilih Program Studi --

SELANJUTNYA

© Copyright 2023 Tracer Study - UMPAR

Gambar 4.8 Tampilan kuesioner

User diarahkan untuk mengisi data diri berupa Nim, Nama, Tanggal Lahir, Fakultas dan Program Studi mereka.

3. Tampilan Halaman kuesioner

Setelah *user* melakukan pengisian kusioner maka *admin* dapat melihat data tersebut didalam aplikasi.

TRACER STUDY
Universitas Muhammadiyah Parepare

BERANDA ISI KUISIENER

Data Pribadi

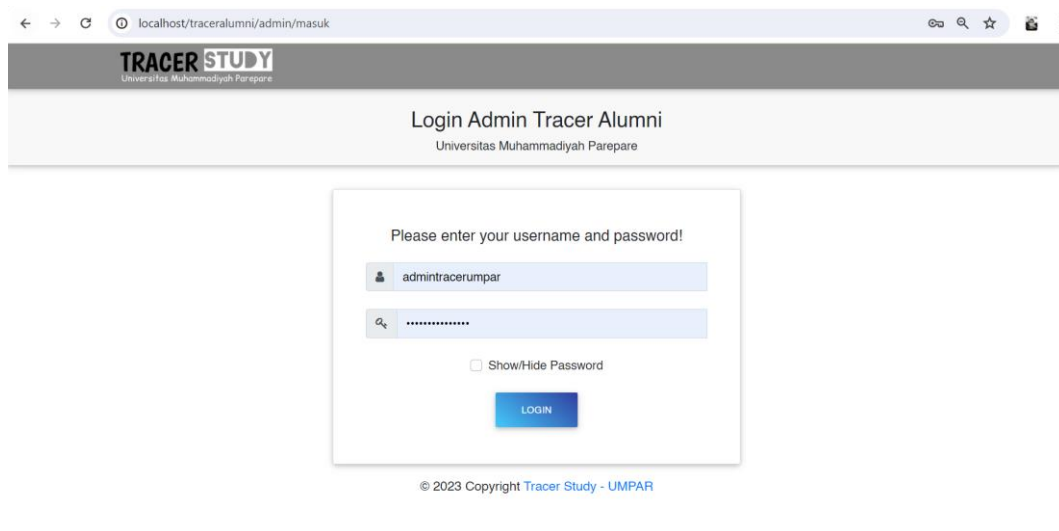
1. Nama	:	HABIBIE SALIM TADJO
2. Umur	:	26
3. Jenis Kelamin	:	Laki-laki
4. Status Perkawinan	:	Belum Menikah
5. Alamat Tempat Tinggal	:	JL. PISANG, PINRANG
6. No. HP	:	088176543121
7. Email	:	hst@gmail.com

Gambar 4.9 Tampilan kuesioner

Setelah itu halaman akan berpindah ke tampilan halaman kuesioner yang akan diisi oleh alumni dengan instrumen kuesioner yang sesuai dengan ketentuan.

4. Tampilan Login Admin

Tampilan pada Gambar di bawah, admin memasukan *username* dan *password* yang telah dibuat.



The screenshot shows a web browser window with the address bar displaying 'localhost/traceralumni/admin/masuk'. The page header features the 'TRACER STUDY' logo and 'Universitas Muhammadiyah Parepare'. The main heading is 'Login Admin Tracer Alumni' with the subtitle 'Universitas Muhammadiyah Parepare'. The login form is centered and contains the following elements:

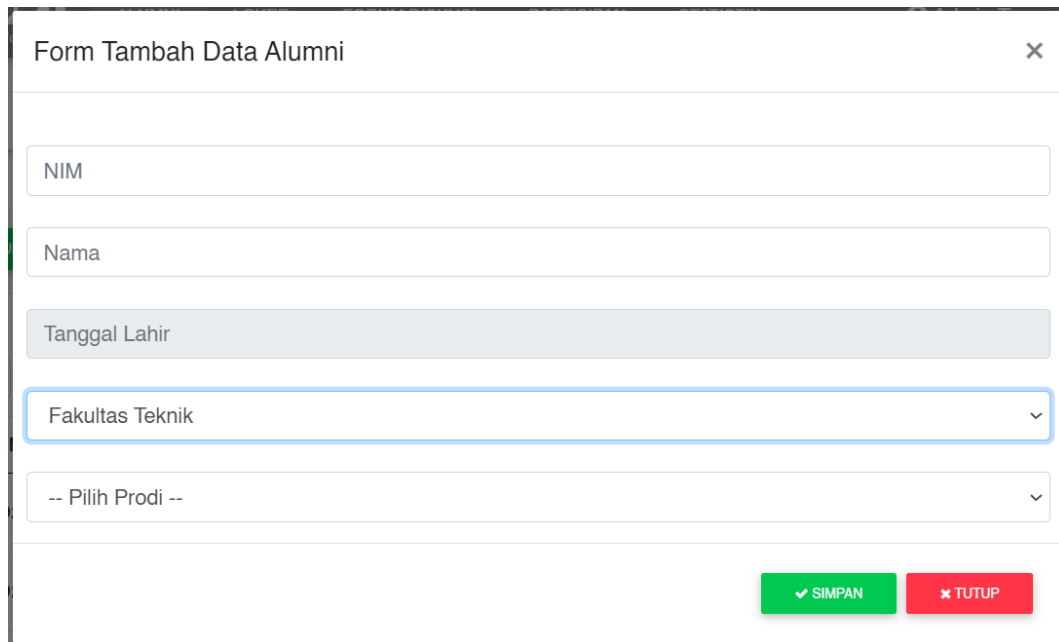
- A prompt: 'Please enter your username and password!'
- A username input field with the value 'admintracerrumpar'.
- A password input field with masked characters '.....'.
- A checkbox labeled 'Show/Hide Password'.
- A blue 'LOGIN' button.

At the bottom of the form, there is a copyright notice: '© 2023 Copyright Tracer Study - UMPAR'.

Gambar 4.10 Tampilan *Login Admin*

5. Tampilan Tambah Alumni

Tampilan pada Gambar di bawah, Admin sedang melakukan tambah data alumni dengan mengisi Nim, Nama, Fakultas dan Program Studi yang sesuai dengan alumni.



Form Tambah Data Alumni

NIM

Nama

Tanggal Lahir

Fakultas Teknik

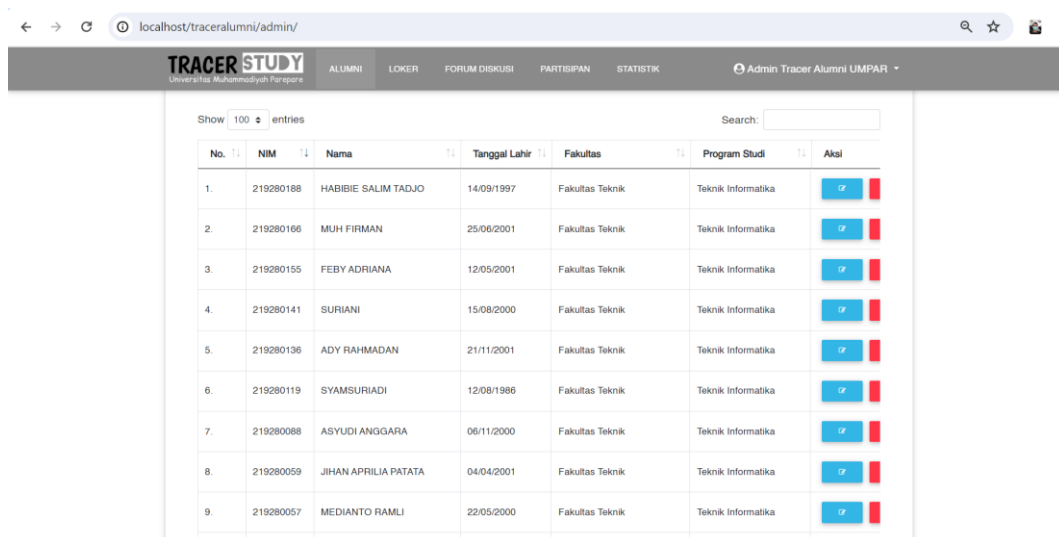
-- Pilih Prodi --

SIMPAN TUTUP

Gambar 4.11 Tampilan Tambah Alumni

6. Tampilan Data Alumni

Tampilan pada Gambar di bawah, Admin sedang menampilkan data alumni.



TRACER STUDY

ALUMNI LOKER FORUM DISKUSI PARTISIPAN STATISTIK Admin Tracer Alumni UMPAR

Show 100 entries Search:

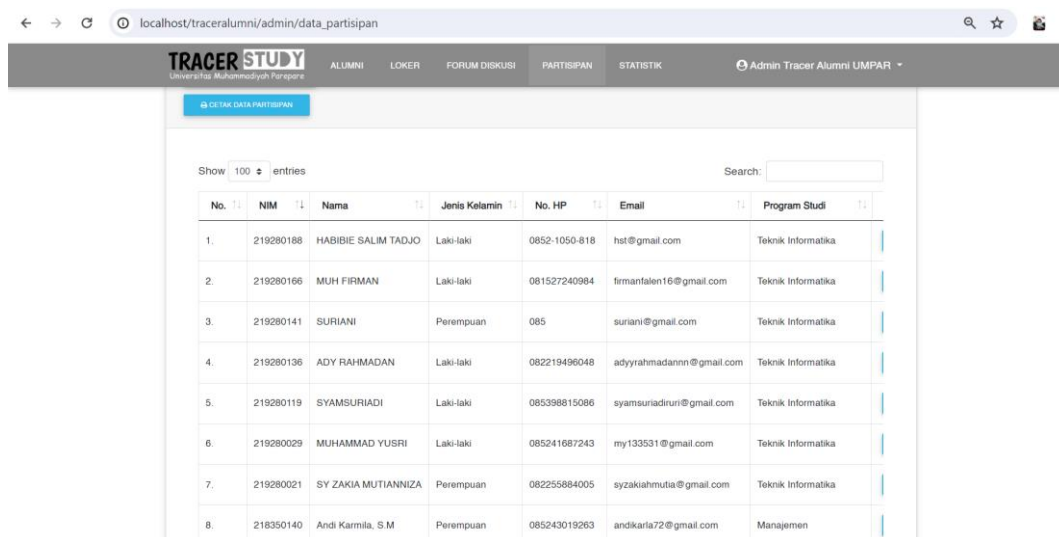
No.	NIM	Nama	Tanggal Lahir	Fakultas	Program Studi	Aksi
1.	219280188	HABIBIE SALIM TADJO	14/09/1997	Fakultas Teknik	Teknik Informatika	✕ +
2.	219280166	MUH FIRMAN	25/06/2001	Fakultas Teknik	Teknik Informatika	✕ +
3.	219280155	FEBY ADRIANA	12/05/2001	Fakultas Teknik	Teknik Informatika	✕ +
4.	219280141	SURIANI	15/08/2000	Fakultas Teknik	Teknik Informatika	✕ +
5.	219280136	ADY RAHMADAN	21/11/2001	Fakultas Teknik	Teknik Informatika	✕ +
6.	219280119	SYAMSURIADI	12/08/1986	Fakultas Teknik	Teknik Informatika	✕ +
7.	219280088	ASYUDI ANGGARA	06/11/2000	Fakultas Teknik	Teknik Informatika	✕ +
8.	219280059	JIHAN APRILIA PATATA	04/04/2001	Fakultas Teknik	Teknik Informatika	✕ +
9.	219280057	MEDIANTO RAMLI	22/05/2000	Fakultas Teknik	Teknik Informatika	✕ +

Gambar 4.12 Tampilan Data Alumni

Setelah admin menambahkan data alumni Maka akan tampil pada menu Data Alumni, Admin dapat mengubah kembali data yang dimasukan ataupun menghapus data alumni dengan memilih fitur pada kolom aksi.

7. Tampilan data Partisipan

Tampilan pada Gambar di bawah, Admin sedang menampilkan data partisipan.



localhost/traceralumni/admin/data_partisipan

TRACER STUDY
Universitas Muhammadiyah Purwokerto

ALUMNI LOKER FORUM DISKUSI PARTISIPAN STATISTIK Admin Tracer Alumni UMPAR

CEK DATA PARTISIPAN

Show 100 entries Search:

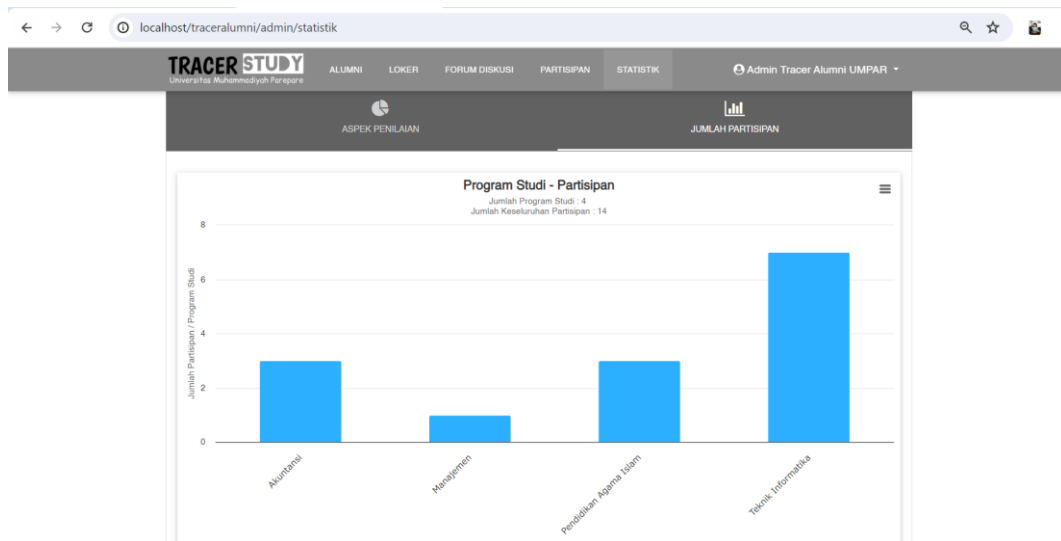
No.	NIM	Nama	Jenis Kelamin	No. HP	Email	Program Studi
1.	219280188	HABIBIE SALIM TADJO	Laki-laki	0852-1050-818	hst@gmail.com	Teknik Informatika
2.	219280166	MUH FIRMAN	Laki-laki	081527240984	firmanfalan16@gmail.com	Teknik Informatika
3.	219280141	SURIANI	Perempuan	085	suriani@gmail.com	Teknik Informatika
4.	219280136	ADY RAHMADAN	Laki-laki	082219496048	adyrahmadannn@gmail.com	Teknik Informatika
5.	219280119	SYAMSURIADI	Laki-laki	085398815086	syamsuriadiirni@gmail.com	Teknik Informatika
6.	219280029	MUHAMMAD YUSRI	Laki-laki	085241687243	my133531@gmail.com	Teknik Informatika
7.	219280021	SY ZAKIA MUTIANNIZA	Perempuan	082255884005	syzakhmutia@gmail.com	Teknik Informatika
8.	218350140	Andi Kamilia, S.M	Perempuan	085243019263	andikarla72@gmail.com	Manajemen

Gambar 4.13 Tampilan Data Partisipan

Bagi alumni yang telah mengisi kuesioner akan tampil pada menu data partisipan, Admin dapat melihat hasil kuesioner dan dapat mengunduh hasil kuesioner alumni.

8. Tampilan Data Statistik Partisipan

Tampilan pada gambar di bawah, Admin sedang menampilkan data statistik partisipan.



Gambar 4.14 Tampilan Data Statistik Partisipan

Pada menu statistik dapat dilihat grafik dari jumlah partisipan berdasarkan program studi alumni.

9. Tampilan Data Statistik Penilaian

Tampilan pada Gambar di bawah, Admin sedang menampilkan data penilaian statistik.

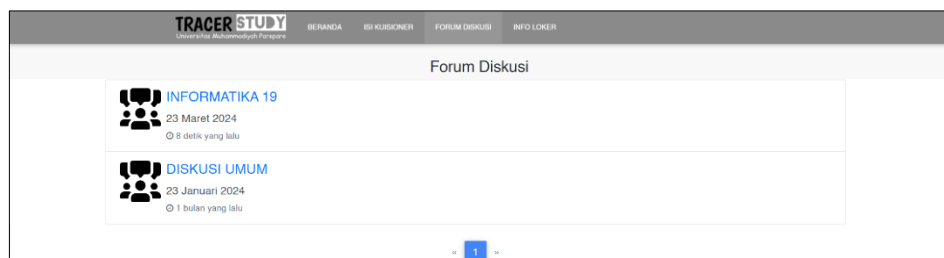
No.	Aspek Yang Dinilai	Kategori	Akumulasi (%)				
			5	4	3	2	1
1.	Kesesuaian bidang studi	Aspek-aspek Pengembangan Karir / Usaha Mandiri	21.43	50.00	21.43	-	7.14
2.	Sertifikat profesi		28.57	57.14	14.29	-	-
3.	Indeks prestasi akademik (transkrip)		28.57	57.14	14.29	-	-
4.	Keterampilan praktis yang diperoleh semasa kuliah		28.57	50.00	21.43	-	-
5.	Keterampilan praktis yang diluar bangku kuliah		28.57	64.29	7.14	-	-
6.	Reputasi almamater		28.57	57.14	14.29	-	-
7.	Kemampuan berbahasa asing		28.57	50.00	21.43	-	-
8.	Keterampilan komputer		28.57	57.14	14.29	-	-

Gambar 4.15 Tampilan Data Statistik Penilaian

Pada aspek penilaian akan ditampilkan akumulasi dari nilai instrumen yang diisi oleh alumni, Admin dapat melihat nilai instrumen keseluruhan ataupun berdasarkan program studi yang ada.

10. Tampilan Forum Diskusi

Setelah *user* melakukan pengisian kusioner maka *user* dapat melakukan diskusi tersebut di dalam aplikasi.

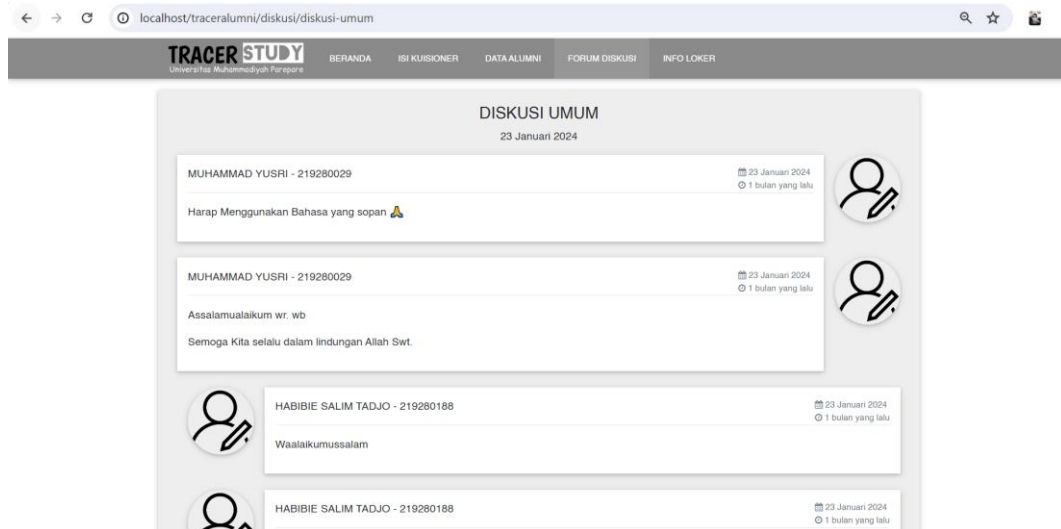


Gambar 4.16 Tampilan Forum Diskusi

Halaman ini berisikan forum diskusi yang dapat dibuat oleh para alumni dan dapat mengatur peserta yang dapat ikut dalam forum tersebut dengan memberikan judul yang sesuai dengan topik forum diskusi mereka.

11. Tampilan Diskusi Umum

Setelah *user* melakukan pengisian kusioner maka *user* dapat melihat data diskusi tersebut didalam aplikasi.

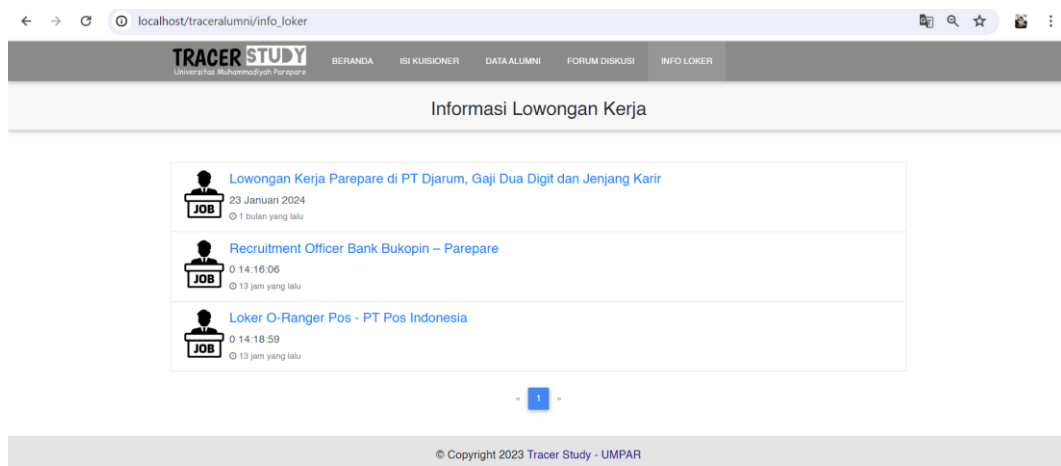


Gambar 4.17 Tampilan Forum Diskusi

Sebagai identitas dari pengirim pesan, Dicantumkan nama, nim dan tanggal kirim pesan sehingga dapat dikenali *user* yang mengirim pesan

12. Tampilan Info Loker

Setelah *user* melakukan pengisian kusioner maka *user* dapat melihat data loker tersebut di dalam aplikasi.

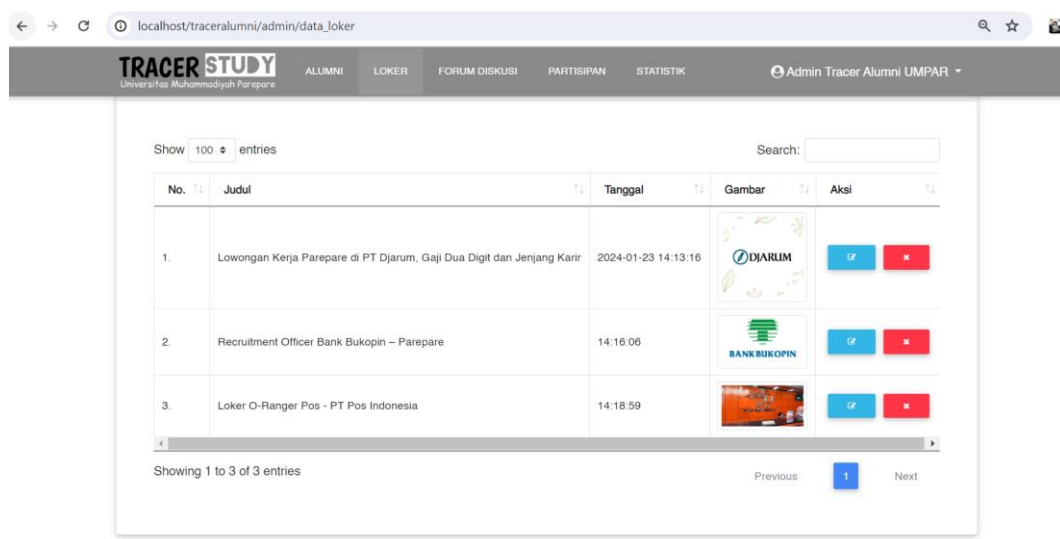


Gambar 4.18 Tampilan Info Loker

Pada Menu Utama menampilkan beberapa menu yaitu ; Beranda, Isi Kuesioner, Forum Diskusi dan juga Info Loker.

13. Tampilan Halaman Admin Loker

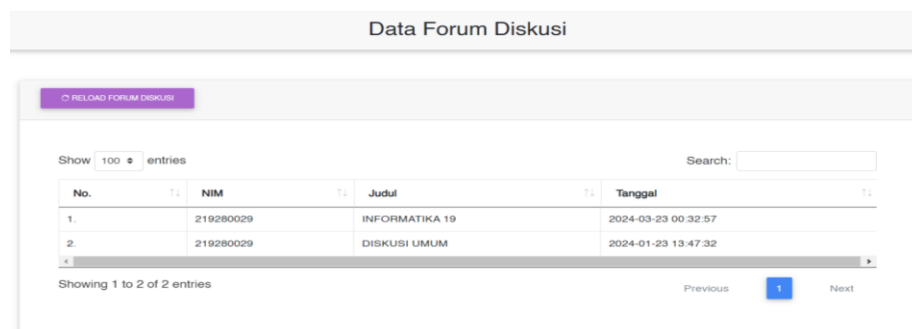
Tampilan pada Gambar di bawah, Admin sedang menampilkan data loker yang sudah dimasukan berupa judul, tanggal berlaku dan gambar dari penyedia lowongan pekerjaan.



Gambar 4.19 Tampilan Admin Loker

14. Tampilan Halaman Forum Diskusi

Tampilan pada Gambar di bawah, *Admin* sedang menampilkan data diskusi yang sudah dimasukan oleh admin dan *user*.



Gambar 4.20 Tampilan Admin Forum Diskusi

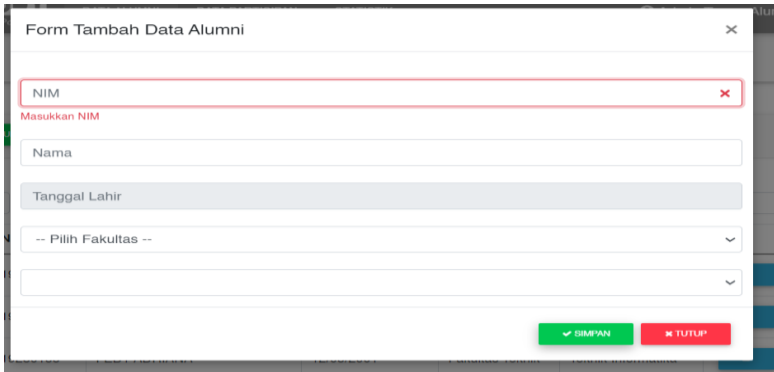
Admin dapat melihat forum yang dibuat oleh *user* dan dapat juga menutup atau menghapus forum diskusi tersebut.

D. Pengujian

Pengujian Sistem dengan Pengujian *Black box*. Pengujian ini bersifat menguji waktu perintah sesuai dengan fungsinya yang tanpa ada kesalahan atau *error*.

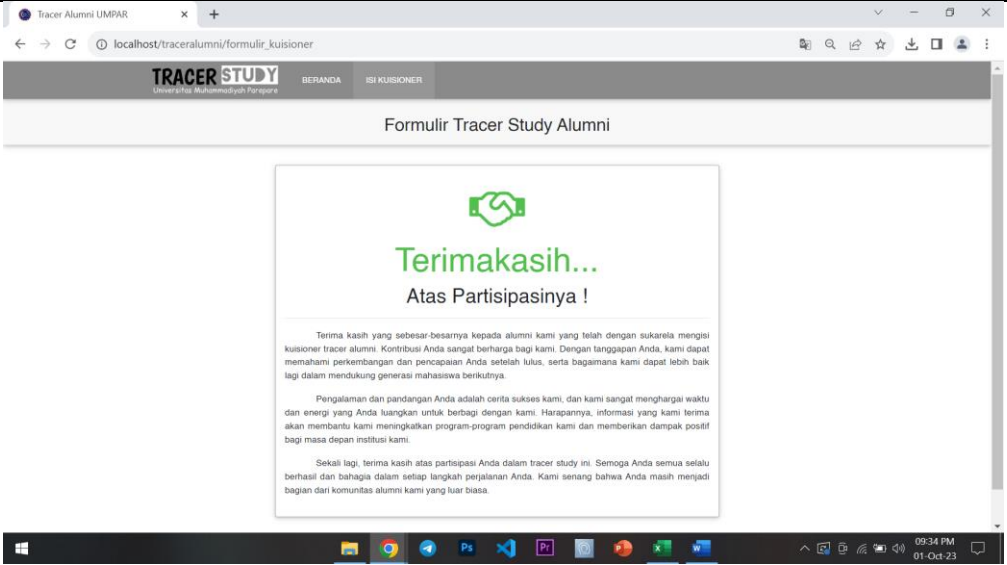
1. *Black box* melakukan *input* data alumni .

Tabel 4.2 *Black Box* Melakukan proses *input* data alumni

Test Factor	Hasil	Kesimpulan
Jika tombol tambah data alumni ditekan akan menampilkan proses pengisian tambah data alumni	✓	Berhasil karena proses data alumni dapat ditampilkan data detail pada sistem.
Screenshot		
		

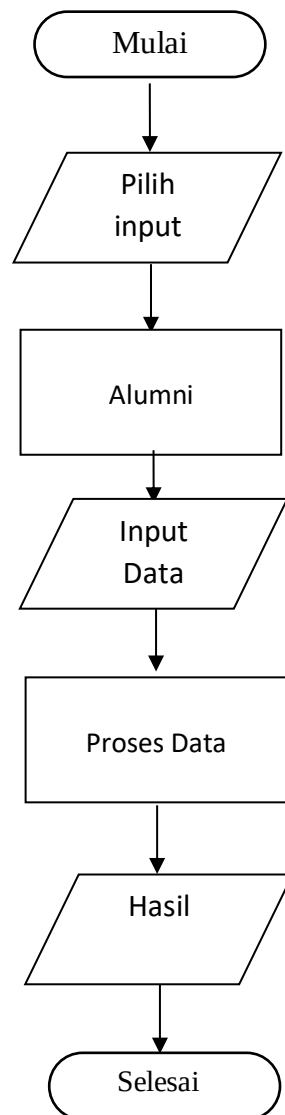
2. *Black box* isi kusioner .

Tabel 4.3 *Black Box* Melakukan proses isi kusioner

Test Factor	Hasil	Kesimpulan
Jika proses pengisian kusioner dilakukan maka system akan melakukan proses pengisian data kusiner.	✓	Berhasil, karena data kusioner dapat dikenali oleh system secara otomatis
Screenshot		
		

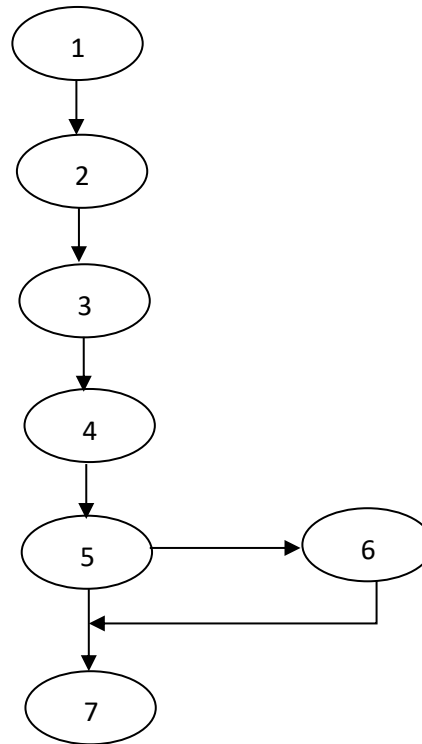
1. Pengujian *White box*

a. *Flowchart* Aplikasi



Gambar 4.21. *Flowchart* Aplikasi

b. *Flowgraph* Aplikasi



Gambar 4.22 *Flowgraph* Aplikasi

1) Proses Perhitungan

Dari Gambar *FlowGraph* di atas dapat dilakukan proses perhitungan sebagai berikut :

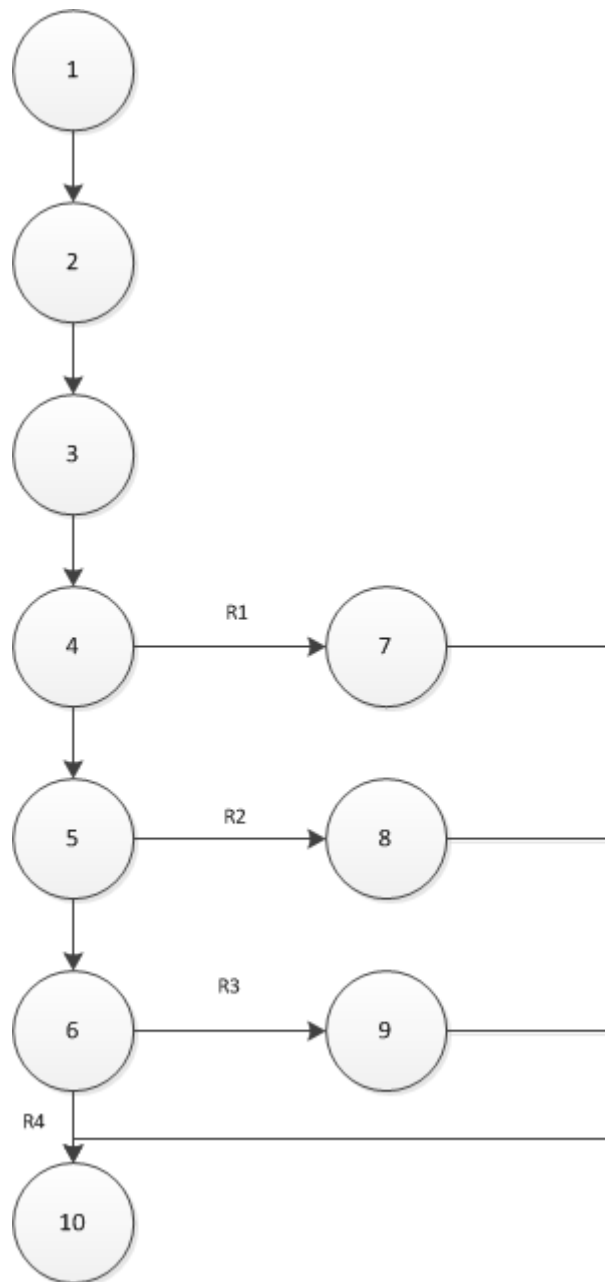
a. Menghitung *Cyclomatic Complexity* $V(G) = E - N + 2$

$$N (\text{node}) = 7$$

$$E (\text{edge}) = 7$$

$$P (\text{predikat node}) = 1$$

1) *Flowgraph* menjalankan aplikasi



Gambar 4.23 *Flowgraph* menjalankan aplikasi

2) Proses perhitungan menjalankan aplikasi

Dari gambar *flowgraph* di atas dapat dilakukan proses perhitungan sebagai berikut :

a) Menghitung *Cyclomatic Complexity* $V(G)$ dari *Edge* dan *Node*:

Dengan Rumus : $V(G) = E - N + 2$

$$N(\text{node}) = 10$$

$$E(\text{edge}) = 12$$

$$P(\text{predikat node}) = 3$$

$$\text{Penyelesaian : } V(G) = E - N + 2$$

$$= 12 - 10 + 2$$

$$= 4$$

$$\text{Predikat Node (N)} = P + 1$$

$$= 3 + 1$$

$$= 4$$

b) Berdasarkan perhitungan *Cyclomatic Complexity* dari *Flowgraph* di atas memiliki *Region* = 4

c) *Independent Path* pada *flowgraph* di atas adalah :

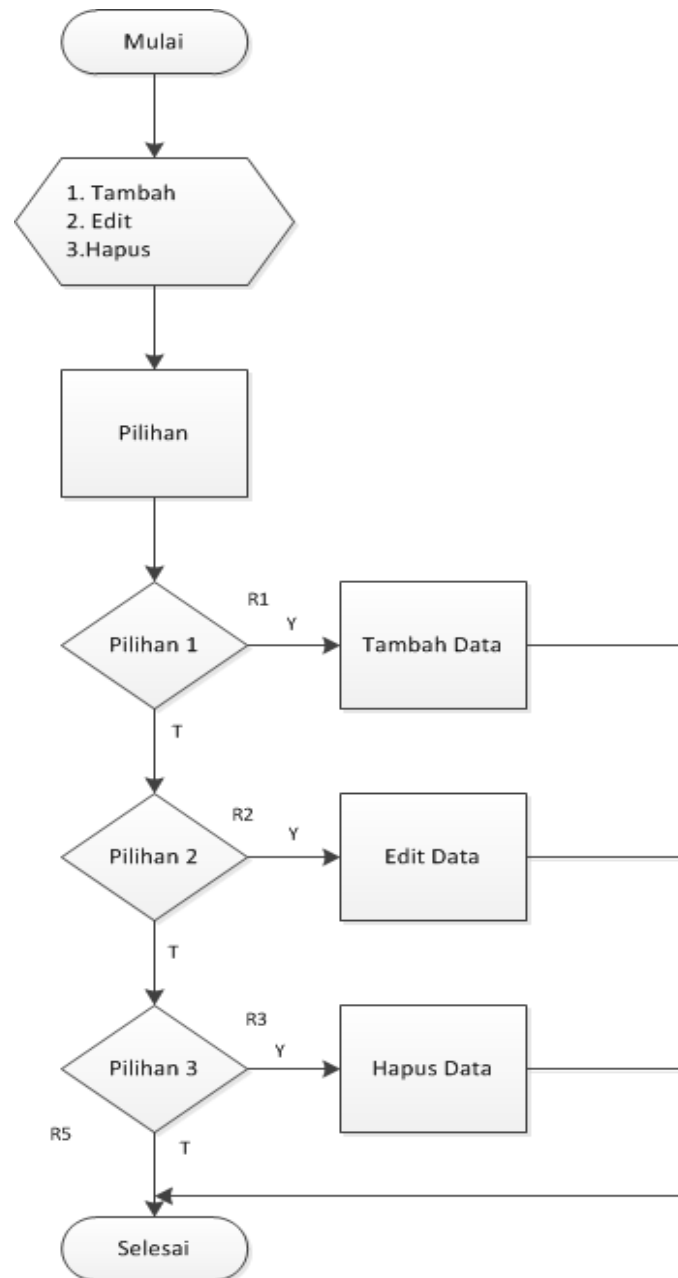
$$\text{Path 1} = 1 - 2 - 3 - 4 - 7 - 10$$

$$\text{Path 2} = 1 - 2 - 3 - 4 - 5 - 8 - 10$$

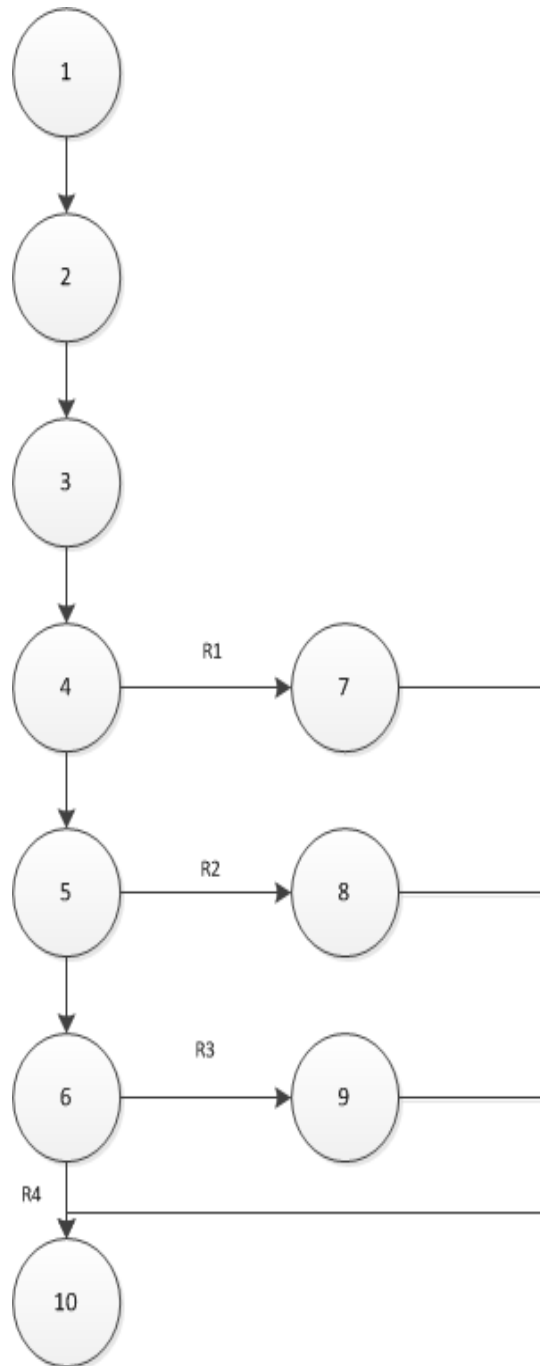
$$\text{Path 3} = 1 - 2 - 3 - 4 - 5 - 6 - 9 - 10$$

$$\text{Path 4} = 1 - 2 - 3 - 4 - 5 - 6 - 10$$

1. *Flowchart input data*



Gambar 4.24 *Flowchart input data*

1) *Flowgraph* input data**Gambar 4.25** *Flowgraph* input data

2) Proses perhitungan android data

Dari gambar *flowgraph* diatas dapat dilakukan proses perhitungan sebagai berikut :

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

Dengan Rumus : $V(G) = E - N + 2$

$$N(\text{node}) = 10$$

$$E(\text{edge}) = 12$$

$$P(\text{predikat node}) = 3$$

$$\text{Penyelesaian : } V(G) = E - N + 2$$

$$= 12 - 10 + 2 = 4$$

$$\text{Predikat Node (N)} = P + 1$$

$$= 3 + 1$$

$$= 4$$

4) Berdasarkan perhitungan *Cyclomatic Complexcity* dari *Flowgraph* di atas memiliki *Region* = 4

5) *Independent Path* pada *flowgraph* di atas adalah :

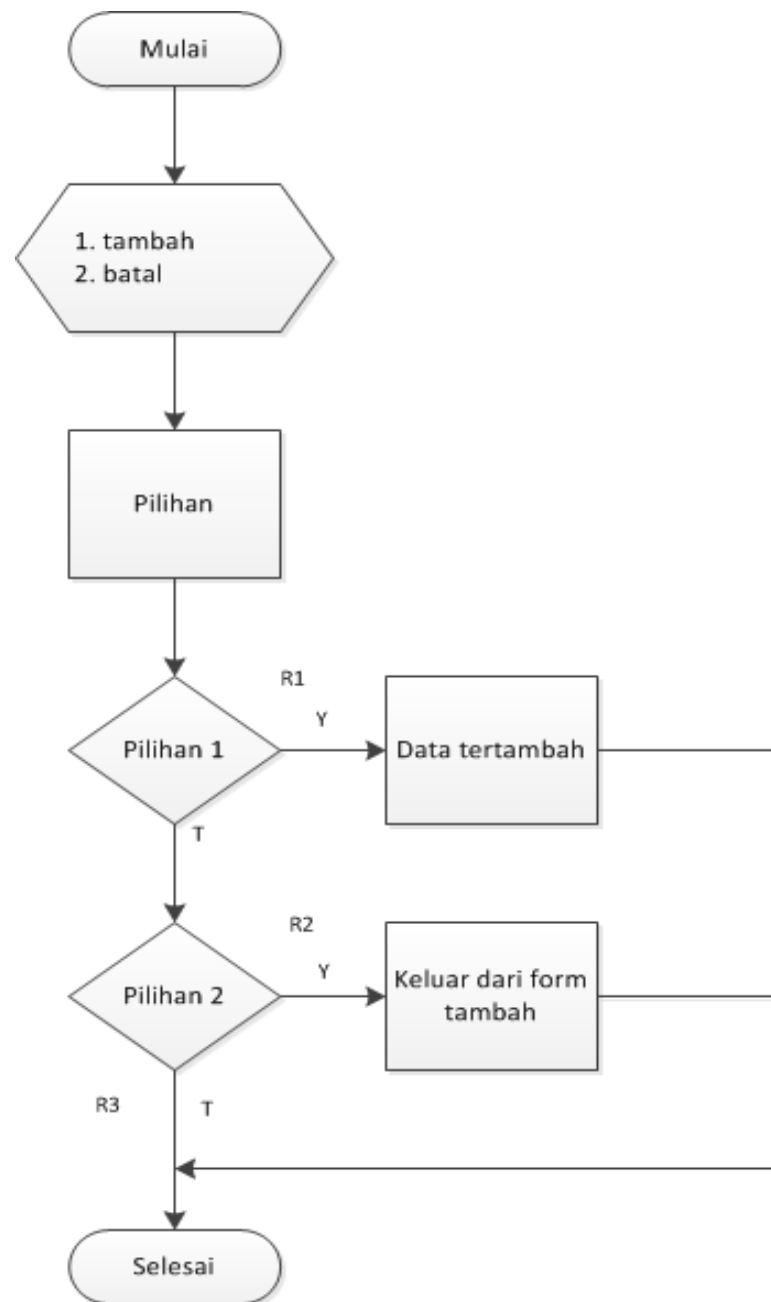
$$\text{Path 1} = 1 - 2 - 3 - 4 - 7 - 10$$

$$\text{Path 2} = 1 - 2 - 3 - 4 - 5 - 8 - 10$$

$$\text{Path 3} = 1 - 2 - 3 - 4 - 5 - 6 - 9 - 10$$

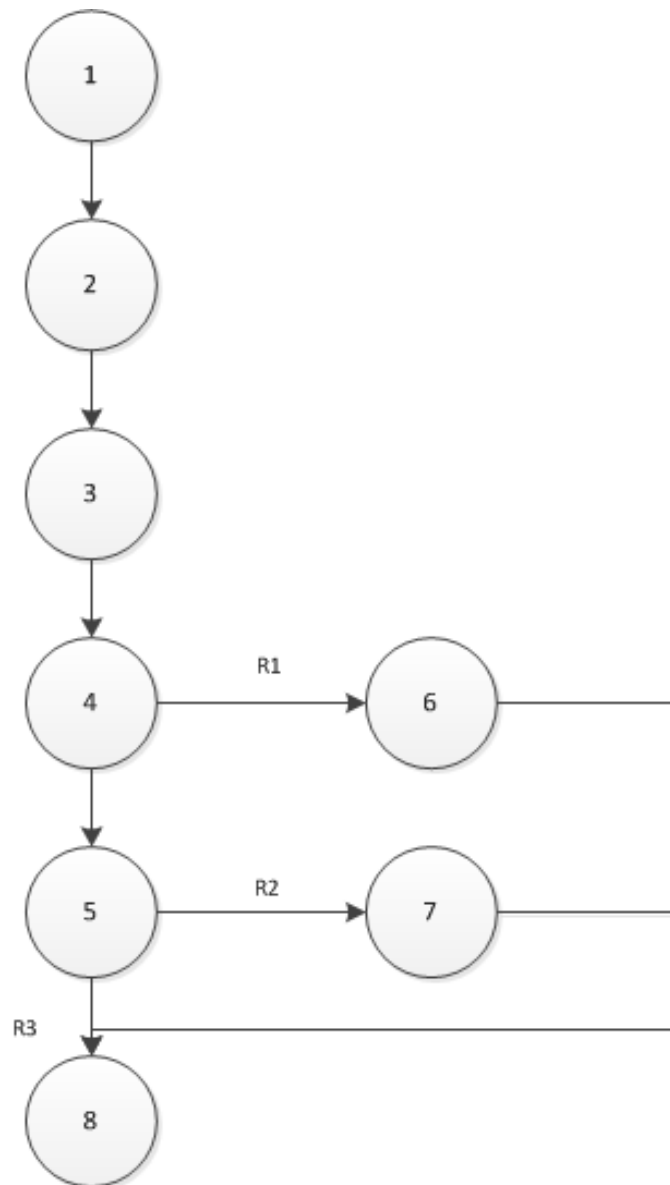
$$\text{Path 4} = 1 - 2 - 3 - 4 - 5 - 6 - 10$$

a. *White box* berdasarkan *form* tambah



Gambar 4.26 *Form* tambah data

1) *Flowgraph form* tambah data



Gambar 4.27 *Flowgraph Form* Tambah

2) Proses perhitungan *form* tambah

Dari gambar *flowgraph* di atas dapat dilakukan proses perhitungan sebagai berikut :

a) Menghitung *Cyclomatic Complexity* $V(G)$ dari *Edge* dan *Node*:

Dengan Rumus : $V(G) = E - N + 2$

$$N(\text{node}) = 8$$

$$E(\text{edge}) = 9$$

$$P(\text{predikat node}) = 2$$

$$\text{Penyelesaian : } V(G) = E - N + 2$$

$$= 9 - 8 + 2$$

$$= 3$$

$$\text{Predikat Node (N)} = P + 1$$

$$= 2 + 1$$

$$= 3$$

b) Berdasarkan perhitungan *Cyclomatic Complexity* dari *Flowgraph* di atas memiliki *Region* = 3

c) *Independent path* pada *flowgraph* di atas adalah :

$$\text{Path 1} = 1 - 2 - 3 - 4 - 6 - 8$$

$$\text{Path 2} = 1 - 2 - 3 - 4 - 5 - 7 - 8$$

$$\text{Path 3} = 1 - 2 - 3 - 4 - 5 - 8$$

BAB V

SIMPULAN DAN SARAN

A. Kesimpulan

Sistem aplikasi *Tracer Study* ini menjadi sebuah sistem yang memudahkan staf kemahasiswaan terutama di bidang *Tracer Study*, untuk mendata alumni, membuat kuisioner secara *online* tidak melakukan secara manual seperti sebelumnya.

1. Aplikasi *Tracer Study* Alumni dibuat dengan basis *website* yang memiliki desain *user interface* yang mudah untuk digunakan.
2. Universitas Muhammadiyah Parepare dapat melihat data alumni dan mengumpulkan data alumni terkait dengan cara melakukan penyebaran kuisioner secara online, sehingga memudahkan staf dalam menemukan data para alumni, Selain itu diberikan beberapa fitur untuk para alumni berupa forum diskusi untuk tetap saling terhubung dengan alumni lainnya serta fitur Info lowongan pekerjaan yang dapat membantu alumni dalam menemukan informasi lowongan pekerjaan sehingga membantu para alumni menemukan pekerjaan yang sesuai dengan kompetensinya. Dari 15 sampel alumni Universitas Muhammadiyah Parepare mendapatkan pekerjaan formal dan sesuai dengan bidang studi sebanyak 35,71%, Lembaga informal dan sesuai bidang studi sebanyak 50,00 %, Lembaga formal dan tidak sesuai bidang studi sebanyak 14,29 % dan Lembaga informal dan tidak sesuai bidang studi sebanyak 0 %.

B. Saran

Saran yang diberikan peneliti pada penelitian ini adalah :

Aplikasi *Tracer Study* berbasis *Website* ini diharapkan dapat dikembangkan dengan menambahkan menu-menu sesuai kebutuhan untuk meningkatkan minat para alumni dalam mengisi *tracer study*.

DAFTAR PUSTAKA

- Abdul Kadir. 2005 *"Dasar Pemrograman Java 2, Edisi II"*, Penerbit buku Andi, Yogyakarta.
- Abdul Kadir & Adhi Susanto. 2013 *"Teori dan Aplikasi Pengolahan Citra" Edisi I*, Penerbit buku Andi, Yogyakarta.
- Adi Nugroho. 2006 *"Rekayasa perangkat lunak menggunakan UML dan java pustaka belajar"*, Yogyakarta.
- Ali Ibrahim. 2008 *"Cara preaktis membuat website dinamis menggunakan XAMPP"*, Penerbit buku Prestasi Pustaka, Bandung.
- Andrea Adelheid, Nst Khairil. 2012. *"Buku Pintar Menguasai Php Mysql"*, Penerbit Buku Mediakita, Jakarta.
- Andy Krisianto. 2012, *"Panduan panel web hosting"*, Penerbit Buku Elex Media Komputido, Jakarta.
- Eko Kurniawan K. 2011 *"Belajar Java Dasar"* .Penerbit buku Strip Bandung, Bandung,.
Tesseract Engine (Online) <https://www.petanikode.com/tutorial/java/>
- Hermawan, Hariadi. 2015 *"Pemanfaatan Augmented Reality sebagai Media Informasi Kampus menggunakan Brosur"*. Seminar Nasional Teknologi Informasi dan Komunikasi 2015 (SENTIKA 2015) ISSN:2089-9815.
- Ir. Yuniar Supardi. 2011 *"Semua bisa menjadi programer android"*, Penerbit buku Elex Media Komputindo, Jakarta.
- Leonardi, Hermawan. 2019 *"Penerapan Augmented Reality Pada Brosur Interior Berbasis Multi Markerless Pada Android"*, Seminar Nasional Teknologi Informasi dan Komunikasi 2015 (SENTIKA 2015) ISSN:2089-9815.
- Nadira, Sujaini, Pratiwi. *"Implementasi Augmented Reality Pada Brosur Teknik Informatika Universitas Tanjungpura Menggunakan Metode Marker "*, Jurnal Sistem dan Teknologi Informasi (JUSTIN).

Nazruddin Safaat H. 2012 *“Pemrograman Aplikasi Mobile Smartphone dan Tablet PC Berbasis Android”*, Penerbit buku Informatika Bandung, Bandung.

Ronger S.Pressman. 2007 *“Rekayasa perangkat lunak”*, Penerbit buku Andi, Makassar.

Satria, Takwir. 2013. *“Aplikasi Data Mahasiswa Berbasis Visual Basic 6.0”*. Penerbit buku Elex Media Komputindo, Jakarta.

Simarmata, Janner. 2009 *“Rekayasa Perangkat Lunak”*, Penerbit Buku Andi, Medan.

Wahana. 2013 *“Android Programming With Eclipse”*, Penerbit buku Andi, Semarang.

Wahana Komputer. 2013 *“Step by Stp Menjadi Programmer Android”*, Penerbit buku Andi, Semarang.