

**PENGEMBANGAN SISTEM INFORMASI *TRACER STUDY*
PROGRAM STUDI ILMU KOMPUTER FMIPA
UNIVERSITAS NEGERI JAKARTA**

Skripsi

**Disusun untuk memenuhi salah satu syarat
memperoleh gelar Sarjana Komputer**



**Oleh:
Saulia Karina
3145153851**

**PROGRAM STUDI ILMU KOMPUTER
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS NEGERI JAKARTA**

2020

LEMBAR PENGESAHAN

Dengan ini saya mahasiswa Fakultas Matematika dan Ilmu Pengetahuan Alam,
Universitas Negeri Jakarta

Nama : Saulia Karina

No. Registrasi : 3145153851

Jurusan : Ilmu Komputer

Judul : Pengembangan Sistem Informasi *Tracer Study*

Program Studi Ilmu Komputer FMIPA Universitas Negeri Jakarta.

Menyatakan bahwa skripsi ini telah siap diajukan untuk sidang skripsi.

Menyetujui,

Dosen Pembimbing I

Dosen Pembimbing II

Ratna Widyati, S.Si, M.Kom.

NIP. 19750925 200212 2 002

Ir. Fariani Hermin I., M.T.

NIP. 19600211 198703 2 001

Mengetahui,

Ketua Program Studi Ilmu Komputer

Ir. Fariani Hermin I., M.T.

NIP. 19600211 198703 2 001

LEMBAR PERNYATAAN

Saya menyatakan dengan sesungguhnya bahwa skripsi dengan judul "**Pengembangan Sistem Informasi *Tracer Study* Program Studi Ilmu Komputer FMIPA Universitas Negeri Jakarta**" yang disusun sebagai syarat untuk memperoleh gelar Sarjana komputer dari Program Studi Ilmu Komputer Universitas Negeri Jakarta adalah karya ilmiah saya dengan arahan dari dosen pembimbing.

Sumber informasi yang diperoleh dari penulis lain yang telah dipublikasikan yang disebutkan dalam teks skripsi ini, telah dicantumkan dalam Daftar Pustaka sesuai dengan norma, kaidah dan etika penulisan ilmiah.

Jika dikemudian hari ditemukan sebagian besar skripsi ini bukan hasil karya saya sendiri dalam bagian-bagian tertentu, saya bersedia menerima sanksi pencabutan gelar akademik yang saya sanding dan sanksi-sanksi lainnya sesuai dengan peraturan perundang-undangan yang berlaku.

Jakarta, 25 Januari 2020

Saulia Karina

HALAMAN PERSEMBAHAN

*Untuk Mamah, Bapak,
dan Adik-adikku tercinta.*

KATA PENGANTAR

Dengan memanjatkan puji dan syukur ke hadirat Allah SWT karena dengan rahmat dan karunia-Nya, Tugas Akhir ini dapat terselesaikan tanpa halangan berarti. Keberhasilan dalam menyusun laporan Tugas Akhir ini tidak lepas dari bantuan berbagai pihak yang memberikan dukungan moril maupun materil dan masukan guna sempurnanya Tugas Akhir ini. Oleh karena itu dalam kesempatan ini penulis mengucapkan terima kasih kepada:

1. Ibu Ir. Fariani Hermin Indiyah, M.T., selaku Koordinator Program Studi Ilmu Komputer Fakultas MIPA Universitas Negeri Jakarta sekaligus dosen pembimbing kedua yang telah memberikan banyak bantuan, bimbingan, serta arahan dalam Tugas Akhir,
2. Ibu Ratna Widyati, S.Si., M.Kom., selaku dosen pembimbing pertama yang telah memberikan banyak bantuan, bimbingan, serta arahan dalam Tugas Akhir ini,
3. Bapak Med Irzal, M.Kom., selaku dosen pembimbing akademis penulis,
4. Seluruh Dosen di Jurusan Ilmu Komputer FMIPA UNJ, yang tidak bisa disebutkan satu-satu, atas ilmu dan bimbingannya selama penulis berkuliah,
5. Mamah dan Bapak yang selama ini telah sabar mendukung, mengarahkan, dan mendoakan penulis.
6. Bapak Agus Agung Permana, yang telah mendukung dan memberi masukan terkait sistem yang dikembangkan pada penelitian ini,

7. Teman-teman Ilmu Komputer angkatan 2015 yang senantiasa menemani, memberikan semangat, mendukung dan memotivasi dari semenjak awal dunia perkuliahan sampai penulisan Tugas Akhir ini,
8. Hidayatul Rizkiyanti dan Maulana Rahman Nur, teman seperjuangan yang senantiasa mendukung, membantu dan memberi saran masukan kepada penulis
9. Mega, Della, Pipit, Adel, Farah, dan Yemima yang senantiasa memberikan semangat dan motivasi kepada penulis dalam menyelesaikan Tugas Akhir ini
10. Penulis juga mengucapkan terima kasih sebesar-besarnya kepada alumni Ilmu Komputer UNJ karena telah membantu dan memberikan saran masukan dalam penelitian ini

Penulis menyadari bahwa laporan Tugas Akhir ini masih jauh dari kesempurnaan. Masih banyaknya kekurangan dalam penyusunan laporan ini baik materi maupun tata cara penulisan. Untuk itu, kritik dan saran yang bersifat membangun sangat diharapkan bagi penulis demi tercapainya laporan yang lebih baik lagi. Akhir kata penulis mohon maaf apabila ada kekeliruan di dalam penulisan Tugas Akhir ini.

Jakarta, 25 Januari 2020

Penulis

ABSTRAK

SAULIA KARINA. Pengembangan Sistem Informasi *Tracer Study* Program Studi Ilmu Komputer FMIPA Universitas Negeri Jakarta . Skripsi. Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Negeri Jakarta. 2020. Di bawah bimbingan Ratna Widyati, S.Si, M.Kom dan Ir. Fariani I., M.T.

Tracer Study merupakan studi penelusuran alumni dengan tujuan menggali informasi tentang perjalanan alumni setelah lulus dari perguruan tinggi. Melalui kegiatan tersebut perguruan tinggi dapat mengetahui apakah kemampuan yang dimiliki lulusannya sudah sesuai yang diharapkan program studi. Informasi yang didapat berguna untuk evaluasi terhadap proses pendidikan, sumber informasi bagi pihak perguruan tinggi dalam membuat kebijakan khususnya kurikulum, dan kelengkapan persyaratan akreditasi. Saat ini telah tersedia beberapa sistem *tracer study*, yang pertama ialah yang dikembangkan oleh Dikti. Sistem *tracer study* Dikti bertujuan untuk merintis kompilasi data *tracer study* nasional dimana informasi yang dihasilkan dapat membantu pemerintah dalam memetakan kebutuhan dunia kerja. Kedua, Universitas Negeri Jakarta telah memiliki sistem *tracer study* yang dapat membantu pihak perguruan tinggi maupun program studi untuk mengarsipkan data alumni. Pada penelitian ini dilakukan analisis dan evaluasi dari perbandingan kedua sistem tersebut yang kemudian hasilnya akan dituangkan pada pengembangan sistem informasi *tracer study* pada program studi Ilmu Komputer. Pada pengembangan sistem ini tidak hanya bersifat repositori, namun disediakan *open access* bagi pengunjung sistem untuk melihat data alumni dan hasil *tracer*. Sistem *tracer study* ini memiliki lima *user*, yaitu admin, koorprodi, alumni, dan pengunjung yang memiliki sub-aktor pengguna alumni. Admin dapat mengelola data alumni, kuesioner dan hasil *tracer*, koorprodi dapat melihat data alumni dan hasil atau laporan *tracer study*, alumni dapat mengelola data diri dan mengisi pertanyaan kuesioner, dan pengguna alumni dapat mengisi kuesioner. Sistem ini dikembangkan dengan menggunakan metode pengembangan perangkat lunak SDLC (*System Development Life Cycle*) dengan model Spiral. Dalam pengkodean sistem akan digunakan konsep MVC (*Model View Controller*). Berdasarkan pengujian UAT (*User Acceptance Test*) baik fungsional maupun kebergunaan *usability* dapat dikatakan bahwa Sistem Informasi *Tracer Study* Prodi Ilmu Komputer telah berjalan dengan baik dan sesuai yang diharapkan dengan tingkat kebergunaan senilai 85.66% pada keseluruhan sistem.

Kata kunci : *Tracer Study*, Sistem Informasi, *System Development Life Cycle*, *Spiral Model*, *Model View Controller*.

ABSTRACT

SAULIA KARINA. Development of Tracer Study Information System at Computer Science Department FMIPA State University of Jakarta. Thesis. Faculty of Mathematics and Science, State University of Jakarta. 2020. Under supervised by Ratna Widyati, S.Si, M.Kom and Ir. Fariani Hermin I., M.T.

Tracer study is tracking studies trace graduates / alumni aims to get information about alumni's life path or movement after graduating from college. Through the study, the higher education institutions can determine graduates' capabilities, the relevance of their training to their tasks. The information from the study can use for evaluating the educational process, a source of information for the college to make policies, especially for curriculum, and completing of requirements accreditation. Currently, there are several tracer study information systems available, the first is developed by Dikti. The Dikti Tracer Study system aims to obtain compilation of national tracer study data where information can be generated assist the government in mapping the needs of the field of work. The second is Tracer Study Information System of State University of Jakarta. The system can help institution and department to archive the graduates data. On this research, conducted analysis and evaluation of the comparison of the two systems, then the result is used for developing tracer study information system at Computer Science Department. On this system development, there is not only repositories, but provide open access for visitors of the system to see graduates data and some tracer results. This tracer study system has five users, i.e. admin, department coordinator, alumni, and visitors who have sub-actors stakeholders. Admin can manage alumni data, questionnaires and tracer results, department coordinator can see alumni data and results or tracer study reports, alumni can manage self data and answer the questionnaire questions, and stakeholder can answer the questionnaires too. This system was developed using the software development method, SDLC (System Development Life Cycle) with Spiral model. In the coding phase used the concept of MVC (Model View Controller). Based on UAT testing (User Acceptance Test) both functionality and usability can be said that Tracer Study Information Systems of Computer Science Department has been running well and as expected and the usability percentage is 85.66% for the whole system.

Keywords : Tracer Study, Information System, System Development Life Cycle, Spiral Model, Model View Controller.

DAFTAR ISI

HALAMAN PERSEMBAHAN	iv
KATA PENGANTAR	v
ABSTRAK	vii
<i>ABSTRACT</i>	viii
DAFTAR ISI	xiv
DAFTAR GAMBAR	xiv
DAFTAR TABEL	xv
I PENDAHULUAN	1
A. Latar Belakang	1
B. Identifikasi Masalah	5
C. Rumusan Masalah	5
D. Batasan Masalah	5
E. Tujuan Penelitian	6
F. Manfaat Penelitian	6
II KAJIAN TEORI	8
A. <i>Tracer Study</i> (Studi Penelusuran)	8
B. Sistem Informasi	9
C. SDLC (<i>System Development Life Cycle</i>)	11
1. <i>Spiral Model</i>	14
D. <i>Unified Modeling Language</i> (UML)	15

1.	<i>Use Case Diagram</i>	16
2.	<i>Activity Diagram</i>	18
3.	<i>Class Diagram</i>	20
E.	<i>Entity Relationship Database(ERD)</i>	23
F.	<i>Model View Controller (MVC)</i>	23
G.	<i>Basis Data</i>	25
III IMPLEMENTASI PROGRAM		28
A.	<i>Analisis Kebutuhan</i>	28
B.	<i>Desain Sistem</i>	29
1.	<i>Use Case Diagram</i>	29
2.	<i>Entity Relationship Diagram</i>	32
3.	<i>Class Diagram</i>	33
4.	<i>Activity Diagram</i>	34
5.	<i>Rancangan Antar Muka Program</i>	39
C.	<i>Pengembangan</i>	47
1.	<i>Membangun Basis Data</i>	47
2.	<i>Implementasi Desain Tampilan</i>	48
3.	<i>Implementasi Sistem (Back End)</i>	59
IV UJI COBA DAN HASIL UJI COBA		62
A.	<i>Uji Coba</i>	62
B.	<i>Hasil Percobaan</i>	66
1.	<i>Pengujian oleh Admin</i>	66
2.	<i>Pengujian oleh Koorprodi</i>	68
3.	<i>Pengujian oleh Alumni</i>	71
4.	<i>Pengujian oleh Pengguna Alumni</i>	73

5.	Hasil Pengujian Keseluruhan Sistem	75
V	KESIMPULAN DAN SARAN	77
A.	Kesimpulan	77
B.	Saran	78
	DAFTAR PUSTAKA	79
	LAMPIRAN	83
A	Perbandingan Fitur Sistem <i>Tracer Study</i>	83
B	Perbandingan Kuesioner Sistem <i>Tracer Study</i>	85
C	Analisis Kebutuhan Sistem dengan Pihak Prodi Ilmu Komputer	88
D	Sampel Kode <i>Controller</i> Alumni pada Admin	90
E	Sampel Kode <i>Model</i> Alumni pada Admin	100
F	Kuesioner <i>User Acceptance Test</i> pada Admin	106
G	Kuesioner <i>User Acceptance Test</i> pada Koorprodi	110
H	Kuesioner <i>User Acceptance Test</i> pada Alumni	114
I	Kuesioner <i>User Acceptance Test</i> pada Pengguna Alumni	118
J	Dokumentasi Pengembangan Sistem dengan Model <i>Spiral</i>	122

DAFTAR GAMBAR

Gambar 2.1	<i>Spiral Model</i>	15
Gambar 2.2	<i>Contoh Use Case Diagram</i>	18
Gambar 2.3	<i>Contoh sederhana Class Diagram</i>	20
Gambar 2.4	Alur Kerja MVC	24
Gambar 3.1	<i>Use Case Diagram</i> Sistem Informasi <i>Tracer Study</i> Ilmu Kom- puter	31
Gambar 3.2	<i>Entity Relationship Diagram</i> Sistem Informasi <i>Tracer Study</i> Ilmu Komputer	32
Gambar 3.3	Desain <i>Class Diagram</i> Sistem Informasi <i>Tracer Study</i> Ilmu Komputer	33
Gambar 3.4	<i>Activity Diagram</i> Membuat Kuesioner <i>Tracer Study</i>	34
Gambar 3.5	<i>Activity Diagram</i> Melihat Laporan <i>Tracer Study</i>	35
Gambar 3.6	<i>Activity Diagram</i> Mengisi Kuesioner Alumni	36
Gambar 3.7	<i>Activity Diagram</i> Mengisi Riwayat Pekerjaan Alumni	37
Gambar 3.8	<i>Activity Diagram</i> Mengisi Kuesioner Pengguna Alumni	38
Gambar 3.9	<i>Activity Diagram</i> Melihat Daftar Pengguna Alumni	39
Gambar 3.10	Desain Halaman Daftar Pengguna Alumni untuk Pengunjung	40
Gambar 3.11	Desain Halaman Statistik Alumni untuk Pengunjung	41
Gambar 3.12	Desain Halaman Admin	42
Gambar 3.13	Desain Halaman Kelola Data Alumni	42
Gambar 3.14	Desain Halaman Kelola Data Pengguna Alumni	43
Gambar 3.15	Desain Halaman Kelola Kuesioner	43
Gambar 3.16	Desain Halaman Laporan <i>Tracer Study</i>	44
Gambar 3.17	Desain Halaman Laporan Pengguna Alumni	44

Gambar 3.18	Desain Halaman Koorprodi	45
Gambar 3.19	Desain Halaman Alumni	45
Gambar 3.20	Desain Halaman Kuesioner Alumni	46
Gambar 3.21	Desain Halaman Daftar Pengguna Alumni untuk Alumni . . .	47
Gambar 3.22	Basis Data Sistem Informasi <i>Tracer Study</i> Prodi Ilmu Kom- puter	48
Gambar 3.23	Tampilan Halaman Beranda Admin	49
Gambar 3.24	Tampilan Halaman Kelola Alumni pada Admin	49
Gambar 3.25	Tampilan Halaman Kelola Pengguna Alumni pada Admin . .	50
Gambar 3.26	Tampilan Halaman Kelola Kuesioner Alumni pada Admin .	50
Gambar 3.27	Tampilan Halaman Buat Kuesioner Alumni pada Admin . .	51
Gambar 3.28	Tampilan Halaman Buat Kuesioner Pengguna pada Admin .	51
Gambar 3.29	Tampilan Halaman Kelola Beranda Alumni pada Admin . .	52
Gambar 3.30	Tampilan Halaman Kriteria Laporan Alumni pada Admin . .	52
Gambar 3.31	Tampilan Halaman Kriteria Laporan Pengguna Alumni pada Admin	53
Gambar 3.32	Tampilan Halaman Hasil <i>Tracer Study</i> pada Admin dan Ko- orprodi	53
Gambar 3.33	Tampilan Halaman Beranda pada Alumni	54
Gambar 3.34	Tampilan Halaman Data Diri pada Alumni	54
Gambar 3.35	Tampilan Halaman Form Riwayat Pekerjaan pada Alumni . .	55
Gambar 3.36	Tampilan Halaman Kuesioner pada Alumni	55
Gambar 3.37	Tampilan Halaman Daftar Pengguna pada Alumni	56
Gambar 3.38	Tampilan Halaman Daftar Pengguna pada Alumni	56
Gambar 3.39	Tampilan Halaman Daftar Alumni pada Koorprodi	57
Gambar 3.40	Tampilan Halaman Kuesioner untuk Pengguna Alumni	58

Gambar 3.41 Tampilan Halaman Daftar Pengguna Alumni untuk Pengunjung	58
Gambar 3.42 Tampilan Halaman Statistik Alumni untuk Pengunjung	59
Gambar 3.43 Struktur Pemrograman <i>Controller</i>	60
Gambar 3.44 Struktur Pemrograman <i>Model</i>	60
Gambar 3.45 Struktur Pemrograman <i>View</i>	61

DAFTAR TABEL

Tabel 2.1	Simbol-simbol <i>Use Case Diagram</i> [13]	17
Tabel 2.2	Simbol-simbol <i>Activity Diagram</i> [13]	19
Tabel 2.3	<i>Multiplicity Class Diagram</i>	21
Tabel 2.4	Simbol-simbol <i>Class Diagram</i> [3]	22
Tabel 4.1	Hasil Uji Fungsional pada <i>user Admin</i>	66
Tabel 4.2	Daftar Pertanyaan Uji <i>Usability</i> pada Admin	67
Tabel 4.3	Hasil uji <i>usability</i> pada Admin	68
Tabel 4.4	Hasil Uji Fungsional pada Koorprodi	69
Tabel 4.5	Daftar Pertanyaan Uji <i>Usability</i> pada Koorprodi	70
Tabel 4.6	Hasil Uji <i>Usability</i> pada Koorprodi	70
Tabel 4.7	Hasil Uji Fungsional pada Alumni	71
Tabel 4.8	Daftar Pertanyaan Uji <i>Usability</i> pada Alumni	72
Tabel 4.9	Hasil Uji Fungsional pada Pengguna Alumni	74
Tabel 4.10	Daftar Pertanyaan Uji <i>Usability</i> pada Pengguna Alumni	74
Tabel 4.11	Hasil Uji <i>Usability</i> pada Pengguna Alumni	75

BAB I

PENDAHULUAN

A. Latar Belakang

Universitas Negeri Jakarta (UNJ) sebagai salah satu institusi perguruan tinggi mempunyai peran dan fungsi mempersiapkan sumber daya manusia yang handal dan kompetitif sesuai bidangnya yang menjadi aset masyarakat, pemerintah dan bangsa, sehingga dapat memberikan kontribusi dalam pembangunan bangsa dan negara Indonesia [8]. Dengan demikian, UNJ diharapkan dapat menghasilkan lulusan berkompeten di bidangnya yang siap terjun ke jenjang pekerjaan dan masyarakat. Untuk menciptakan lulusan yang berkualitas tentunya suatu perguruan tinggi harus memiliki sistem pendidikan yang baik. Sistem pendidikan suatu program studi dikatakan baik jika lulusannya dibutuhkan oleh dunia kerja pengguna lulusan, sehingga untuk mencapai dalam taraf tersebut maka program studi harus tahu keinginan para pengguna lulusan [20]. Untuk mengetahui tingkat relevansi (kesesuaian) antara kemampuan lulusan yang diperoleh melalui proses pendidikan di perguruan tinggi dengan kebutuhan dunia kerja dapat dilakukan upaya penelusuran alumni (*tracer study*).

Tracer study adalah studi penelusuran jejak alumni dilakukan setelah kelulusan dan bertujuan untuk mengetahui *outcome* pendidikan dalam bentuk transisi dari dunia pendidikan tinggi ke dunia kerja [23]. *Tracer study* sangat berguna untuk evaluasi terhadap hasil pendidikan tinggi, relevansi dan sumber informasi bagi pemangku kepentingan dalam penentuan kebijakan salah satunya pengembangan kurikulum, serta kelengkapan persyaratan bagi akreditasi Dikti [24]. Oleh karena itu, *tracer study* menjadi sangat penting untuk menjadi bahan pertimbangan dalam penentuan kebijakan akademik, khususnya dalam penyusunan kurikulum dan penilaian akreditasi.

Sejak tahun 2011 Dikti telah mengembangkan sistem *online* untuk merintis kompilasi data *tracer study* nasional mengenai transisi dan posisi pekerjaan lulusan perguruan tinggi di Indonesia. Hasil *tracer study* yang kemudian dilaporkan ke Dikti akan membantu program Pemerintah dalam rangka memetakan kebutuhan dunia kerja dengan pembangunan pendidikan di Indonesia [25]. Namun baru sejumlah 675 data alumni UNJ yang tersedia pada sistem tersebut.

Berdasarkan website resmi Universitas Negeri Jakarta, UNJ memiliki delapan fakultas yang mengelola berbagai program studi baik pendidikan maupun non-pendidikan. Salah satu program studi yang ada di UNJ ialah Ilmu Komputer yang merupakan program studi non-pendidikan. Berdasarkan informasi yang didapat dari Koorprodi Ilmu Komputer, prodi Ilmu Komputer telah melakukan kegiatan *tracer study* sejak prodi ini memiliki lulusan pertama pada semester 2016/2017. Pelaksanaan *tracer study* tersebut dilakukan melalui pengiriman kuesioner melalui pos, *e-mail* dan media sosial seperti *whatsapp*. Bentuk pelaksanaan tersebut dirasa kurang efektif karena dapat membutuhkan waktu lama terhadap respon alumni. Saat itu, di UNJ memang belum ada sistem informasi *tracer study* baik ditingkat Universitas, Fakultas, dan Prodi, semua kegiatan *tracer study* dilakukan secara manual dengan mengirimkan *e-mail* atau *google docs*. Menurut Bapak Prasetyo selaku staf wakil rektorat bidang kemahasiswaan cara itu sangat manual dan tidak tervalidasi, artinya bisa saja ada kemungkinan orang lain yang bukan alumni bisa mengisi data-data *tracer study* tersebut [1].

Berdasarkan uraian di atas diadakan penelitian oleh Rifqi Syahirul Alim pada tahun 2019 dalam skripsinya berjudul “Perancangan dan Implementasi Sistem Informasi Penelusuran Alumni (*Tracer Study*) Universitas Negeri Jakarta”. Penelitian tersebut telah menghasilkan sistem informasi penelusuran alumni yang dapat memudahkan pihak universitas maupun prodi dalam mengelola dan mengarsipkan data

alumni.

Hasil analisis dari perbandingan sistem *tracer study* yang telah ada dituangkan pada tabel perbandingan fitur [Lampiran A] dan diuraikan sebagai berikut. Sistem *Tracer Study* yang dikembangkan oleh Dikti mempermudah setiap perguruan tinggi untuk melihat hasil atau laporan tracer study karena disediakan halaman yang menampilkan hasil *tracer study* dalam bentuk grafik dan tabel yang informatif. Sistem ini menyajikan data *tracer study* yang dapat dilihat oleh pengunjung tanpa harus masuk ke sistem. Kuesioner pada sistem ini telah disediakan dan distandarisasi oleh Dikti. Namun, data yang ada pada sistem tidak dapat diekspor, sehingga pihak perguruan tinggi akan kesulitan ketika akan membuat laporan. Selain itu, sistem tidak menyediakan kuesioner bagi pengguna lulusan. Informasi yang didapatkan dari Koorprodi Ilmu Komputer bahwa penelitian terhadap pengguna diperlukan untuk mengetahui bagaimana penilaian pengguna terhadap kompetensi lulusan dan kurikulum yang berjalan apakah sudah mencukupi dan relevan dengan kebutuhan dunia kerja saat ini. Selain itu, akreditasi perguruan tinggi juga membutuhkan informasi mengenai evaluasi kinerja lulusan dan umpan balik dari pengguna lulusan [16]. Penelitian terkait telah dilakukan oleh Achmad Ghazaly et al dalam jurnalnya berjudul “Rancang Bangun Aplikasi *Tracer Study* Berbasis Web pada Stikes Yayasan Rs. Dr. Soetomo Surabaya”. Pada penelitian tersebut selain kuesioner untuk alumni pada sistem juga disediakan kuesioner dan umpan balik yang diperuntukkan bagi pengguna alumni [10].

Sistem informasi *tracer study* Universitas Negeri Jakarta menyediakan layanan tracer study di tingkat perguruan tinggi dan juga program studi. Admin baik universitas maupun prodi dapat mengelola form kuesioner sesuai dengan kebutuhan. Sistem tersebut dapat menampilkan hasil *tracer study* yaitu berupa grafik dan tabel yang juga informatif. Admin universitas dapat mengunggah data alumni dari seluruh prodi di UNJ. Alumni dapat melakukan pengisian form kuesioner secara *online*. Selain itu,

hasil *tracer study* dapat diekspor ke dalam format *excel* sehingga memudahkan pihak universitas maupun prodi dalam membuat laporan *tracer study*. Namun, pada sistem tersebut belum tersedia kuesioner yang diperuntukkan bagi pengguna lulusan. Kemudian, baik prodi maupun universitas tidak dapat mengelola halaman beranda. Dimana pada beranda alumni terdapat kata pengantar yang tidak dapat disunting oleh admin melalui suatu tatap muka. Admin prodi tidak dapat melihat daftar alumni prodinya masing-masing. Selain itu, belum tersedianya halaman bagi pengunjung untuk dapat melihat data *tracer study*.

Berdasarkan uraian di atas akan dikembangkan Sistem *Tracer Study* Program Studi Ilmu Komputer Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Negeri Jakarta berdasarkan analisis dan evaluasi dari perbandingan sistem *tracer study* yang telah tersedia, yaitu Sistem *Tracer Study* Dikti dan Sistem Informasi *Tracer Study* UNJ. Pada penelitian ini diusulkan beberapa fitur, yaitu pertama berdasarkan penjelasan di atas peran pengguna alumni diperlukan untuk penilaian terhadap kompetensi lulusan, masukan bagi program studi, dan memenuhi kebutuhan akreditasi, maka pada pengembangan sistem ini akan disediakan akses bagi pengguna alumni untuk mengisi kuesioner. Pengunjung sistem dapat melihat daftar pengguna alumni sehingga dapat memfasilitasi pengunjung dalam mencari informasi pekerjaan di bidang Ilmu Komputer. Kedua, informasi yang didapat dari koorprodi Ilmu Komputer bahwa koorprodi membutuhkan akses terhadap sistem untuk mengakses data *tracer* yang diperlukan bagi akreditasi maupun penjaminan mutu internal. Ketiga, pada sistem akan dibuat suatu tatap muka untuk admin agar dapat mengelola kata pengantar *tracer study* pada beranda alumni sesuai kebutuhan. Terakhir, disediakan halaman data *tracer* yang dapat diakses oleh pengunjung sehingga dapat memfasilitasi masyarakat untuk lebih mengenal lulusan prodi Ilmu Komputer UNJ. Pengembangan sistem *tracer study* ini tertuang pada penelitian yang berjudul **“Pengembangan Sistem In-**

formasi *Tracer Study* Program Studi Ilmu Komputer FMIPA Universitas Negeri Jakarta”.

B. Identifikasi Masalah

Dari uraian yang dikemukakan pada latar belakang, maka dapat diidentifikasi beberapa masalah sebagai berikut:

1. Pada sistem *tracer study* yang telah ada belum disediakan kuesioner yang diperuntukkan bagi pengguna alumni.
2. Pada Sistem *Tracer Study* Dikti hasil *tracer* tidak dapat diekspor.
3. Pada Sistem *Tracer Study* UNJ tidak disediakan tatap muka (*interface*) untuk mengelola kata pengantar *tracer study*, tidak adanya halaman pengunjung untuk melihat data *tracer*, dan admin prodi tidak dapat melihat daftar lulusannya.

C. Rumusan Masalah

Adapun rumusan masalah berdasarkan pada latar belakang yang telah dipaparkan adalah sebagai berikut :

1. Bagaimana konsep rancangan dari pengembangan Sistem *Tracer Study* Program Studi Ilmu Komputer FMIPA UNJ ?
2. Bagaimana implementasi rancangan pengembangan Sistem *Tracer Study* Program Studi Ilmu Komputer FMIPA UNJ ke dalam program berbasis *website* ?

D. Batasan Masalah

Adapun batasan masalah pada penelitian ini ialah :

1. Metode pengembangan sistem yang digunakan merupakan salah satu model pengembangan perangkat lunak dari metode *System Development Life Cycle*, yaitu *spiral model*.
2. Data hasil *tracer study* pada penelitian ini dideskripsikan dalam bentuk grafik dan tabel.

E. Tujuan Penelitian

Penyusunan tugas akhir ini bertujuan untuk mengembangkan sistem informasi *tracer study* pada prodi Ilmu Komputer FMIPA UNJ berdasarkan analisis dan evaluasi dari perbandingan sistem yang telah tersedia, yaitu Sistem Informasi *Tracer Study* UNJ dan Sistem *Tracer Study* Dikti.

F. Manfaat Penelitian

Adapun manfaat yang diharapkan dari penelitian ini adalah sebagai berikut :

1. Bagi Peneliti
 - Memberikan gambaran mengenai pekerjaan yang akan digeluti setelah lulus dari program studi Ilmu Komputer
 - Menambah pengetahuan dan keterampilan dalam mengembangkan suatu *website*.
2. Bagi Alumni, memberikan referensi terkait jenjang pekerjaan yang dapat digeluti oleh lulusan Ilmu Komputer.
3. Bagi Program Studi
 - Mengetahui tingkat relevansi antara sistem pendidikan pada program studi Ilmu Komputer dengan kebutuhan di dunia kerja.

- Mengetahui apakah kompetensi yang dimiliki lulusan ilmu komputer sesuai dengan yang diharapkan program studi.
- Memberikan informasi terkait data akreditasi yang dibutuhkan oleh program studi Ilmu Komputer.
- Memberikan masukan kepada program studi terkait kualitas alumni dan kurikulum yang ada berdasarkan penilaian dan umpan balik dari pengguna alumni.
- Mengetahui sejauh mana daya serap program studi Ilmu Komputer pada lapangan pekerjaan.

4. Bagi Masyarakat

- Memfasilitasi masyarakat untuk lebih mengenal prospek kerja dari prodi Ilmu Komputer UNJ dan kualitas lulusannya.
- Meningkatkan kepercayaan dan pengakuan kiprah lulusan prodi Ilmu Komputer UNJ di masyarakat
- Memberikan referensi mahasiswa dalam mencari tempat Praktik Kerja Lapangan
- Memberikan bekal bagi mahasiswa agar setelah lulus dapat lebih siap untuk memasuki dunia kerja dan dapat beradaptasi dengan baik.

BAB II

KAJIAN TEORI

A. *Tracer Study* (Studi Penelusuran)

Menurut Nuryake Fajaryati et al dalam Jurnal ELINVO Vol 1 No 1 (2015), *tracer study* merupakan studi yang tujuan utamanya untuk memperoleh informasi mengenai lulusan yang sudah bekerja dan belum bekerja. Selain itu *tracer study* bertujuan untuk mengetahui hasil pendidikan dalam bentuk penguasaan dan pemerolehan kompetensi lulusan yang diaplikasikan di dunia kerja serta transisi dari dunia pendidikan tinggi ke dunia usaha dan industri. Melalui *tracer study* ini penyelenggara pendidikan dapat mengetahui bagaimana penyelenggaraan dan mutu layanan program melalui penilaian para alumni sehingga mampu untuk memperbaiki dan meningkatkan kualitas layanannya [7].

Tracer study adalah studi penelusuran jejak lulusan dilakukan setelah kelulusan dan bertujuan untuk mengetahui *outome* pendidikan dalam bentuk transisi dari dunia pendidikan tinggi ke dunia kerja. Memberikan informasi mengenai *output* pendidikan yaitu penilaian diri terhadap penguasaan dan pemerolehan kompetensi, proses pendidikan berupa evaluasi proses pembelajaran dan kontribusi pendidikan tinggi terhadap pemerolehan kompetensi serta input pendidikan dalam bentuk penggalan lebih lanjut terhadap informasi sosiobiografis lulusan [23].

Berdasarkan uraian di atas *tracer study* adalah suatu studi yang dilakukan untuk menggali informasi terkait kondisi alumni, yaitu dalam masa transisi dari pendidikan tinggi ke dunia kerja, bertujuan untuk mengetahui hasil dari proses pendidikan suatu perguruan tinggi berupa kompetensi lulusan, hasil yang didapat dijadikan bahan evaluasi bagi perguruan tinggi untuk meningkatkan mutu layanan atau program pen-

didikannya. Menurut Soemantri (2010) dalam *Jurnal Electronics, Informatics, and Vocational Education (ELINVO)*, Volume 1, Nomor 1, November 2015, terdapat tiga manfaat yang dapat diperoleh dari pelaksanaan *tracer study*, yaitu [7] :

1. Mengetahui kepuasan *stakeholders*, dalam hal ini lulusan, terkait dengan *learning experiences* yang mereka alami, untuk dijadikan alat evaluasi kerja institusi.
2. Mendapatkan masukan yang relevan sebagai dasar pijakan pengembangan institusi, terkait dengan kemampuan bersaing, kualitas, dan *working experiences* lulusan yang bisa digunakan untuk menangkap kesempatan dan menanggulangi ancaman ke depan.
3. Meningkatkan hubungan lulusan dan almamater, karena apabila dilihat dari pengalaman institusi-institusi pendidikan terkenal, ikatan lulusan dan almamater yang kuat akan membawa banyak manfaat kepada almamater seiring dengan diakuinya kiprah lulusan di masyarakat.

B. Sistem Informasi

Sistem Informasi terdiri dari kata sistem dan informasi. Sistem adalah kumpulan orang yang saling bekerja sama dengan ketentuan-ketentuan aturan yang sistematis dan terstruktur untuk membentuk satu kesatuan yang melaksanakan suatu fungsi untuk mencapai tujuan. Sedangkan informasi adalah data yang diolah menjadi lebih berguna dan berarti bagi penerimanya, serta untuk mengurangi ketidakpastian dalam proses pengambilan keputusan mengenai suatu keadaan. Sistem informasi merupakan suatu kombinasi teratur dari orang-orang, *hardware*, *software*, jaringan komunikasi dan sumber daya data yang mengumpulkan, mengubah, dan menyebarkan informasi dalam sebuah organisasi [2]

Menurut Sebastian K Boell dan Dubravka Cecez-Kecmanovic (2015), Sistem Informasi melibatkan berbagai teknologi informasi seperti komputer, perangkat lunak, basis data, sistem komunikasi, internet, perangkat seluler dan masih banyak lagi, untuk melakukan tugas tertentu, berinteraksi dengan dan memberi tau berbagai pengguna dari organisasi yang berbeda [5]. Sistem Informasi dari suatu organisasi terdiri komponen-komponen berikut [19] :

1. Perangkat keras, yaitu perangkat keras komponen untuk melengkapi kegiatan memasukkan data, memproses data, dan keluaran data.
2. Perangkat lunak, yaitu program dan instruksi yang diberikan ke komputer.
3. Basis data, yaitu kumpulan data dan informasi yang diorganisasikan sedemikian rupa, sehingga mudah diakses pengguna sistem informasi.
4. Telekomunikasi, yaitu komunikasi yang menghubungkan antara pengguna sistem dengan sistem komputer secara bersama-sama ke dalam suatu jaringan kerja yang efektif.
5. Manusia, yaitu personel dari sistem informasi, meliputi manajer, analis, programmer, dan operator, serta bertanggung jawab terhadap sistem.

Berdasarkan definisi di atas dapat dikatakan bahwa Sistem Informasi adalah sekumpulan dari berbagai komponen baik teknologi informasi maupun penggunaannya yang saling terkait dan bekerja sama dalam melaksanakan tugas seperti mengumpulkan, mengolah, menyajikan dan menyebarkan data dan informasi untuk mencapai suatu tujuan yaitu memberikan informasi yang dapat dimanfaatkan dalam membuat keputusan.

C. SDLC (*System Development Life Cycle*)

Untuk mengembangkan suatu sistem perlu melewati serangkaian tahapan dari mulai sistem tersebut direncanakan sampai dengan sistem diterapkan dan dipelihara. Tahapan-tahapan tersebut mengacu pada proses-proses standar, yakni analisis, desain, implementasi dan pemeliharaan. Proses-proses standar tersebut dituangkan dalam satu metode yang dinamakan *System Development Life Cycle* (SDLC).

SDLC merupakan konsep yang digunakan dalam rekayasa perangkat lunak yang menggambarkan sebuah prosedur mulai dari perencanaan, pembuatan, pengkodean, pengujian dan implementasi dari spesifikasi kebutuhan pengguna. SDLC adalah proses bertahap untuk membuat perangkat lunak berkualitas bagi pengguna dalam waktu yang ditentukan. SDLC melibatkan beberapa fase berbeda yang dijalankan satu per satu secara berurutan, dimana sangat penting bagi pengembang perangkat lunak, seperti perencanaan, analisis, desain, *coding*, dan implementasi. Dan juga termasuk evaluasi perangkat lunak, pengumpulan informasi, studi kelayakan, dan permintaan persetujuan [26].

Berdasarkan penjelasan di atas, *System Development Life Cycle* (SDLC) adalah serangkaian prosedur yang digunakan dalam mengembangkan perangkat lunak mencakup tahapan-tahapan berbeda yang dijalankan secara berurutan, yakni analisis atau perencanaan, desain, implementasi, pengujian dan pemeliharaan. Tahapan-tahapan yang terdapat dalam metode SDLC adalah sebagai berikut [26] :

1. Spesifikasi dan pengumpulan kebutuhan perangkat lunak

Tahap ini dimulai oleh pihak pengembang *software* dengan mengumpulkan semua kebutuhan atau *requirements* dari pengguna. Pengumpulan kebutuhan dapat dilakukan dengan mempelajari sistem yang sudah ada, mengadakan wawancara dengan pengguna, merujuk ke basis data atau dengan mengumpulkan

jawaban dari kuesioner. Fase ini harus dijalankan secara hati-hati karena perangkat lunak yang berkualitas bergantung pada semua informasi yang dikumpulkan dari pengguna.

2. Studi kelayakan kebutuhan

Pada tahap ini tim pengembang melakukan analisis apakah perangkat lunak dapat dirancang untuk memenuhi kebutuhan pengguna. Selain itu, juga menganalisis apakah perangkat lunak layak secara finansial, praktis, dan teknis untuk digunakan oleh suatu organisasi. Secara finansial dalam arti sesuai dengan anggaran. Sedangkan layak secara praktis dan teknis berarti mudah dioperasikan oleh pengguna dimasa depan.

3. Analisis Perangkat Lunak

Analisis perangkat lunak mencakup pemahaman tentang batasan produk perangkat lunak, masalah terkait sistem atau perubahan yang harus dilakukan dalam sistem yang sudah ada. Tim pengembang menganalisis ruang lingkup perangkat lunak dan merencanakan jadwal dan sumber daya yang sesuai.

4. Desain Perangkat Lunak

Fase selanjutnya adalah menuangkan seluruh informasi mengenai kebutuhan pengguna dan analisis yang telah dilakukan kedalam desain perangkat lunak. Berbagai alat seperti *Unified Modeling Language* (UML) dan *Entity Relationship Diagram* (ERD) dapat digunakan untuk mendesain sistem.

5. Pengembangan Perangkat lunak

Fase ini juga dinamakan fase pemrograman atau pengkodean. Pengembangan perangkat lunak dimulai dengan menulis kode program ke dalam bahasa pemrograman yang sesuai. Perangkat lunak biasanya terintegrasi dengan *libraries*,

basis data, dan program lainnya.

6. Pengujian Perangkat Lunak Pengujian perangkat lunak dilakukan untuk menghilangkan kesalahan atau bugs sehingga dapat dihasilkan produk perangkat lunak yang berkualitas.

7. Implementasi Perangkat Lunak

Tahap ini meliputi penginstalan perangkat lunak ke perangkat pengguna. Terkadang perangkat lunak membutuhkan konfigurasi pasca-instalasi. Ini mencakup semua persyaratan perangkat keras dan perangkat lunak untuk menjalankan perangkat lunak yang dikembangkan dan diuji. Ini juga mencakup pelatihan perangkat lunak kepada pengguna agar bekerja secara efisien.

8. Pemeliharaan Perangkat Lunak

Tujuan pemeliharaan perangkat lunak untuk menghilangkan kesalahan atau *bugs*, mengubah kebutuhan sistem dan menambah fitur baru ke perangkat lunak. Pemeliharaan terhadap perangkat lunak membuat perangkat lunak lebih andal.

Waterfall model, *spiral model*, *incremental model*, *prototyping model*, dan *agile model* merupakan beberapa model SDLC. Pada penelitian ini akan digunakan *spiral model* sebagai model pengembangan perangkat lunak. Dipilihnya model spiral dikarenakan fleksibilitasnya terhadap kemungkinan perubahan sepanjang siklus hidup pengembangan perangkat lunak. Spiral merupakan model pengembangan evolusioner artinya dapat mengakomodasi kebutuhan yang berubah-ubah atau berevolusi terus menerus. Proses pengembangan evolusioner memungkinkan perekrut perangkat lunak mengembangkan suatu produk perangkat lunak menjadi versi yang lebih lengkap secara bertahap [22].

1. *Spiral Model*

Pada awalnya diusulkan oleh Barry Boehm (1988), spiral model merupakan model proses perangkat lunak yang evolusioner yang merangkai sifat iteratif dari *prototype* dan aspek sistematis dari model sekuensial linier. *Spiral model* terdiri dari beberapa tahapan, tahapan-tahapan tersebut adalah [21]:

1. Komunikasi

Merupakan tahapan untuk membangun komunikasi yang efektif di antara pengembang dan pelanggan

2. Perencanaan

Mendefinisikan sumber-sumber daya, ketepatan waktu, dan informasi lain yang berhubungan

3. Desain

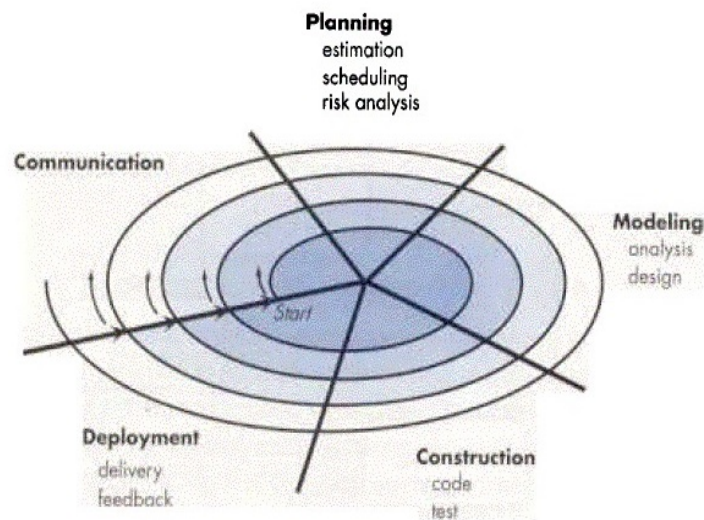
Membangun satu atau lebih representasi dari sistem.

4. Konstruksi

Pada tahap ini dilakukan pembangunan dan pengujian perangkat lunak yang dimaksud

5. Evaluasi

Tahap evaluasi dilakukan untuk memperoleh umpan balik dari pelanggan didasarkan pada evaluasi representasi perangkat lunak.



Gambar 2.1: *Spiral Model*

Ketika proses pengembangan dimulai, tim rekayasa perangkat lunak bergerak searah jarum mengelilingi spiral tersebut dengan dimulai dari intinya. Lintasan pertama putaran spiral menghasilkan perkembangan dari spesifikasi produk; putaran spiral selanjutnya mungkin dipakai untuk mengembangkan sebuah *prototype* dan secara progresif mengembangkan versi perangkat lunak yang lebih lengkap [21].

D. Unified Modeling Language (UML)

Salah satu tahap dalam pengembangan sistem adalah tahap desain dimana pada tahap ini semua informasi kebutuhan pengguna dan analisis sistem dituangkan ke dalam suatu desain perangkat lunak. Mendesain suatu produk perangkat lunak salah satunya dapat menggunakan pemodelan berorientasi objek (OOP). Menurut Haviluddin dalam *Jurnal Informatika Mulawarman Vol 6 No. 1 (2011)*, *Unified Modelling Language* merupakan alat perancangan sistem yang berorientasi pada objek. Secara filosofi UML diilhami oleh konsep yang telah ada yaitu konsep pemodelan *Object Oriented* karena konsep ini menganalogikan sistem seperti kehidupan nyata yang didominasi oleh objek dan digambarkan atau dinotasikan dalam simbol-simbol yang

cukup spesifik [12].

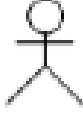





Menurut Ir. M. Farid Azis, M.Kom (2005) berpendapat bahwa UML adalah sekumpulan simbol dan diagram untuk memodelkan perangkat lunak. Desain dalam bentuk simbol dan diagram, kemudian diterjemahkan menjadi kode program. Implementasi kode program dari diagram UML dapat menggunakan bahasa pemrograman apa saja dengan syarat bahasa pemrograman tersebut harus mendukung pemrograman berorientasi objek (OOP) [4].

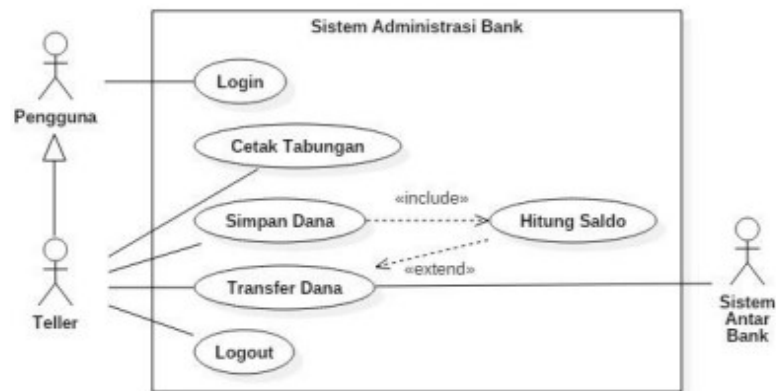
Berdasarkan kedua pendapat di atas dapat disimpulkan bahwa UML adalah suatu alat yang digunakan untuk memodelkan atau mendesain perangkat lunak dalam bentuk simbol dan diagram dimana sistem dari perangkat lunak tersebut menggunakan pemrograman berorientasi objek (OOP). Jenis UML yang dipergunakan untuk penelitian ini adalah *use case diagram*, *class diagram*, dan *activity diagram*.

1. Use Case Diagram

Menurut Ade Hendini dalam *Jurnal Khatulistiwa Informatika*, Vol. IV, No. 2 (2016), *Use Case Diagram* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut [13]. Simbol-simbol yang digunakan dalam *Use Case Diagram* yaitu:

Tabel 2.1: Simbol-simbol *Use Case Diagram* [13]

No	Nama Simbol	Fungsi	Notasi
1	<i>Actor</i>	Adalah <i>abstraction</i> dari orang atau sistem yang mengaktifkan fungsi dari target sistem	 Actor/Role
2	<i>Use Case</i>	Mewakili sebuah bagian dari fungsionalitas yang disediakan sistem	 Use Case
3	<i>Association</i>	Menghubungkan <i>actor</i> untuk berinteraksi dengan <i>use case</i>	
4	<i>Generalization</i>	Hubungan generalisasi dan spesialis (umum-khusus)	
5	<i>Include</i>	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya	
6	<i>Extend</i>	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> yang ditambahkan dapat berdiri sendiri walaupun tanpa <i>use case</i> tambahan tersebut	




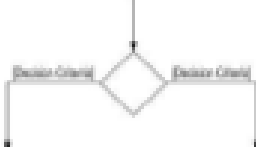



Gambar 2.2: Contoh Use Case Diagram

2. Activity Diagram

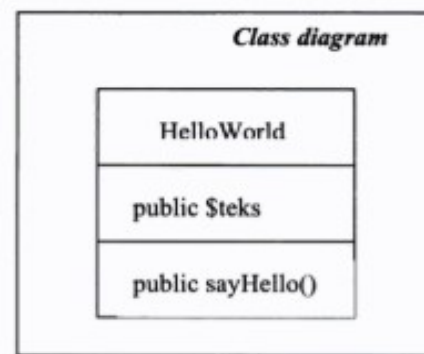
Activity Diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis [13]. Berikut simbol dari activity diagram:

Tabel 2.2: Simbol-simbol *Activity Diagram* [13]

No	Nama Simbol	Fungsi	Notasi
1	Start Point	Merupakan awal dari aktivitas	
2	End Point	Merupakan akhir dari aktivitas	
3	Activities	Menggambarkan suatu proses/kegiatan bisnis	
4	Decision point	Menggambarkan pilihan untuk pengambilan keputusan	
5	Swimlane	Pembagian activity diagram untuk menunjukkan siapa melakukan apa	

3. *Class Diagram*

Class diagram menggambarkan *class* dan hubungan antar-*class* di dalam sistem. *Class* digambarkan dengan sebuah kotak dibagi menjadi tiga bagian. Bagian paling atas diisi nama *class*, bagian tengah diisi variabel yang dimiliki *class* dan bagian bawah diisi *method* dari *class* [4].



Gambar 2.3: Contoh sederhana Class Diagram

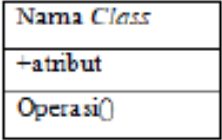



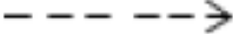


Class Diagram secara khas meliputi: Kelas (*Class*), Relasi Asosiasi/*Associations*, Generalisasi/*Generalization* dan Agregasi/*Aggregation*, atribut (*Attributes*), operasi (*operation/method*), dan hubungan antar kelas mempunyai keterangan yang disebut dengan *Multiplicity* atau *Cardinality*.

Tabel 2.3: *Multiplicity Class Diagram*

<i>Multiplicity</i>	<i>Penjelasan</i>
1	Satu dan hanya satu
0..*	Boleh tidak ada atau 1 atau lebih
1..*	1 atau lebih
0..1	Boleh tidak ada, maksimal 1
n..m	Batasan antara, minimal n maksimal m

Berikut simbol-simbol yang terdapat pada *class diagram* beserta deskripsinya:

Tabel 2.4: Simbol-simbol *Class Diagram* [3]

No	Simbol	Deskripsi
1	<p>Kelas/class</p> 	Kelas pada struktur sistem
2	<p>Asosiasi/association</p> 	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
3	<p>Asosiasi berarah/directed association</p> 	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
4	<p>Generalisasi/generalization</p> 	Relasi antar kelas dengan makna umum khusus
5	<p>Kebergantungan/dependency</p> 	Relasi antar kelas dengan makna kebergantungan antar kelas.
6	<p>Agregasi/aggregation</p> 	Relasi antar kelas dengan makna suatu <i>class</i> merupakan bagian dari <i>class</i>
7	<p>Komposisi/composition</p> 	Bermakna bahwa sebuah <i>class</i> tidak dapat berdiri sendiri dan merupakan bagian dari <i>class</i> lain

E. *Entity Relationship Database(ERD)*

Komponen utama dari suatu sistem informasi adalah data, data akan diolah menjadi suatu informasi yang digunakan untuk pengambilan keputusan. Salah satu cara dalam memodelkan suatu data ialah dengan menggunakan *Entity Relationship Diagram (ERD)*. *Entity Relationship Diagram (ERD)* adalah sekumpulan cara atau peralatan untuk mendeskripsikan data-data atau objek-objek yang dibuat berdasarkan dan berasal dari dunia nyata yang disebut entitas (*entity*) serta hubungan (*relationship*) antar entitas-entitas tersebut [6].

Simbol-simbol dalam ERD (*Entity Relationship Diagram*) adalah sebagai berikut [9].

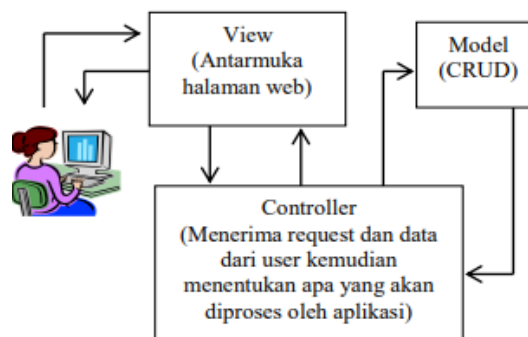
1. Entitas: suatu yang nyata atau abstrak yang mempunyai karakteristik dimana kita akan menyimpan data.
2. Atribut: ciri umum semua atau sebagian besar instansi pada entitas tertentu.
3. Relasi: hubungan alamiah yang terjadi antara satu atau lebih entitas. Jenis relasi yang ada pada ERD, yaitu one-to-one, one-to-many, dan many-to-many.

F. *Model View Controller (MVC)*

MVC merupakan arsitektur yang membagi aplikasi menjadi tiga bagian secara konsep yang terpisah yaitu *Model*, *View*, dan *Controller*, masing-masing dapat dikembangkan secara terpisah antara satu dengan yang lainnya, sehingga perubahan pada satu bagian memiliki dampak minimal pada bagian lain. Bagian model digunakan untuk mendefinisikan suatu cara dimana data dapat diakses, bagian view menghasilkan keluaran jika diberikan data, dan bagian controller menerima perintah dan mengatur aplikasi untuk tugas dan tampilan yang sesuai [15].

Untuk lebih jelasnya berikut tiga komponen yang membangun MVC [11] :

1. *Model*, digunakan untuk mengelola informasi atau data dan merepresentasikannya kepada pengguna. Pada umumnya, *Model* berisi fungsi-fungsi yang berhubungan dengan *database*, seperti pengambilan data, *update* data, *insert* data, *delete* data, dan lain sebagainya.
2. *View*, merupakan informasi yang direpresentasikan kepada pengguna. *View* biasanya berupa halaman web, dimana pengguna dapat melihat informasi yang ditampilkan.
3. *Controller*, memberikan pelayanan yang menjembatani *Model* dan *View*. *Controller* berisi fungsi-fungsi yang dapat membantu menjembatani *Model* dan *View*. Request datang dan di-*response* melalui *Controller*, kemudian *Controller* berkomunikasi serta melakukan kontrol terhadap *View* dan *Model*.



Gambar 2.4: Alur Kerja MVC

MVC dimulai pada saat pengguna memulai mengakses aplikasi. Awalnya pengguna akan menjalankan operasi pada bagian *View*. Proses ini dapat berupa proses *Login*, Pendaftaran ataupun proses lainnya. Kemudian perintah akan direspon oleh bagian *Controller*. Bagian *Controller* akan menentukan apakah permintaan dari pengguna akan diproses atau tidak, ketika diproses maka permintaan akan diarahkan ke bagian *Model*. Setelah permintaan dari pengguna dapat dipenuhi oleh *model* maka

informasi akan dikirimkan kembali ke bagian *controller* lalu diberikan ke bagian *view* sehingga pengguna dapat memperoleh informasi sesuai yang diinginkan [17].

G. Basis Data

Basis data terdiri dari dua kata, yaitu basis dan data. Basis dapat diartikan sebagai markas, gudang, dan tempat berkumpul. Sedangkan data adalah fakta yang mewakili suatu objek seperti manusia, barang hewan, peristiwa dan sebagainya, yang direkam dalam bentuk angka, huruf, simbol, teks, gambar, bunyi atau kombinasinya. Menurut Robi Yanto, M.Kom basis data dapat didefinisikan sebagai himpunan kelompok data yang saling berhubungan yang diorganisasi sedemikian rupa agar dapat dimanfaatkan kembali dengan cepat dan mudah [29].

Sedangkan menurut Edhy Sutanta basis data dapat dipahami sebagai suatu kumpulan data terhubung (*interrelatet data*) yang disimpan secara bersama-sama pada suatu media, data disimpan dengan cara-cara tertentu sehingga mudah untuk digunakan atau ditampilkan kembali; data dapat digunakan oleh satu atau lebih program-program aplikasi secara optimal; data disimpan sedemikian rupa sehingga proses penambahan, pengambilan, dan modifikasi data dapat dilakukan dengan mudah dan terkontrol [28].

Berdasarkan dari kedua pendapat tersebut, basis data dapat dikatakan merupakan sekumpulan kelompok data yang saling terhubung satu sama lain, disimpan dan diatur sedemikian rupa pada suatu media agar data dapat diambil dan dikelola dengan mudah.

Sistem adalah sekumpulan komponen-komponen yang saling berhubungan dan secara bersama-sama bertujuan untuk memenuhi suatu pekerjaan tertentu. Komponen penting dalam sistem basis data adalah [29]:

1. Data

Merupakan informasi yang disimpan dalam suatu struktur tertentu yang terintegrasi

2. *Hardware*

Merupakan perangkat keras berupa komputer dengan media penyimpanan data karena pada umumnya basis data memiliki ukuran yang besar

3. Sistem Operasi

Program yang mengaktifkan dan memfungsikan sistem komputer, mengendalikan seluruh sumber daya dalam komputer, dan melakukan operasi dasar dalam komputer meliputi input, proses, dan output.

4. Basis Data

Sebagai inti dari sistem basis data. Basis data menyimpan data serta struktur sistem basis data baik untuk entitas maupun objek-objek secara detail.

5. *Database Management System*

Merupakan perangkat lunak yang digunakan untuk melakukan pengelolaan basis data sebagai contoh *Microsoft access, Sql Server, Mysql*, dan *Oracle*.

6. *User*

Merupakan pengguna yang menggunakan data yang tersimpan dan terkelola. *User* dapat berupa seseorang yang mengelola basis data yang disebut database administrator (DBA), bisa juga disebut *end user*.

Basis data dapat digolongkan berdasarkan beberapa kriteria, salah satunya ialah berdasarkan seberapa sering basis data mengalami perubahan dikenal dua pengkategorian yaitu [27] :

1. Tabel/data master adalah tabel yang datanya cenderung jarang berubah dan tidak memiliki ketergantungan dengan tabel lain.
2. Tabel/data transaksi adalah tabel yang datanya sering berubah dan membutuhkan data master untuk membangun komponennya

BAB III

IMPLEMENTASI PROGRAM

Secara umum tahapan pengembangan sistem dalam metode *System Development Life Cycle* (SDLC) adalah analisis kebutuhan perangkat lunak, desain sistem, konstruksi atau pengembangan, dan pengujian atau evaluasi.

A. Analisis Kebutuhan

Kebutuhan mengenai sistem didapat berdasarkan hasil analisis perbandingan sistem *tracer study* yang sudah ada yaitu, Sistem *Tracer Study* Dikti, dan Sistem Informasi *Tracer Study* Universitas Negeri Jakarta [Lampiran A]. Selain itu, melalui pengumpulan data yang diperoleh dari analisis kebutuhan prodi Ilmu Komputer [Lampiran C]. Berikut beberapa kebutuhan yang didapat:

1. Memiliki lima pengguna, yaitu admin, koorprodi, alumni, dan pengunjung yang memiliki sub-aktor pengguna alumni.
2. Admin dapat mengelola data alumni dan pengguna alumni, membuat kuesioner, menyunting konten beranda, menampilkan hasil *tracer study*, dan menampilkan statistik jumlah lulusan Ilmu Komputer
3. Koorprodi dapat menampilkan data alumni dan pengguna alumni, dapat menampilkan hasil *tracer study*, dan menampilkan statistik jumlah lulusan Ilmu Komputer
4. Alumni dapat mengelola data diri dan mengisi kuesioner yang telah dibuat oleh admin

5. Pengunjung dapat melakukan pencarian alumni, melihat daftar pengguna alumni, dan melihat data hasil tracer.
6. Pengguna alumni dapat mengisi form kuesioner.

B. Desain Sistem

Pada tahap ini perancangan dan pemodelan sistem dituangkan dalam bentuk visual mulai dari *Use Case Diagram*, *Activity Diagram*, *Class Diagram*, *Entity Relationship Diagram* dan *mock-up* atau rancangan dari tampilan sistem.

1. Use Case Diagram

Use Case digunakan untuk mengetahui fungsi apa saja yang terdapat di dalam suatu sistem dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut. Pada sistem ini terdapat lima aktor yaitu admin, koorprodi, alumni, dan penunjang yang memiliki sub-aktor pengguna alumni. Peran-peran aktor tersebut adalah sebagai berikut: 7

1. Admin

- Admin dapat mengelola data alumni
- Admin dapat mengelola data pengguna alumni
- Admin dapat membuat form kuesioner *tracer study*
- Admin dapat menyunting kata pengantar *tracer study* pada beranda alumni
- Admin dapat melihat hasil *tracer study*

2. Koorprodi

- Koorprodi dapat melihat data alumni
- Koorprodi dapat melihat data pengguna alumni
- Koorprodi dapat melihat hasil *tracer study*

3. Alumni

- Alumni dapat mengelola data diri
- Alumni dapat mengisi form kuesioner *tracer study*
- Alumni dapat melihat daftar pengguna alumni beserta alumni yang bekerja

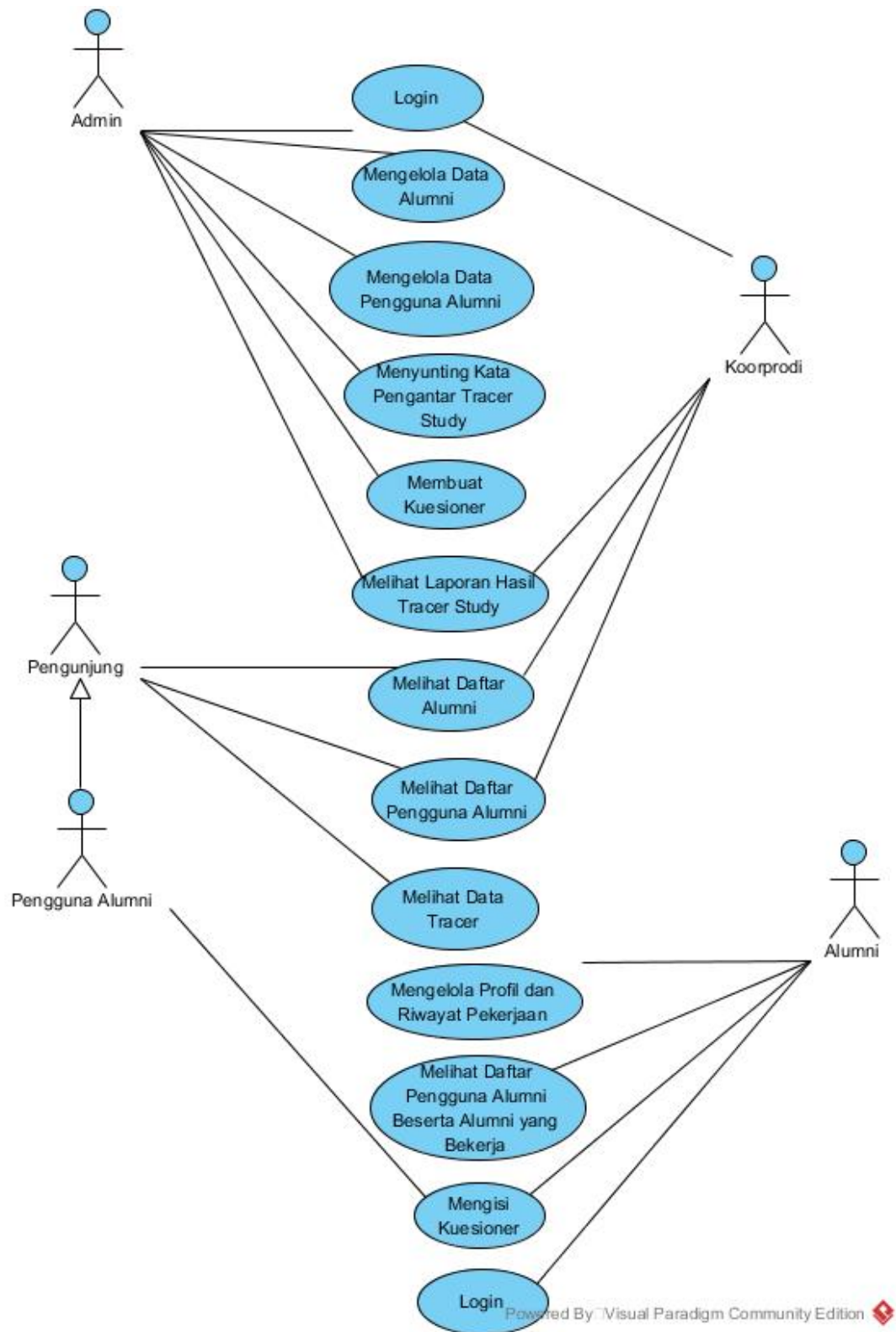
4. Pengunjung

- Pengunjung dapat melihat daftar alumni
- Pengunjung dapat melihat daftar pengguna alumni
- Pengunjung dapat melihat data *tracer*

5. Pengguna Alumni

- Pengguna Alumni dapat mengisi form kuesioner

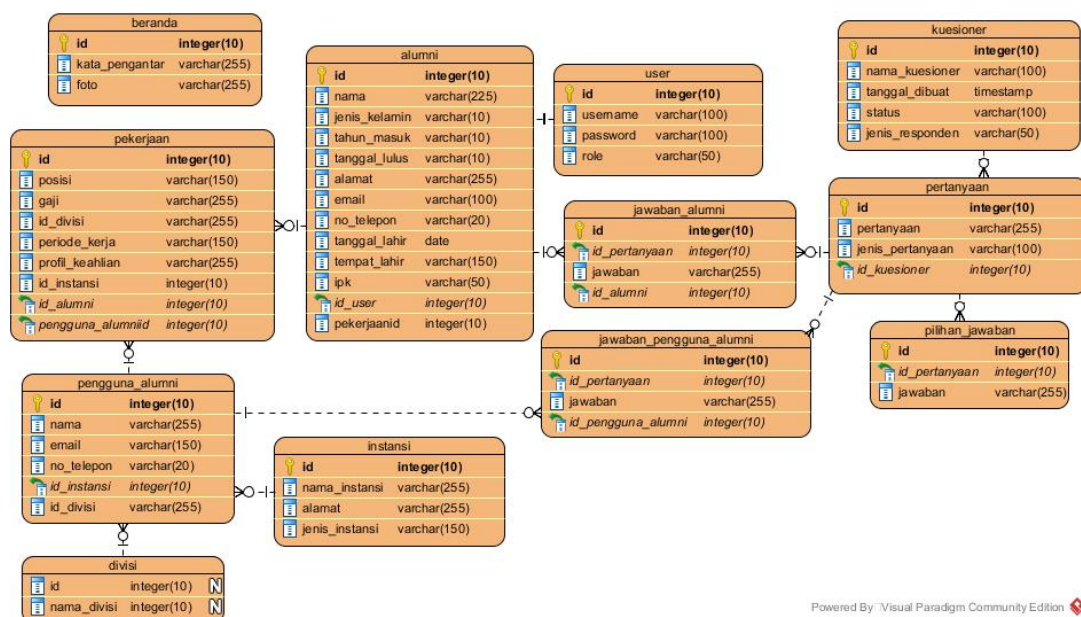
Berikut adalah *Use Case Diagram* dari Sistem Informasi *Tracer Study* Ilmu Komputer FMIPA UNJ



Gambar 3.1: Use Case Diagram Sistem Informasi Tracer Study Ilmu Komputer

2. Entity Relationship Diagram

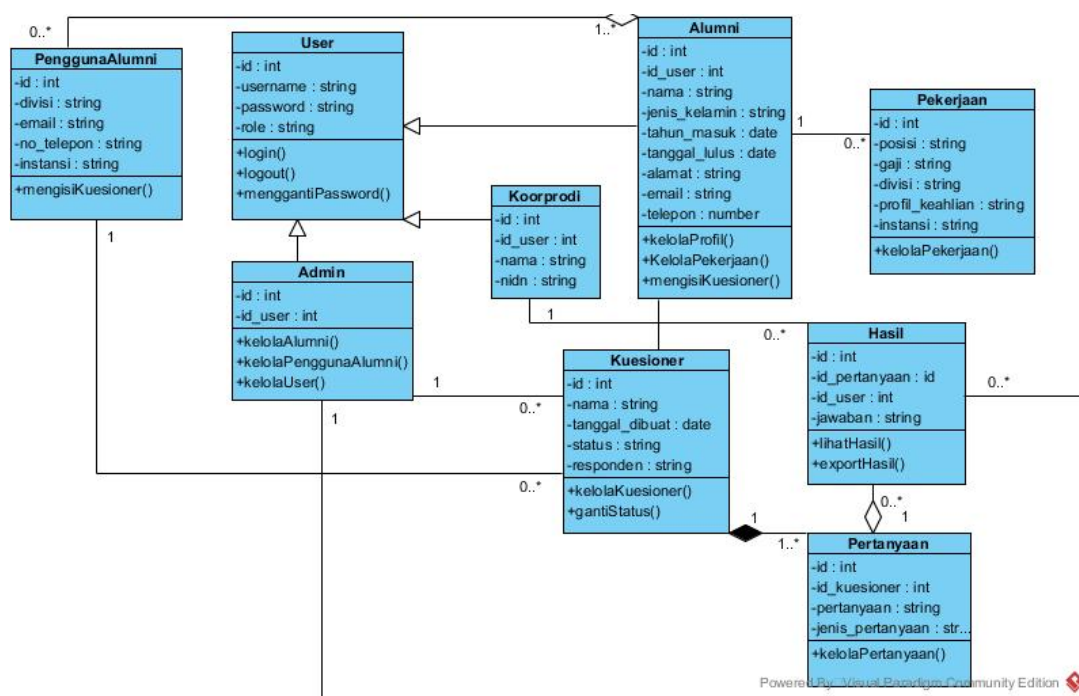
ERD dibawah ini menggambarkan masing-masing entitas dan relasi antar entitas untuk basis data Sistem Informasi *Tracer Study* Ilmu Komputer yang akan menyimpan data yang dibutuhkan sistem di antaranya data alumni, pengguna alumni, kuesioner dan hasil kuesioner. Pada ERD ini dibuat tabel pekerjaan dimana berfungsi untuk menampung data riwayat pekerjaan setiap alumni dan tabel instansi untuk menyimpan perusahaan-perusahaan tempat alumni bekerja. Kedua tabel tersebut berelasi dengan tabel alumni dan pengguna alumni. Tabel kuesioner berfungsi untuk menampung kuesioner baik kuesioner alumni maupun pengguna alumni dan berelasi dengan tabel pertanyaan untuk menyimpan pertanyaan-pertanyaan kuesioner. Selain itu disediakan tabel jawaban untuk menampung jawaban kuesioner. Berikut pemoodelan ERD sistem informasi *Tracer Study* Ilmu Komputer.



Gambar 3.2: Entity Relationship Diagram Sistem Informasi *Tracer Study* Ilmu Komputer

3. Class Diagram

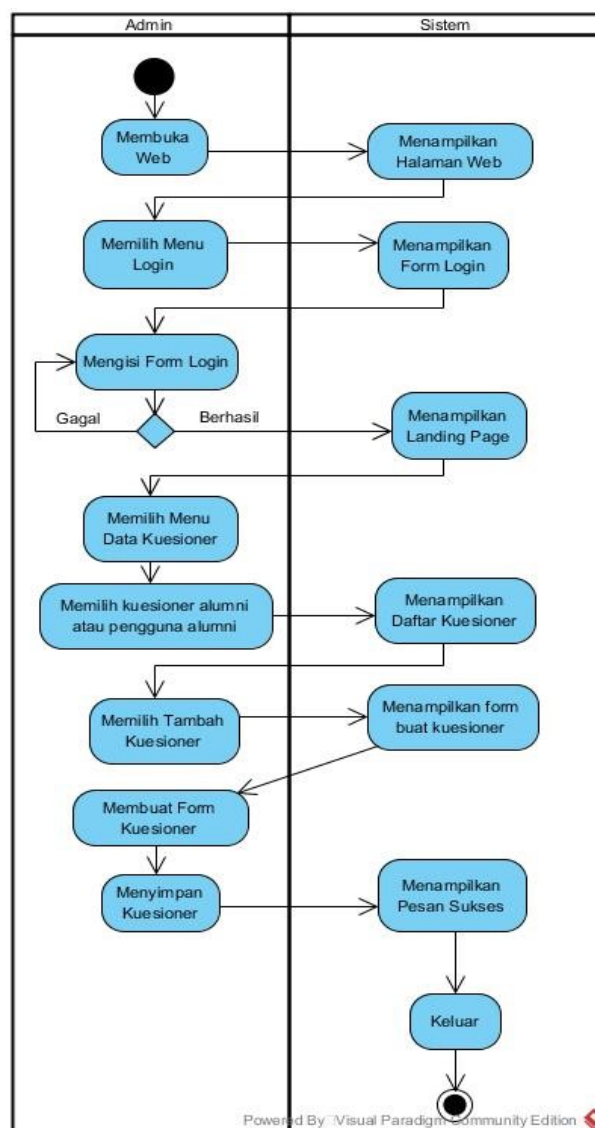
Pemodelan *Class Diagram* pada Sistem Informasi *Tracer Study* Ilmu Komputer memiliki sembilan *Class*. Terdapat *Class user* yang menyimpan metode *login*, *logout*, dan mengganti *password*. *Class* ini memiliki tiga *subclass* dimana dapat mengakses metode dan atribut dari *Class* tersebut yaitu *admin*, *koorprodi*, dan *alumni*. *Class* kuesioner untuk menampung kuesioner dan berelasi dengan *Class* pertanyaan yang berisi metode untuk mengelola pertanyaan kuesioner. Selanjutnya terdapat *Class* hasil berfungsi untuk mengelola dan menampilkan hasil kuesioner dan *Class* pekerjaan untuk mengelola data pekerjaan alumni. Berikut adalah *Class Diagram* dari sistem Informasi *Tracer Study* Ilmu Komputer



Gambar 3.3: Desain *Class Diagram* Sistem Informasi *Tracer Study* Ilmu Komputer

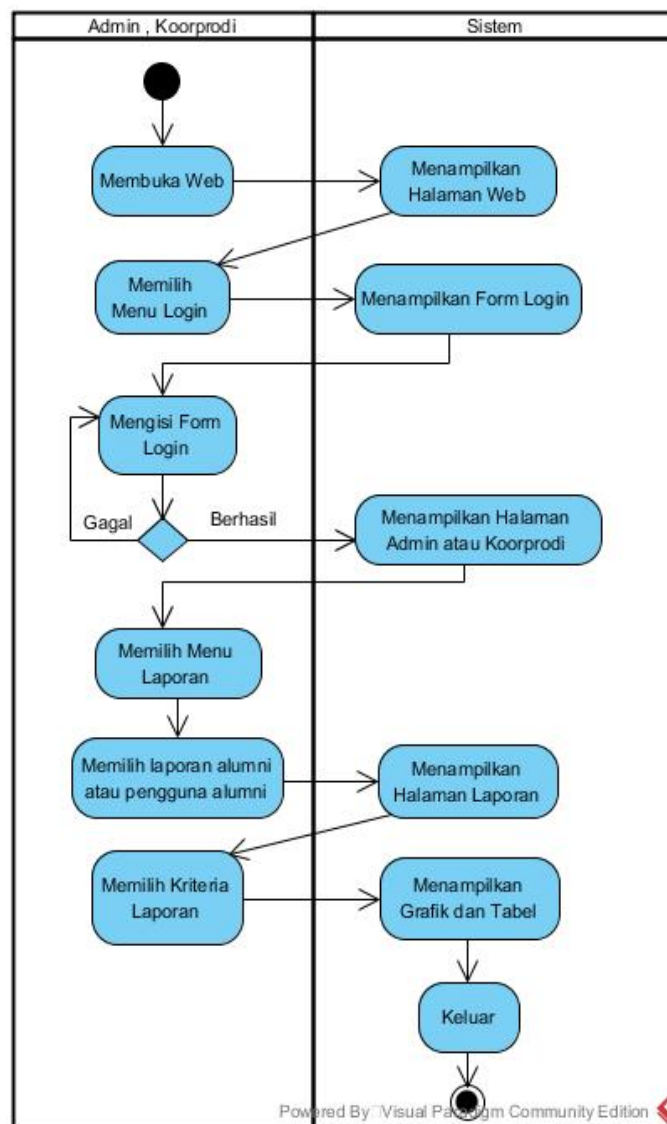
4. Activity Diagram

Desain *Activity Diagram* pada sistem ini dibagi menjadi lima diagram, yaitu membuat form kuesioner untuk admin, melihat hasil atau laporan *tracer study* untuk admin dan koorprodi, mengisi riwayat pekerjaan untuk alumni dan mengisi form kuesioner untuk alumni dan pengguna alumni. Berikut ini *Activity Diagram* dari Sistem Informasi *Tracer Study* Ilmu Komputer



Gambar 3.4: Activity Diagram Membuat Kuesioner Tracer Study

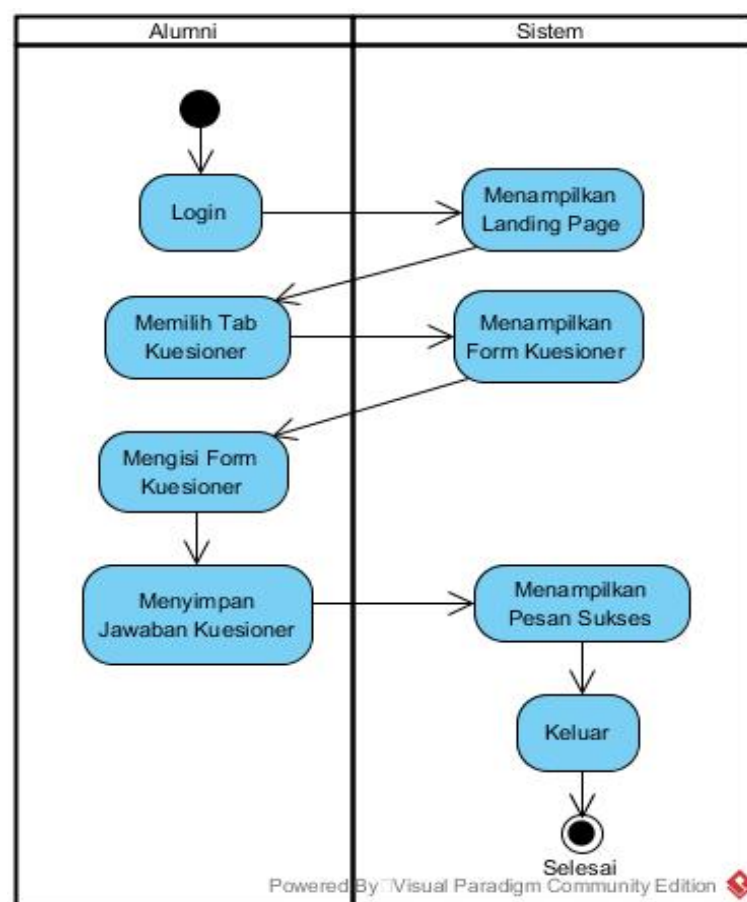
Gambar 3.4 menggambarkan aktivitas dalam membuat form kuesioner baik untuk kuesioner alumni maupun pengguna alumni yang dilakukan oleh admin. Dalam pembuatan form kuesioner, admin dapat menambahkan pertanyaan kuesioner dimana disediakan empat jenis pertanyaan, yaitu pertanyaan isian, pilihan, pilihan ganda, dan pertanyaan dengan jawaban skala.



Gambar 3.5: Activity Diagram Melihat Laporan Tracer Study

Pada Activity Diagram melihat laporan, admin dan koorprodi dapat melihat

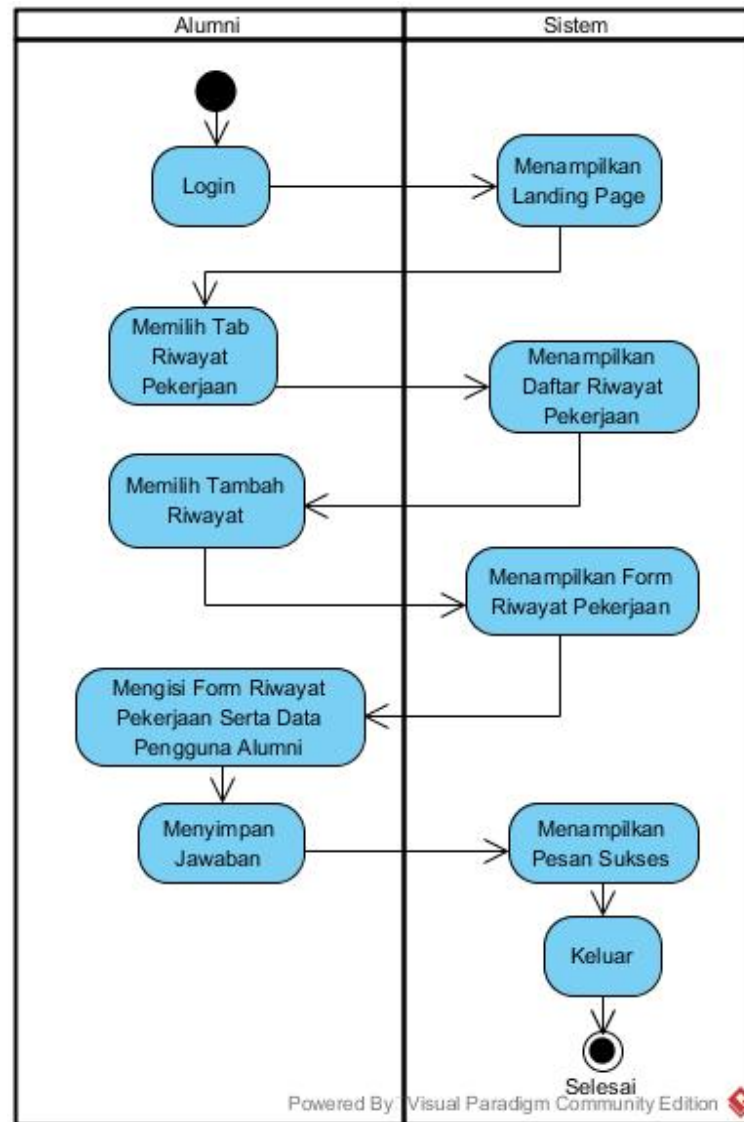
laporan hasil *tracer study* baik alumni dan pengguna alumni. Sebelum menampilkan laporan, admin dan koorprodi mengisi kriteria yang dipilih yaitu berdasarkan jenis kuesioner, pertanyaan yang ingin dilihat hasilnya, dan juga tahun lulus, kemudian sistem akan menampilkan hasil *tracer study* dalam bentuk grafik dan tabel. Selain itu, pada halaman hasil *tracer study* admin dan koorprodi dapat mengekspor data *tracer* dalam format *excel*.



Gambar 3.6: Activity Diagram Mengisi Kuesioner Alumni

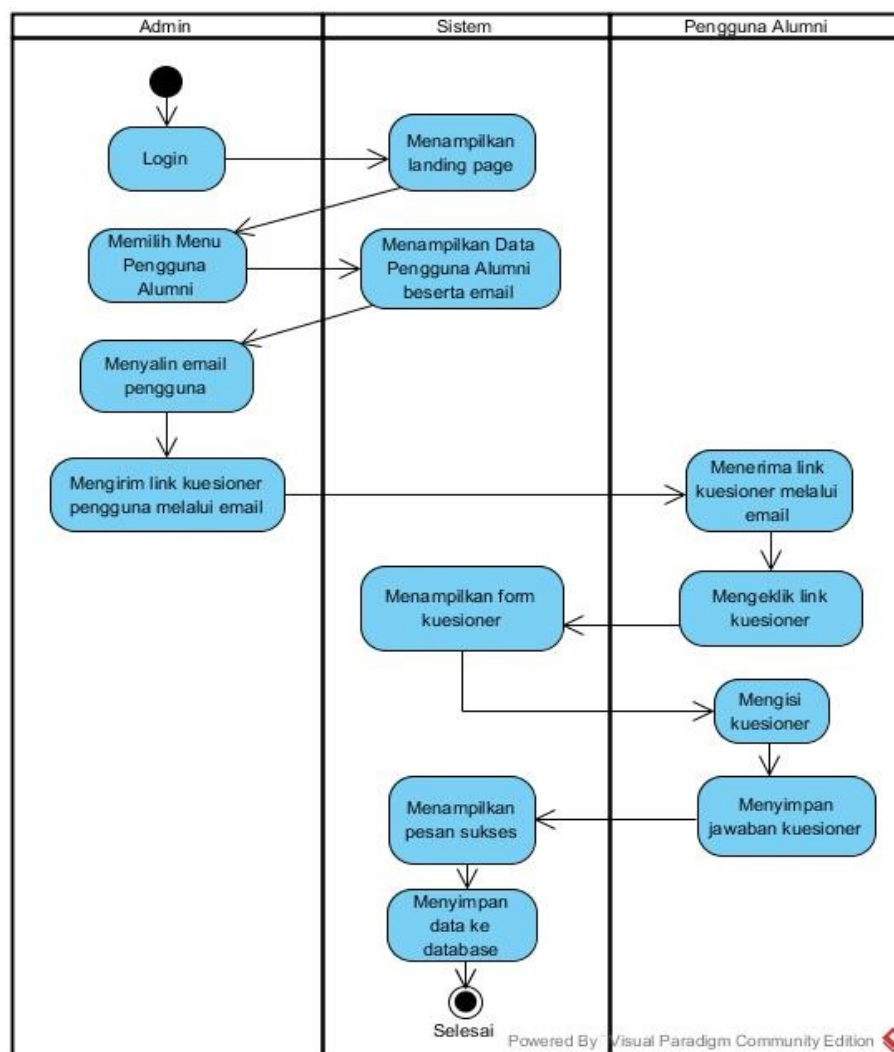
Gambar 3.6 menjelaskan aktivitas mengisi kuesioner yang dilakukan oleh alumni. Pada sistem ini juga disediakan form riwayat pekerjaan agar terlihat rekam jejak dari setiap pekerjaan alumni yang pernah digeluti dan juga untuk mengisi data mengenai pengguna alumni. Desain tampilan dari aktivitas mengisi riwayat pekerjaan

untuk alumni sebagai berikut.

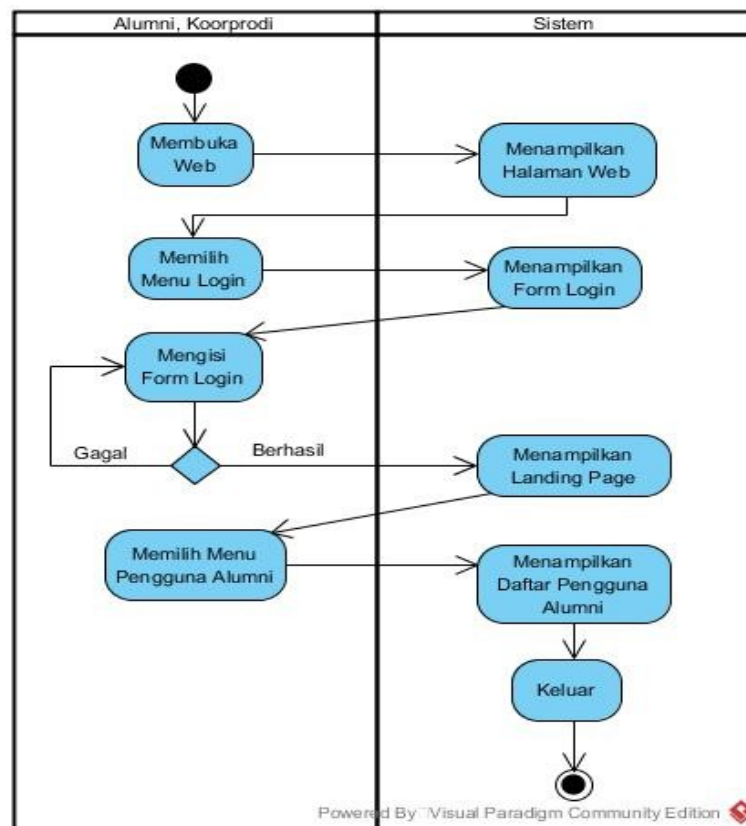


Gambar 3.7: Activity Diagram Mengisi Riwayat Pekerjaan Alumni

Pada sistem *tracer study* ini pengguna alumni juga dapat mengisi kuesioner. Alur dari aktivitas pengisian kuesioner oleh pengguna alumni dapat dilihat pada gambar berikut.



Gambar 3.8: Activity Diagram Mengisi Kuesioner Pengguna Alumni



Gambar 3.9: Activity Diagram Melihat Daftar Pengguna Alumni

Alumni dan koorprodi dapat melihat daftar pengguna alumni, alur dari aktivitas tersebut dapat dilihat pada diagram di atas. Alumni dan koorprodi juga dapat melihat siapa saja alumni yang bekerja pada setiap pengguna alumni.

5. Rancangan Antar Muka Program

Rancangan antar muka atau desain tampilan sistem dapat disebut sebagai *mock-up*. Tampilan awal sistem akan menampilkan halaman pengunjung dimana disediakan menu daftar pengguna alumni dan statistik alumni. Di bawah ini adalah *mock-up* untuk tampilan daftar pengguna alumni dimana pengunjung dapat melihat daftar pengguna alumni Ilmu Komputer beserta posisi alumni yang bekerja di instansi tersebut.

Tracer Study Ilmu Komputer

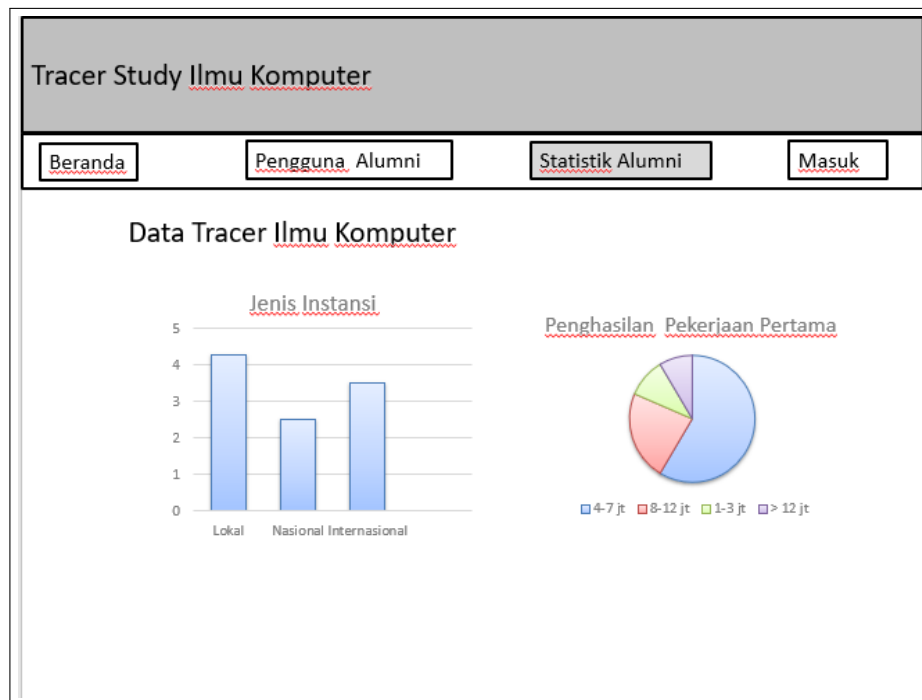
Beranda Pengguna Alumni Statistik Alumni Masuk

Pengguna Alumni Ilmu Komputer

No	Instansi	Divisi	Alamat	Posisi Alumni

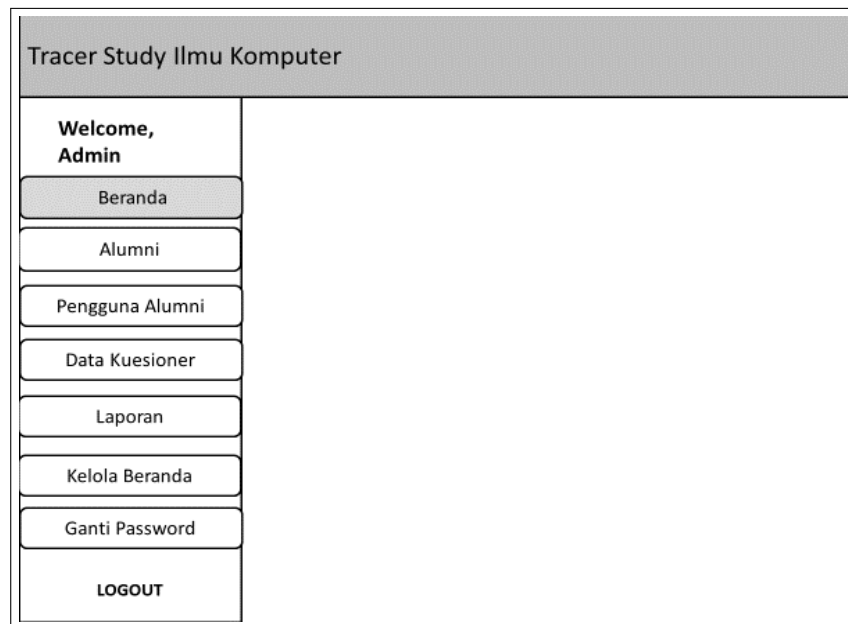
Gambar 3.10: Desain Halaman Daftar Pengguna Alumni untuk Pengunjung

Sistem juga menyediakan halaman statistik alumni untuk menampilkan data *tracer* yang dapat dilihat oleh pengunjung. Rancangan dari tampilan halaman tersebut dapat dilihat di bawah ini.



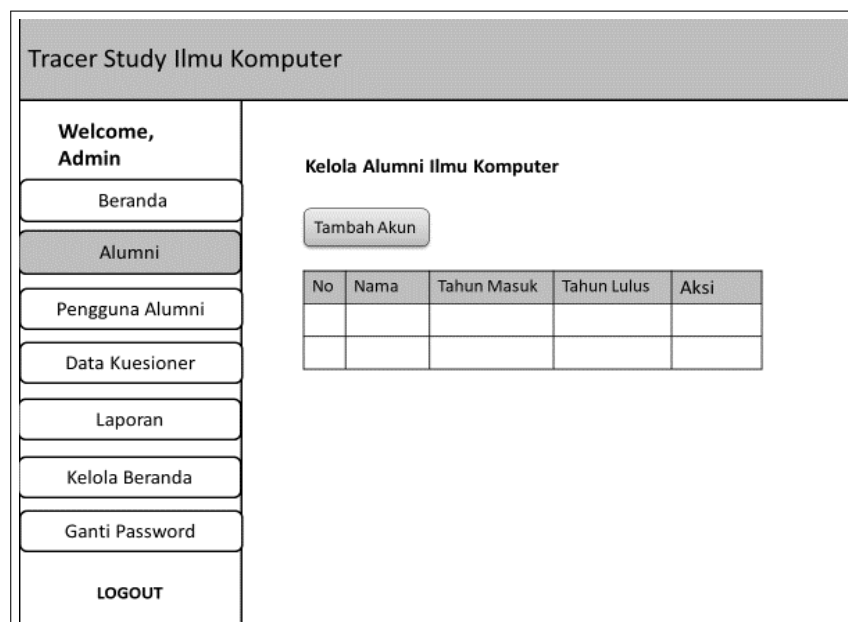
Gambar 3.11: Desain Halaman Statistik Alumni untuk Pengunjung

Di bawah ini adalah rancangan tampilan dari halaman admin. Pada tampilan admin terdapat beberapa menu yaitu, beranda, alumni, pengguna alumni, kuesioner, laporan, kelola beranda, dan ganti *password*.



Gambar 3.12: Desain Halaman Admin

Admin dapat mengelola data alumni dan pengguna alumni. Berikut ini rancangan tampilan dari kedua halaman tersebut.



Gambar 3.13: Desain Halaman Kelola Data Alumni

Tracer Study Ilmu Komputer

Welcome, Admin

Beranda

Alumni

Pengguna Alumni

Data Kuesioner

Laporan

Kelola Beranda

Ganti Password

LOGOUT

Kelola Pengguna Alumni Ilmu Komputer

No	Nama	Divisi	Instansi	Email	Aksi

Gambar 3.14: Desain Halaman Kelola Data Pengguna Alumni

Untuk mengelola kuesioner baik kuesioner alumni dan pengguna alumni dapat dilakukan pada halaman kelola kuesioner dibawah ini.

Tracer Study Ilmu Komputer

Welcome, Admin

Beranda

Alumni

Pengguna Alumni

Data Kuesioner

Laporan

Kelola Beranda

Ganti Password

LOGOUT

Kelola Kuesioner Alumni/Pengguna Alumni

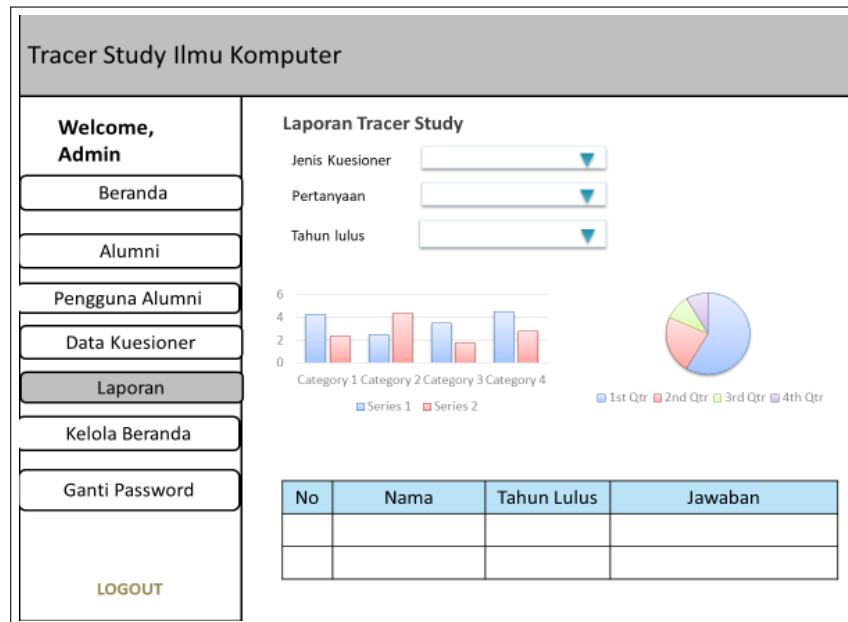
Tambah Kuesioner

No	Kategori Kuesioner	Tanggal Dibuat	Status	Aksi

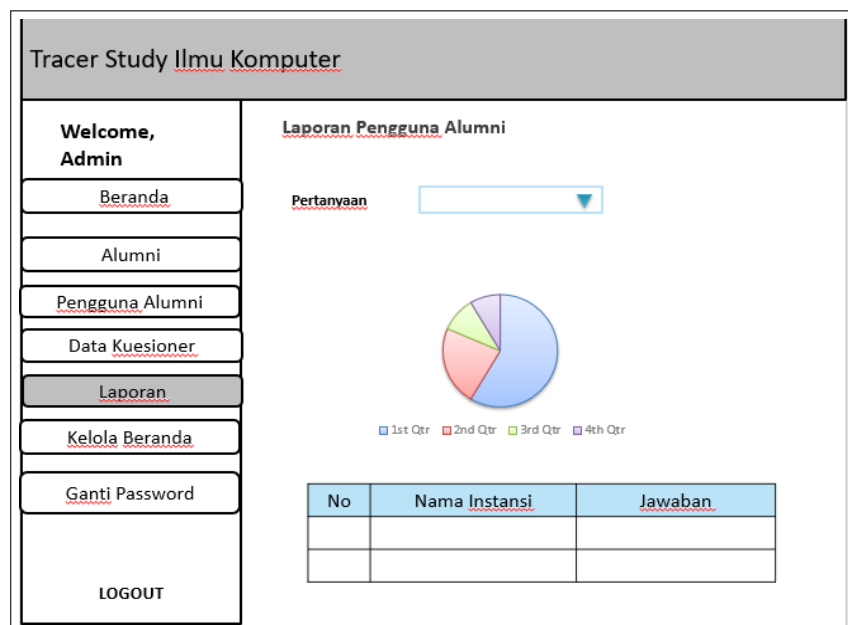
Gambar 3.15: Desain Halaman Kelola Kuesioner

Gambar dibawah merupakan *mock-up* dari tampilan halaman laporan tracer

study. Pada halaman ini hasil *tracer study* ditampilkan melalui grafik dan tabel.

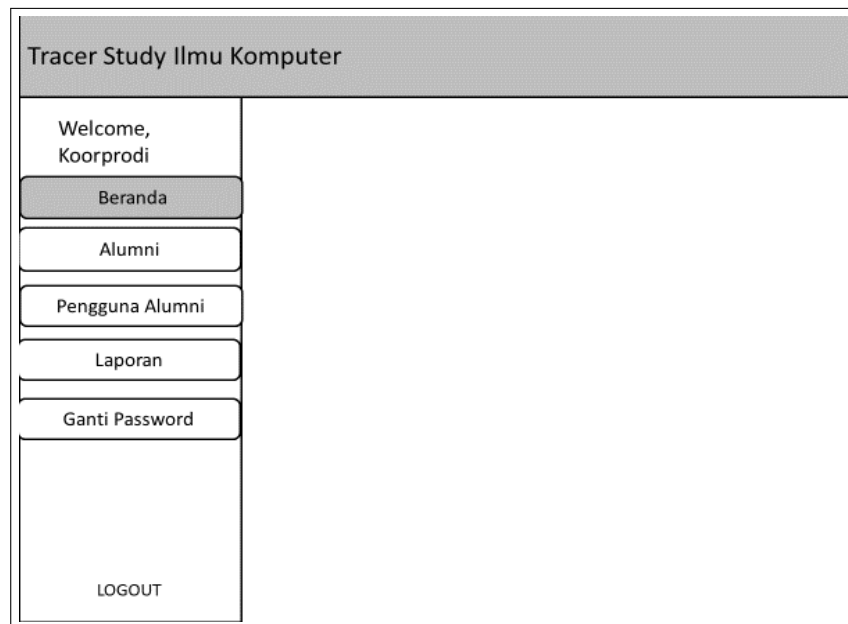


Gambar 3.16: Desain Halaman Laporan *Tracer Study*

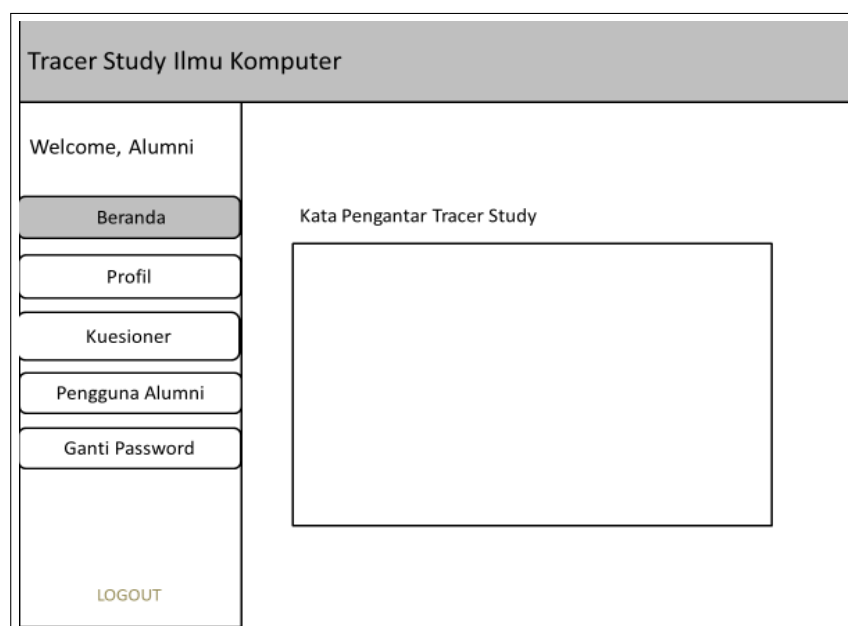


Gambar 3.17: Desain Halaman Laporan Pengguna Alumni

Pada tampilan koorprodi terdapat beberapa menu, yaitu beranda, alumni, pengguna alumni, laporan, dan ganti password.



Gambar 3.18: Desain Halaman Koorprodi



Gambar 3.19: Desain Halaman Alumni

Selanjutnya beralih ke halaman alumni dimana terdapat beberapa menu, yaitu beranda, profil, kuesioner, dan pengguna alumni. Rancangan tampilan halaman alumni dapat dilihat pada gambar 3.19. Alumni dapat mengisi kuesioner melalui halaman

kuesioner. Berikut rancangan tampilan dari halaman kuesioner alumni.

Tracer Study Ilmu Komputer	
<p>Welcome, Alumni</p> <p>Beranda</p> <p>Profil</p> <p>Kuesioner</p> <p>Pengguna Alumni</p> <p>Ganti Password</p> <p>LOGOUT</p>	<p>Form Kuesioner</p> <hr/> <p>Pekerjaan</p> <p>Masa tunggu lulus : <input type="text"/></p> <p>Cara memperoleh Pekerjaan :</p> <p><input type="radio"/> Melalui iklan dikoran/majalah, brosur</p> <p><input type="radio"/> Pergi ke bursa/pameran kerja</p>

Gambar 3.20: Desain Halaman Kuesioner Alumni

Selanjutnya sistem menyediakan tampilan daftar pengguna alumni yang dapat dilihat alumni beserta siapa saja alumni yang bekerja pada instansi dari pengguna alumni terkait. *Mock-up* dari tampilan daftar pengguna alumni dapat dilihat sebagai berikut.

Tracer Study Ilmu Komputer																			
<div> <p>Welcome, Alumni</p> <p>Beranda</p> <p>Profil</p> <p>Kuesioner</p> <p>Pegguna Alumni</p> <p>Ganti Password</p> <p>LOGOUT</p> </div> <div> <p>Daftar Pengguna Alumni</p> <table border="1"> <thead> <tr> <th>No</th> <th>Instansi</th> <th>Divisi</th> <th>Alamat</th> <th>DaftarAlumni</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td></td> <td></td> <td>Lihat</td> </tr> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> </tbody> </table> </div>					No	Instansi	Divisi	Alamat	DaftarAlumni					Lihat					
No	Instansi	Divisi	Alamat	DaftarAlumni															
				Lihat															

Gambar 3.21: Desain Halaman Daftar Pengguna Alumni untuk Alumni

C. Pengembangan

Tahap pengembangan atau juga disebut tahap konstruksi merupakan tahapan dimana menuangkan seluruh desain yang telah dibuat kedalam kode program. tahapan ini dimulai dari membangun basis data, mengimplementasikan desain tampilan dan dilanjutkan dengan membangun sistem.

1. Membangun Basis Data

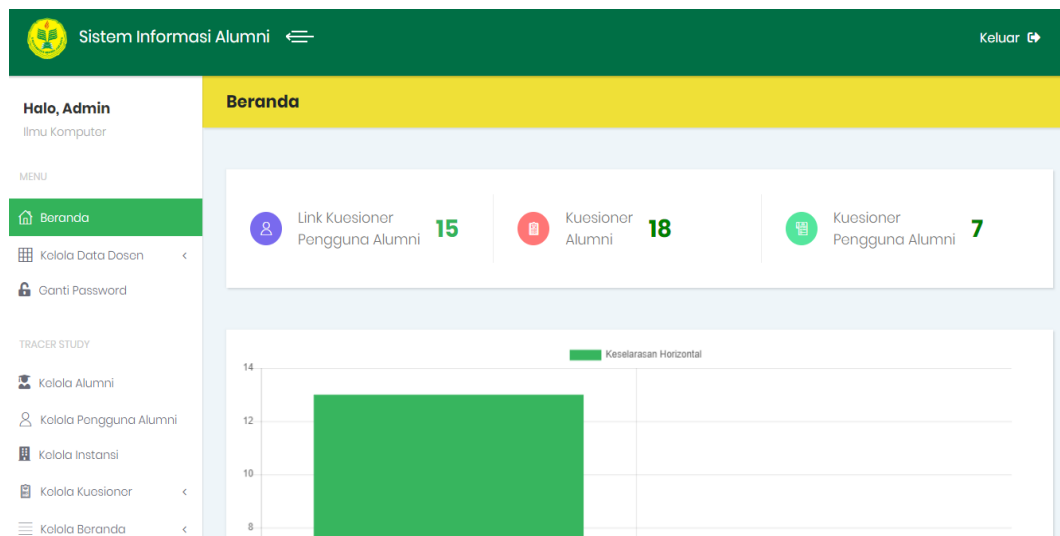
Pada tahap ini dibangun basis data berdasarkan *Entity Relationship Diagram* yang dibuat pada tahap sebelumnya, yaitu desain sistem. Basis data dibuat menggunakan MySQL dan memanfaatkan aplikasi phpMyAdmin untuk membuat tabel beserta relasinya. Berikut bentuk basis data Sistem Informasi *Tracer Study* Prodi Ilmu Komputer yang memiliki 12 tabel:

Table	Action	Rows	Type	Collation	Size	Overhead
alumni	Browse Structure Search Insert Empty Drop	6	InnoDB	latin1_swedish_ci	32 KiB	-
beranda	Browse Structure Search Insert Empty Drop	1	InnoDB	latin1_swedish_ci	16 KiB	-
divisi	Browse Structure Search Insert Empty Drop	8	InnoDB	latin1_swedish_ci	16 KiB	-
instansi	Browse Structure Search Insert Empty Drop	7	InnoDB	latin1_swedish_ci	16 KiB	-
jawaban_alumni	Browse Structure Search Insert Empty Drop	0	InnoDB	latin1_swedish_ci	16 KiB	-
jawaban_pengguna	Browse Structure Search Insert Empty Drop	0	InnoDB	latin1_swedish_ci	16 KiB	-
kuesioner	Browse Structure Search Insert Empty Drop	1	InnoDB	latin1_swedish_ci	32 KiB	-
pekerjaan	Browse Structure Search Insert Empty Drop	3	InnoDB	latin1_swedish_ci	48 KiB	-
pengguna_alumni	Browse Structure Search Insert Empty Drop	9	InnoDB	latin1_swedish_ci	32 KiB	-
pertanyaan	Browse Structure Search Insert Empty Drop	3	InnoDB	latin1_swedish_ci	32 KiB	-
pilihan_jawaban	Browse Structure Search Insert Empty Drop	0	InnoDB	latin1_swedish_ci	32 KiB	-
user	Browse Structure Search Insert Empty Drop	15	InnoDB	latin1_swedish_ci	32 KiB	-
12 table(s)	Sum	53	InnoDB	latin1_swedish_ci	320 KiB	0 B

Gambar 3.22: Basis Data Sistem Informasi *Tracer Study* Prodi Ilmu Komputer

2. Implementasi Desain Tampilan

Untuk mengimplmentasikan desain tampilan, digunakan *Bootstrap* sebagai *framework* untuk membuat tampilan atau *front-end* dan *CodeIgniter* sebagai *framework* untuk *back-end*. Dengan menggunakan *CodeIgniter* memudahkan dalam mengim-plementasikan arsitektur MVC dimana sudah disediakan struktur folder yang sesuai. *Framework Bootstrap* digunakan pada bagian View. Berikut tampilan *website* sistem *tracer study* Ilmu Komputer yang terintegrasi dengan Sistem Informasi Alumni hasil penelitian Ibu Fariani Hermin selaku dosen Ilmu Komputer yang dapat dilihat pada *browser*.



Gambar 3.23: Tampilan Halaman Beranda Admin

Menu *tracer study* terdapat pada bagian bawah kiri seperti pada Gambar 3.23. Gambar tersebut merupakan tampilan halaman beranda admin yang berisi statistik alumni ilmu komputer, jumlah pengguna alumni yang baru terdaftar, jumlah responden kuesioner alumni dan pengguna alumni.

The screenshot shows the Alumni Management (Data Alumni) page with the following data:

No	Nama	NIM	Angkatan	Tahun Lulus
1	Agustinus Purimbaga	3145136223	2013	2017
2	Altinia Prastiantari	3145136208	2013	2017
3	Anantassa Fitri Andini	3145136212	2013	2017
4	Gregorius Andito H	3145136218	2013	2017
5	Hana Maulinda	3145136193	2013	2017

Gambar 3.24: Tampilan Halaman Kelola Alumni pada Admin

Gambar 3.24 adalah tampilan pada admin untuk mengelola data alumni seperti tambah, sunting, hapus, dan *import* data alumni. Selanjutnya, halaman untuk menge-

lola data pengguna alumni pada admin. Berikut tampilan dari halaman tersebut yang dapat dilihat pada Gambar 3.25.

Kelola Data Pengguna Alumni

Perhatian!
Pengisian kuesioner untuk pengguna alumni dilakukan melalui link pada tabel dibawah ini, copy link tersebut kemudian kirimkan melalui email masing-masing pengguna dan untuk pengisian kuesioner bagi pengguna alumni yang belum terdaftar dilakukan melalui link berikut:
[Link Kuesioner Pengguna yang Belum Terdaftar](#)

Pengguna Alumni Ilmu Komputer

Show 10 entries Search:

No	Nama	Divisi	Instansi	Email	No Telepon	Alumni
11	Agus Ramdhani	Informasi dan Teknologi	PT Kompas Media Nusantara			Lihat Hapus
12	Ubaidillah	Pendidik	Dinas Pendidikan DKI Jakarta			Lihat Hapus
13	Satri Nugraha	Solution Development Government	PT Electronic Data Interchange Indonesia	satri.nugraha@gmail.com	081213893731	Lihat Hapus
14	Wendy	Developer	PT Difini Teknologi	wendy@difinite.com		Lihat Hapus
15	Abdullah Izzudin	Engineering	PT Fliptech Lentera Inspirasi Pertiwi	abdullahizzuddin@gmail.com	+6287878458371	Lihat Hapus

Gambar 3.25: Tampilan Halaman Kelola Pengguna Alumni pada Admin

Kelola Kuesioner Alumni

Alumni / Pengguna

Kategori Kuesioner [+ Buat Kuesioner](#)

No	Kuesioner	Status
1	Pekerjaan	aktif Nonaktifkan Hapus
2	Pendidikan	aktif Nonaktifkan Hapus
3	Kompetensi	aktif Nonaktifkan Hapus
4	Wirausaha	aktif Nonaktifkan Hapus
5	Saran Masukan	aktif Nonaktifkan Hapus

Gambar 3.26: Tampilan Halaman Kelola Kuesioner Alumni pada Admin

Gambar 3.26 merupakan tampilan pada admin yang ditujukan untuk mengelola kuesioner-kuesioner alumni. Pada halaman ini tersedia fitur untuk menambah, menyunting, menghapus kuesioner dan dapat mengaktifkan serta menonaktifkan kuesioner.

Gambar 3.27: Tampilan Halaman Buat Kuesioner Alumni pada Admin

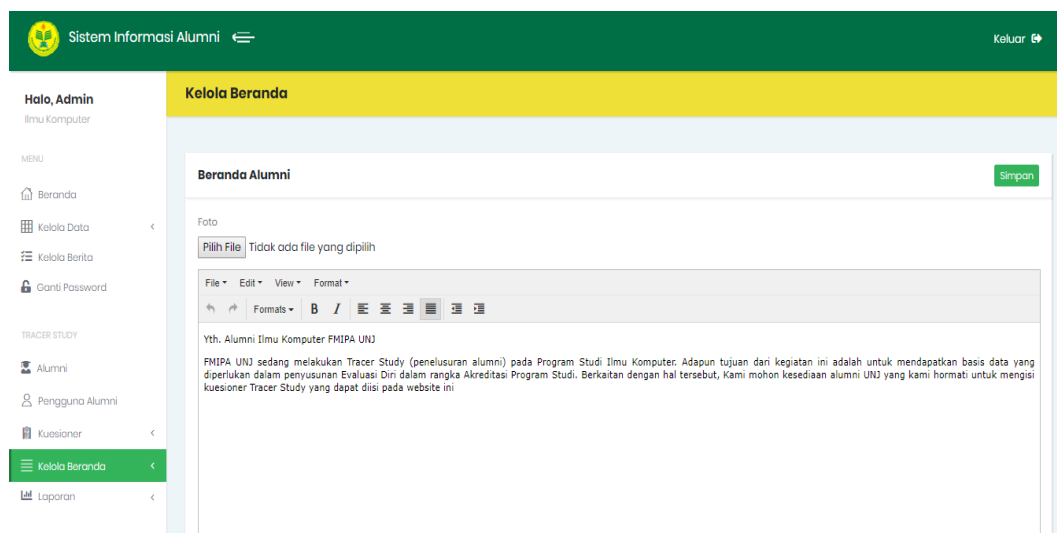
Jika admin memilih tombol buat kuesioner maka sistem akan menampilkan halaman buat kuesioner seperti pada Gambar 3.27. Pada halaman tersebut admin dapat membuat empat jenis pertanyaan, yaitu isian, pertanyaan dengan jawaban pilihan, pertanyaan dengan jawaban lebih dari satu atau ganda, dan pertanyaan dengan jawaban menggunakan skala penilaian.

No	Jenis Kemampuan	Sangat Baik	Baik	Cukup	Kurang
1	Integritas (etika dan moral) lulusan Program Studi Ilmu Komputer UNJ	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
2	Keahlian berdasarkan bidang ilmu (profesionalisme) lulusan Program Studi Ilmu Komputer UNJ	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
3	Kemampuan Bahasa Inggris lulusan Program Studi Ilmu Komputer UNJ	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
4	Kemampuan penggunaan Teknologi Informasi lulusan Program Studi Ilmu Komputer UNJ	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
5	Kemampuan Komunikasi lulusan Program Studi Ilmu Komputer UNJ	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
6	Kemampuan Kerjasama tim lulusan Program Studi Ilmu Komputer UNJ	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
7	Pengembangan diri lulusan Program Studi Ilmu Komputer UNJ	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Gambar 3.28: Tampilan Halaman Buat Kuesioner Pengguna pada Admin

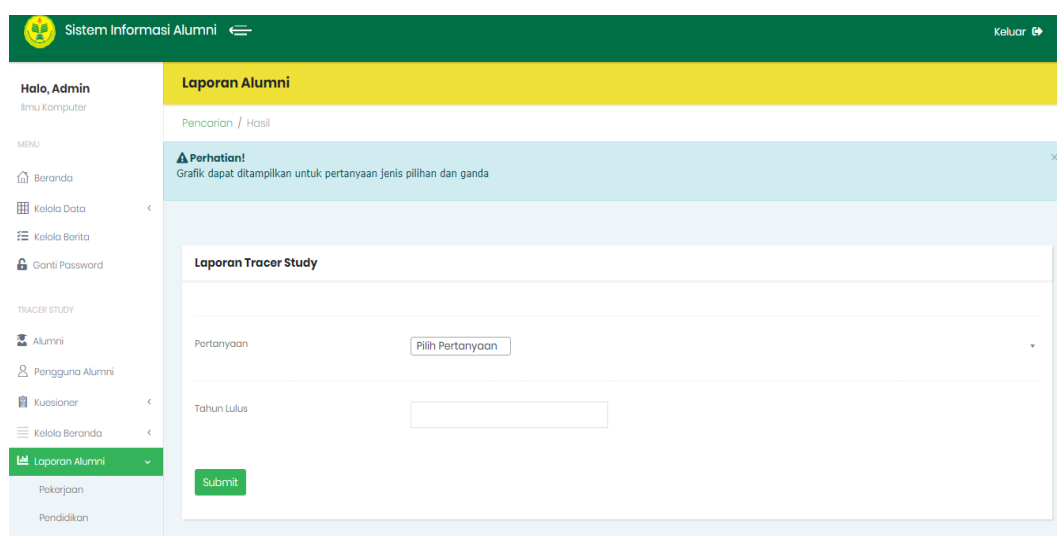
Selain membuat form kuesioner untuk alumni, admin juga dapat membuat

form kuesioner untuk pengguna alumni. Tampilan halaman buat form kuesioner pengguna alumni dapat dilihat pada Gambar 3.28.

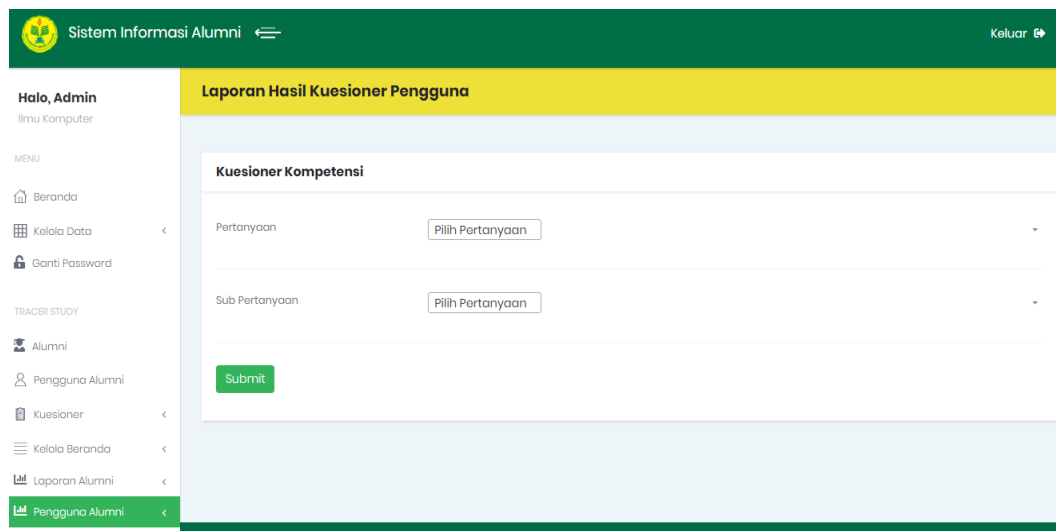


Gambar 3.29: Tampilan Halaman Kelola Beranda Alumni pada Admin

Pada pengembangan sistem ini juga disediakan fitur untuk mengelola kata pengantar pada beranda alumni agar dapat menyesuaikan jika ada perubahan dari pihak prodi di masa mendatang. Fitur tersebut terdapat pada menu kelola beranda yang tampilannya dapat dilihat pada Gambar 3.29.

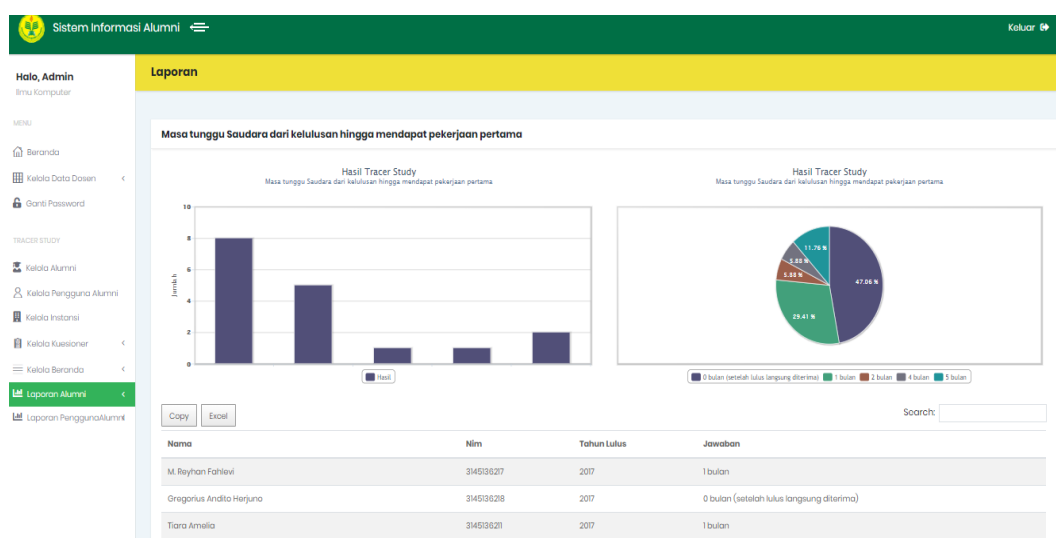


Gambar 3.30: Tampilan Halaman Kriteria Laporan Alumni pada Admin

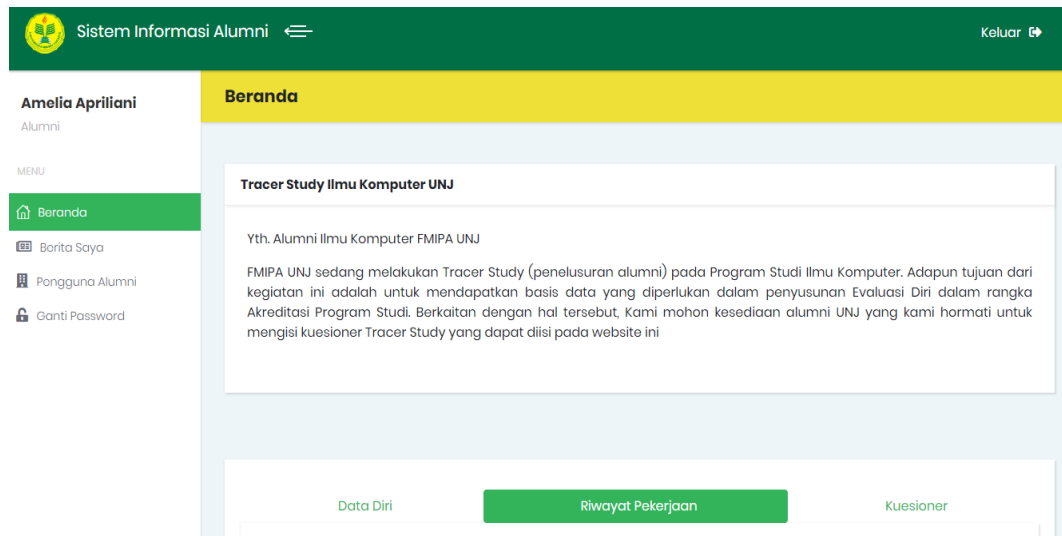


Gambar 3.31: Tampilan Halaman Kriteria Laporan Pengguna Alumni pada Admin

Gambar 3.30 dan Gambar 3.31 merupakan tampilan halaman kriteria laporan atau hasil *tracer study* alumni dan pengguna alumni. Sebelum menampilkan halaman hasil *tracer study*, admin harus memilih terlebih dahulu kriteria laporan yang ingin dilihat, yaitu berupa pertanyaan dan tahun lulus. Setelah admin memilih kriteria maka sistem akan menampilkan halaman hasil *tracer study* berupa grafik dan tabel yang dapat dilihat pada gambar berikut.



Gambar 3.32: Tampilan Halaman Hasil *Tracer Study* pada Admin dan Koorprodi



Gambar 3.33: Tampilan Halaman Beranda pada Alumni

Selanjutnya halaman alumni, setelah alumni melakukan *login* sistem akan menampilkan halaman beranda alumni yang dapat dilihat pada Gambar 3.33. Beranda alumni berisi kata pengantar *tracer study* yang dapat disunting oleh admin serta pengisian data *tracer study* yang terdiri dari data diri, pekerjaan, dan kuesioner.

Data Diri		Riwayat Pekerjaan	Kuesioner
Nama Lengkap	Amelia Apriliani		
Nomor Mahasiswa	3145143626		
Tempat/Tanggal Lahir	Jakarta	23/04/1996	
Jenis Kelamin	Perempuan		
Alamat Rumah			

Gambar 3.34: Tampilan Halaman Data Diri pada Alumni

The screenshot shows the 'Riwayat Pekerjaan' (Work History) section of an alumni form. It features a table with the following data:

No	Instansi	Profesi/Posisi	Periode
1	PT Electronic Data Interchange Indonesia	Technical Writer	2019-sekarang

Below the table, it indicates 'Showing 1 to 1 of 1 entries' and includes navigation buttons for 'Previous', '1', and 'Next'.

Gambar 3.35: Tampilan Halaman Form Riwayat Pekerjaan pada Alumni

Gambar 3.34 adalah tampilan untuk melengkapi data diri alumni. Sedangkan gambar 3.35 merupakan tampilan form untuk memasukkan data riwayat pekerjaan alumni

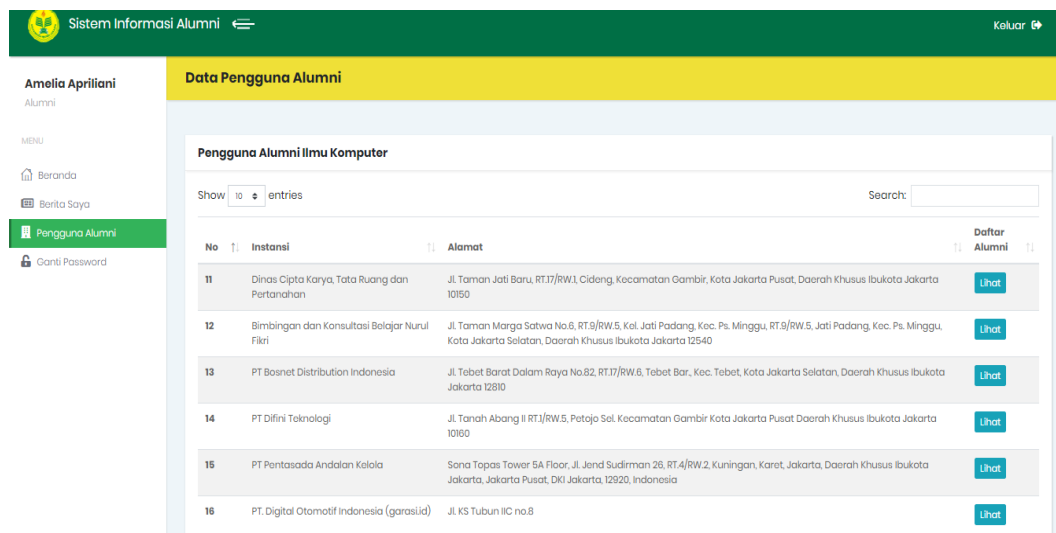
The screenshot shows the 'Kuesioner' (Survey) section of an alumni form. It contains the following questions and options:

- Masa tunggu Saudara dari kelulusan hingga mendapat pekerjaan pertama kali : [Dropdown menu]
- Apakah pekerjaan Saudara ini berhubungan dengan bidang ilmu yang Saudara pelajari di Perguruan Tinggi? Jelaskan jawaban anda :
☐ Ya
☐ Tidak
[Text area for explanation]
- Bagaimana anda mencari pekerjaan? :
☐ Melalui iklan di koran/majalah, brosur
☐ Melamar ke perusahaan tanpa mengetahui lowongan yang ada
☐ Pergi ke bursa/pameran kerja
☐ Mencari lewat internet/iklan online/milis
☐ Dihubungi oleh perusahaan
☐ Memeroleh informasi dari pusat pengembangan karir fakultas/universitas
☐ Menghubungi kantor kemahasiswaan/hubungan alumni
☐ Melalui jaringan (relasi) orang-orang terdekat

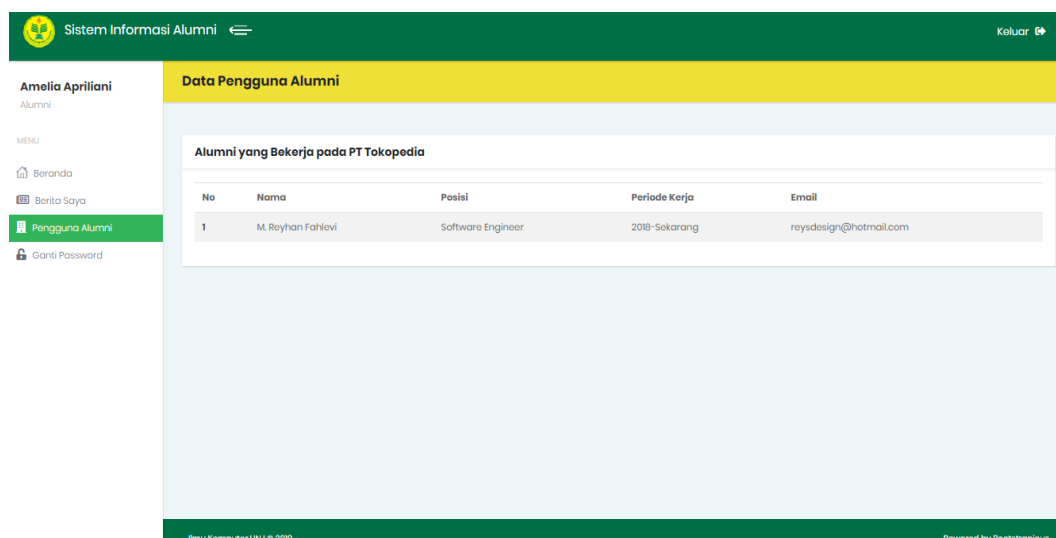
Gambar 3.36: Tampilan Halaman Kuesioner pada Alumni

Gambar 3.36 adalah tampilan halaman mengisi kuesioner untuk alumni. Semua jenis kuesioner ditampilkan dalam satu halaman. Halaman kuesioner ini menyediakan pertanyaan isian, pilihan, ganda atau dapat memilih lebih dari satu jawaban,

dan pertanyaan dengan skala penilaian.



Gambar 3.37: Tampilan Halaman Daftar Pengguna pada Alumni



Gambar 3.38: Tampilan Halaman Daftar Pengguna pada Alumni

Pada pengembangan sistem ini menyediakan fasilitas untuk berbagi informasi antar alumni dapat dilihat pada Gambar 3.37 dan Gambar 3.38. Gambar 3.37 merupakan halaman daftar pengguna alumni pada alumni. Pada halaman ini alumni dapat melihat daftar pengguna alumni dari prodi Ilmu Komputer. Jika alumni memilih tom-

bol lihat pada kolom daftar alumni, sistem akan menampilkan halaman daftar alumni yang bekerja pada pengguna alumni terkait seperti pada Gambar 3.38.

No	Nama	NIM	Angkatan	Tahun Lulus	Pekerjaan
1	Agustinus Purimbaga	3145136223	2013	2017	Lihat
2	Alitina Prastiantari	3145136208	2013	2017	Lihat
3	Anantassa Fitri Andini	3145136212	2013	2017	Lihat
4	Gregorius Andito H	3145136218	2013	2017	Lihat
5	Hana Maulinda	3145136193	2013	2017	Lihat
6	M. Reyhan Fahlevi	3145136217	2013	2017	Lihat

Gambar 3.39: Tampilan Halaman Daftar Alumni pada Koorprodi

Selanjutnya halaman untuk koorprodi. Halaman koorprodi dapat dilihat pada Gambar 3.39. Halaman koorprodi menyediakan menu untuk melihat data alumni beserta pekerjaannya, menu untuk melihat daftar pengguna alumni, menu untuk melihat daftar instansi beserta skala instansi dan menu untuk halaman laporan hasil *tracer study*.

Sistem Informasi Alumni

Welcome,
Pengguna Alumni

Form Kuesioner Pengguna

Bapak/Ibu yang terhormat, saat ini kami sedang melakukan *Tracer Study* (penelusuran alumni) **Program Studi Ilmu Komputer** FMIPA-UNJ. Adapun tujuan dari kegiatan ini adalah untuk mendapatkan basis data yang diperlukan dalam penyusunan Evaluasi Diri dalam rangka Akreditasi Program Studi. Berkaitan dengan hal tersebut, kami mohon Bapak/Ibu dapat mengisi kuesioner ini, data yang Bapak/Ibu isi dijamin kerahasiaannya. Untuk kerjasama dan bantuannya, kami mengucapkan banyak terima kasih.

No	Jenis Kemampuan	Sangat Baik	Baik	Cukup	Kurang
1	Integritas (etika dan moral) lulusan Program Studi Ilmu Komputer UNJ	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
2	Keahlian berdasarkan bidang ilmu (profesionalisme) lulusan Program Studi Ilmu Komputer UNJ	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
3	Kemampuan Bahasa Inggris lulusan Program Studi Ilmu Komputer UNJ	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
4	Kemampuan penggunaan Teknologi Informasi lulusan Program Studi Ilmu Komputer UNJ	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
5	Kemampuan Komunikasi lulusan Program Studi Ilmu Komputer UNJ	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
6	Kemampuan Kerjasama tim lulusan Program Studi Ilmu Komputer UNJ	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
7	Pengembangan diri lulusan Program Studi Ilmu Komputer UNJ	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Gambar 3.40: Tampilan Halaman Kuesioner untuk Pengguna Alumni

Gambar 3.40 merupakan halaman kuesioner yang diperuntukkan untuk pengguna alumni. Halaman ini akan muncul setelah pengguna alumni mengklik *link* kuesioner yang dikirimkan admin melalui *email*.

Sistem Informasi Alumni
Universitas Negeri Jakarta

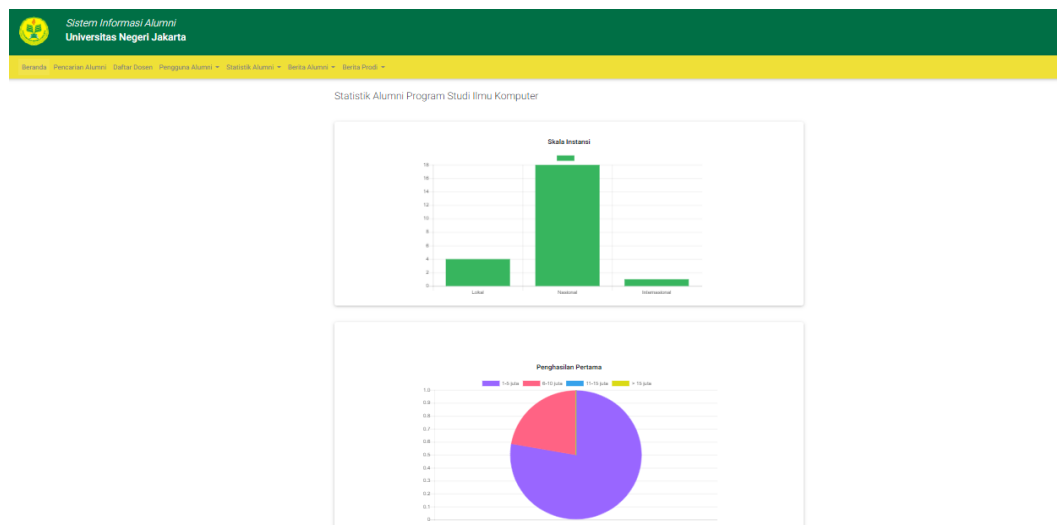
Beranda Pencarian Alumni Daftar Dosen Pengguna Alumni Statistik Alumni Berita Alumni Berita Prodi **Masuk**

Daftar Pengguna Alumni Ilmu Komputer

Show entries Search:

No	Instansi	Posisi Alumni
1	PT Harmoni Solusi Bisnis	• Front End Developer • Front End Developer
2	PT Kompas Media Nusantara	• Front End Developer
3	PT Sinar Mas	• Java Pega Programmer
4	PT Docotel Teknologi	• PHP Engineer
5	PT Tokopedia	• Software Engineer
6	PT Manulife Indonesia	• IT Bisnis

Gambar 3.41: Tampilan Halaman Daftar Pengguna Alumni untuk Pengunjung



Gambar 3.42: Tampilan Halaman Statistik Alumni untuk Pengunjung

Selanjutnya adalah tampilan halaman yang dapat dilihat oleh pengunjung *web-site* tanpa *login*, yaitu terdapat halaman daftar pengguna alumni yang dapat dilihat pada Gambar 3.41 dan halaman statistik alumni yang dapat dilihat pada Gambar 3.42. Halaman daftar pengguna alumni berisi instansi-instansi tempat alumni bekerja beserta posisi pekerjaan alumni pada instansi terkait. Sedangkan halaman statistik alumni berisi beberapa hasil *tracer study* yang dapat dilihat pengunjung.

3. Implementasi Sistem (*Back End*)

Dalam pengimplementasian semua fungsi yang berada dalam sistem digunakan bahasa pemrograman PHP kemudian menggunakan *framework codeIgniter* untuk memudahkan dalam mengimplementasikan arsitektur MVC. Dengan arsitektur MVC proses pengkodean dibagi menjadi tiga bagian, yaitu Model untuk menyimpan fungsi yang berhubungan dengan *database*, kemudian View untuk menampilkan informasi yang direpresentasikan kepada pengguna, dan Controller yang menjembatani antara Model dan View. Berikut merupakan sampel pemrograman terdiri dari *Model*, *View*, dan *Controller*.

```

<?php
defined('BASEPATH') OR exit('No direct script access allowed');

class Alumni extends CI_Controller {
    function __construct(){
        parent::__construct();
        if($SESSION["logged_in"] != 'login') {
            redirect(base_url("login"));
        }

        $this->load->model('m_alumni');
        $this->load->model('m_master');
        $this->load->model('m_pengguna');
        $this->load->library(array('PHPExcel','PHPExcel/IOFactory'));
    }

    public function index()
    {
        $prodIID = $this->session->userdata('prodIID');
        $data = array(
            'role' => $this->session->userdata('role'),
            'userID' => $this->session->userdata('userID'),
            'prodIID' => $prodIID,
            'alumni' => $this->m_alumni->getAlumni($prodIID),
        );
        $this->load->view('element/head');
        $this->load->view('element/header');
        $this->load->view('element/navbar', $data);
        $this->load->view('admin/v_dataAlumni', $data);
        $this->load->view('element/footer');
    }

    function exeAddAlumni()
    {
        $userID = 'ALU' . $this->input->post('nim');
        $cek = $this->m_alumni->cekAlumni($this->input->post('nim'));

        if($cek->num_rows() > 0){
            $this->session->set_flashdata("gagalAddAlumni", '<div><div class="alert alert-danger" id="alert" align="center">Gagal menambahkan! Akun alumni
            sudah terdaftar</div></div>');
        } else {
            $data_user = array(

```

Gambar 3.43: Struktur Pemrograman *Controller*

```

<?php
class M_alumni extends CI_Model{
    public function __construct()
    {
        parent::__construct();
        // $this->db2 = $this->load->database('db_dua', TRUE);
    }

    function getAlumniByUserID($userid)
    {
        $this->db->select('*');
        $this->db->where('userID', $userid);
        $query = $this->db->get('alumni');
        return $query->row();
    }

    function getAlumniByID($id)
    {
        $this->db->select('*');
        $this->db->where('id', $id);
        $query = $this->db->get('alumni');
        return $query->row();
    }

    function getAlumni($prodIID)
    {
        $this->db->select('*');
        $this->db->where('status', 'aktif');
        $this->db->where('prodIID', $prodIID);
        $this->db->order_by('id', 'DESC');
        $query = $this->db->get('alumni');
        if($query->num_rows() > 0)
        {
            return $query->result();
        } else{
            return $query->result();
        }
    }
}

```

Gambar 3.44: Struktur Pemrograman *Model*

```

<div class="content-inner">
<!-- Page Header -->
<header class="page-header" style="background-color: #EFE037">
<div class="container-fluid">
<h2 class="no-margin-bottom">Data Alumni</h2>
</div>
</header>

<?php echo $this->session->flashdata('suksesImpon'); ?>
<?php echo $this->session->flashdata('gagalImpon'); ?>
<?php echo $this->session->flashdata('suksesAddAlumni'); ?>
<?php echo $this->session->flashdata('gagalAddAlumni'); ?>

<section class="tables">
<div class="container-fluid">
<div class="row">
<div class="col-lg-12">
<div class="card">
<div class="card-header d-flex align-items-center">
<h3 class="h4">Alumni <?php echo $this->m_master->getProdiByID($prodiID)->nama_prodi ?></h3>
<button type="button" class="btn btn-primary ml-auto btn-sm" data-toggle="modal" data-target="#modalTambah"><i class="fas fa-user-plus"></i>
> Tambah Akun</button>
<button type="button" class="btn btn-dark ml-3 btn-sm" data-toggle="modal" data-target="#modalImport"><i class="fas fa-cloud-upload-alt"></i>
> Import</button>
</div>
<div class="card-body">
<div class="table-responsive">
<table id="myTable" class="table table-striped table-hover">
<thead>
<tr>
<th>No</th>
<th>Nama</th>
<th>MIPA</th>
<th>Angkatan</th>
<th>Tahun Lulus</th>
</tr>
</thead>
<tbody>
<?php
$no = 1;
foreach($alumni as $d){

```

Gambar 3.45: Struktur Pemrograman View

BAB IV

UJI COBA DAN HASIL UJI COBA

A. Uji Coba

Pada bagian ini akan dijelaskan mengenai pengujian Sistem Informasi *Tracer Study* Program Studi Ilmu Komputer yang telah dikembangkan. Sesuai dengan tahapan SDLC *spiral model* setelah tahapan konstruksi atau pengembangan akan dilanjutkan ke tahapan evaluasi atau pengujian. Tahapan ini bertujuan untuk memastikan sistem yang telah dibuat dapat bekerja sesuai dengan kebutuhan, memperoleh umpan balik dari pengguna, dan menemukan kesalahan atau *bugs* pada sistem.

Uji coba pada sistem dilakukan terhadap satu responden ahli untuk uji keseluruhan sistem, satu responden admin, satu responden koorprodi, 15 responden alumni, dan 5 responden pengguna alumni. Setiap responden akan melakukan pengujian terhadap sistem berdasarkan peran masing-masing responden. Uji coba yang dilakukan menggunakan kuesioner Pengujian Penerimaan Pengguna atau yang disebut dengan UAT (*User Acceptance Test*). Langkah-langkah pengujian Sistem Informasi *Tracer Study* Prodi Ilmu Komputer adalah sebagai berikut.

1. Admin mengelola (menambah, menyunting, menghapus) data alumni.
2. Admin menyunting kata pengantar untuk beranda alumni
3. Admin membuat kuesioner untuk alumni dan pengguna alumni
4. Alumni menyunting profil atau data diri
5. Alumni mengelola (menambah, menyunting, menghapus) riwayat pekerjaan.
6. Alumni mengisi kuesioner yang dibuat admin

7. Admin mengelola (menambah, menyunting, menghapus) data pengguna alumni.
8. Pengguna alumni mengisi kuesioner yang dibuat admin
9. Admin melihat laporan hasil *tracer study*
10. Koorprodi melihat data alumni dan pengguna alumni
11. Koorprodi melihat laporan hasil *tracer study*

Pengujian pada sistem ini terdiri dari dua, yaitu pengujian fungsional dan kebergunaan atau *usability*. Pengujian fungsional berfokus pada fungsionalitas perangkat lunak yang dikembangkan. Sedangkan pengujian kebergunaan dilakukan untuk mengetahui tanggapan pengguna terhadap sistem seberapa efisien dan efektif sistem yang dikembangkan dalam membantu pengguna memenuhi kebutuhannya. Berikut adalah komponen-komponen yang akan diuji pada Sistem Informasi *Tracer Study* Prodi Ilmu Komputer:

1. Admin

- Mengelola (menambah, menyunting, menghapus) data alumni
- Mengelola (menambah, menyunting, menghapus) data pengguna alumni
- Menyunting kata pengantar *tracer study*
- Membuat kuesioner alumni dan pengguna alumni
- Melihat laporan hasil *tracer study*

2. Kooprodi

- Melihat data alumni
- Melihat data pengguna alumni

- Melihat laporan hasil *tracer study*.

3. Alumni

- Menyunting data diri dan mengelola riwayat pekerjaan
- Mengisi form kuesioner *tracer study*
- Melihat daftar pengguna alumni beserta alumni yang bekerja

4. Pengguna Alumni

- Mengisi form kuesioner pengguna alumni

Dari semua komponen penilaian tersebut untuk pengujian fungsional dilakukan dengan dua pilihan yaitu :

1. S : Setuju
2. TS : Tidak Setuju

Sedangkan untuk pengujian kebergunaan (*usability*) dilakukan menggunakan skala *likert*. Skala *likert* merupakan suatu skala penilaian yang menyajikan pilihan skala dengan nilai pada setiap skala untuk mengukur tingkat persetujuan terhadap sesuatu [18]. Skala yang digunakan dengan nilai 1 sampai 5 dengan perincian sebagai berikut :

- 1 : Sangat Tidak Setuju
- 2 : Tidak Setuju
- 3 : Cukup
- 4 : Setuju
- 5 : Sangat Setuju

Berdasarkan nilai yang didapatkan dari pengguna saat pengujian, angka tersebut akan dikalkulasikan sesuai dengan sistem penilaian berikut :

- Nilai total

Nilai total merupakan jumlah keseluruhan nilai/skor yang didapat pada setiap pertanyaan atau dapat ditulis menjadi:

$$Nilai\ Total = (jumlah \times skorSS) + (jumlah \times skorS) + (jumlah \times skorC) + (jumlah \times skorTS) + (jumlah \times skorSTS)$$

- Persentase Kelayakan

Persentase kelayakan atau persentase nilai rata-rata yang didapatkan dari nilai total dibagi skor yang diharapkan. Skor yang diharapkan didapat dari skor maksimal yang dikalikan dengan jumlah responden. Skor maksimal adalah nilai maksimal dari skala *likert* dikalikan dengan jumlah pertanyaan. Perhitungan tersebut dapat ditulis menjadi :

$$Persentase\ Kelayakan(\%) = \frac{NilaiTotal}{SkorDiharapkan} \times 100\%$$

Persentase kelayakan yang didapatkan akan dibandingkan dengan skor pada skala *likert*. Berikut model skala *likert* lima skala :

1. Sangat Kurang Sesuai = 0%-20%
2. Kurang Sesuai = 21%-40%
3. Cukup Sesuai = 41%-60%
4. Sesuai = 61%-80%
5. Sangat Sesuai = 81%-100%

B. Hasil Percobaan

Berdasarkan percobaan yang dilakukan terhadap 1 admin, 1 koorprodi, 15 alumni, dan 5 pengguna alumni, didapatkan hasil percobaan sebagai berikut :

1. Pengujian oleh Admin

Pengujian untuk *user* admin dilakukan oleh seorang responden yang bernama Bapak Alfisyahrizqa selaku staf admin prodi Ilmu Komputer pada tanggal 8 Januari 2020. Penilaian yang diberikan seputar fungsionalitas dan kebergunaan dari halaman *user* admin pada sistem *tracer study* Ilmu Komputer

1. Uji Coba Fungsional

Tabel 4.1 adalah hasil uji fungsional pada admin *tracer study* :

Tabel 4.1: Hasil Uji Fungsional pada *user* Admin

No	Keterangan	S	TS
1	Fitur masuk ke dalam sistem (<i>login</i>) berjalan dengan baik	✓	
2	Fitur mengelola (menambah, menyunting, dan menghapus) data alumni berjalan dengan baik dan sesuai kebutuhan	✓	
3	Fitur mengelola (menyunting, dan menghapus) data pengguna alumni berjalan dengan baik dan sesuai kebutuhan	✓	
4	Fitur menyunting kata pengantar <i>tracer study</i> berjalan dengan baik sesuai kebutuhan	✓	
5	Fitur mengelola (membuat, mengedit, dan menghapus) kuesioner alumni berjalan dengan baik dan sesuai kebutuhan	✓	
5	Fitur mengelola (membuat, mengedit, dan menghapus) kuesioner pengguna alumni berjalan dengan baik sesuai kebutuhan	✓	
7	Fitur melihat laporan hasil <i>tracer study</i> berjalan dengan baik sesuai kebutuhan	✓	
8	Fitur <i>export</i> data <i>tracer</i> berjalan dengan baik dan sesuai kebutuhan	✓	
9	Fitur mengganti <i>password</i> berjalan dengan baik	✓	
10	Fitur keluar sistem (<i>logout</i>) berjalan dengan baik	✓	

Berdasarkan hasil uji coba fungsional yang dilakukan, keseluruhan fitur *user*

admin pada sistem informasi *tracer study* prodi Ilmu Komputer dapat berjalan dengan baik dan sesuai dengan yang diharapkan.

2. Uji Coba Kebergunaan

Berikut merupakan daftar pertanyaan uji *usability* pada Admin:

Tabel 4.2: Daftar Pertanyaan Uji *Usability* pada Admin

No	Keterangan	STS	TS	C	S	SS
1	Fitur mengelola data alumni mudah dimengerti					
2	Fitur mengelola data pengguna alumni mudah dimengerti					
3	Fitur menyunting kata pengantar <i>tracer study</i> mudah dimengerti					
4	Fitur membuat kuesioner alumni dan pengguna alumni mudah dimengerti					
5	Fitur melihat laporan hasil <i>tracer study</i> mudah dimengerti					
6	Fitur <i>export</i> data <i>tracer</i> mudah dimengerti					
7	Sistem ini membantu dalam pengelolaan data alumni					
8	Sistem ini mudah untuk digunakan					
9	Sistem ini nyaman untuk digunakan					

Berikut merupakan hasil uji kebergunaan pada admin *tracer study* yang dapat dilihat pada tabel berikut:

Tabel 4.3: Hasil uji *usability* pada Admin

Pertanyaan	Jawaban Responden				
	STS	TS	C	S	SS
1	0	0	0	1	0
2	0	0	0	1	0
3	0	0	0	1	0
4	0	0	0	1	0
5	0	0	0	1	0
6	0	0	0	1	0
7	0	0	0	1	0
8	0	0	0	1	0
9	0	0	0	1	0
Jumlah	0	0	0	9	0

$$\text{Nilai Total} = (0 \times 1) + (0 \times 1) + (0 \times 3) + (9 \times 4) + (0 \times 5) = 36$$

$$\text{Persentase kelayakan}(\%) = \frac{36}{45} \times 100\% = 80\%$$

Berdasarkan hasil uji coba kebergunaan yang telah dilakukan, didapatkan persentase kelayakan senilai **80%**. Maka dapat dikatakan halaman pada *user* admin mendapatkan predikat sesuai untuk aspek kebergunaan sistem. Dengan catatan data pada sistem di-*update* dengan data terbaru.

2. Pengujian oleh Koorprodi

Pengujian untuk *user* koorprodi dilakukan oleh seorang responden yang bernama Ibu Ir. Fariani Hermin I, M.T. selaku koorprodi Ilmu Komputer pada tanggal 8 Januari 2020. Penilaian yang diberikan terkait fungsionalitas dan kebergunaan dari halaman *user* koorprodi pada sistem tracer study Ilmu Komputer.

1. Uji Coba Fungsional

Tabel 4.4 adalah hasil uji fungsional pada koorprodi :

Tabel 4.4: Hasil Uji Fungsional pada Koorprodi

No	Keterangan	S	TS
1	Fitur masuk ke dalam sistem (<i>login</i>) berjalan dengan baik	✓	
2	Fitur melihat data alumni berjalan dengan baik dan sesuai kebutuhan	✓	
3	Fitur melihat data pengguna alumni berjalan dengan baik dan sesuai kebutuhan	✓	
4	Fitur melihat laporan hasil <i>tracer study</i> berjalan dengan baik dan sesuai kebutuhan	✓	
5	Fitur <i>export</i> data <i>tracer</i> berjalan dengan baik dan sesuai kebutuhan	✓	
6	Fitur mengganti <i>password</i> berjalan dengan baik	✓	
7	Fitur keluar sistem (<i>logout</i>) berjalan dengan baik	✓	

Berdasarkan hasil uji coba fungsional yang dilakukan, sistem informasi *tracer study* prodi Ilmu Komputer dapat berjalan dengan baik dan sesuai dengan yang dibutuhkan. Koorprodi Ilmu Komputer dapat melihat data alumni dan pengguna alumni, dapat melihat laporan hasil *tracer study*, dan *export* data *tracer*.

2. Uji Coba Kebergunaan

Berikut merupakan daftar pertanyaan uji *usability* pada Koorprodi:

Tabel 4.5: Daftar Pertanyaan Uji *Usability* pada Koorprodi

No	Keterangan	STS	TS	C	S	SS
1	Fitur melihat data alumni mudah dimengerti					
2	Fitur melihat data pengguna alumni mudah dimengerti					
3	Fitur melihat hasil laporan <i>tracer study</i> mudah dimengerti					
4	Sistem ini membantu prodi dalam memenuhi kebutuhan data alumni					
5	Sistem ini mudah untuk digunakan					
6	Sistem ini nyaman untuk digunakan					

Berikut merupakan hasil uji *usability* pada koorprodi yang dapat dilihat pada tabel berikut:

Tabel 4.6: Hasil Uji *Usability* pada Koorprodi

Pertanyaan	Jawaban Responden				
	STS	TS	C	S	SS
1	0	0	0	0	1
2	0	0	0	0	1
3	0	0	0	0	1
4	0	0	0	0	1
5	0	0	0	0	1
6	0	0	0	0	1
Jumlah	0	0	0	0	6

$$\text{Nilai Total} = (0 \times 1) + (0 \times 2) + (0 \times 3) + (0 \times 4) + (6 \times 5) = 30$$

$$\text{Persentase kelayakan}(\%) = \frac{30}{30} \times 100\% = 100\%$$

Berdasarkan hasil uji coba kebergunaan yang telah dilakukan pada *user* koorprodi, didapatkan persentase kelayakan senilai **100%**. Maka dapat dikatakan halaman pada *user* koorprodi mendapatkan predikat sangat sesuai untuk aspek kebergunaan sistem.

3. Pengujian oleh Alumni

User Acceptance Test pada *user* Alumni dilakukan oleh 15 responden alumni prodi Ilmu Komputer UNJ terdiri dari angkatan 2013 sampai 2015. Penilaian yang diberikan mengenai fungsionalitas dan kebergunaan dari halaman *user* alumni pada sistem *tracer study* Ilmu Komputer.

1. Uji Coba Fungsional

Tabel 4.7 adalah hasil uji fungsional pada alumni :

Tabel 4.7: Hasil Uji Fungsional pada Alumni

No	Pertanyaan	Jawaban Responden	
		Setuju	Tidak Setuju
1	Fitur masuk kedalam sistem (<i>login</i>) berjalan dengan baik	15	0
2	Fitur menyunting data diri berjalan dengan baik	15	0
3	Fitur mengelola (menambah, mengedit, dan menghapus) riwayat pekerjaan berjalan dengan baik	14	1
4	Fitur mengisi form kuesioner <i>tracer study</i> berjalan dengan baik	14	1
5	Fitur melihat daftar pengguna alumni beserta alumni yang bekerja berjalan dengan baik	14	1
6	Fitur mengganti password berjalan dengan baik	15	0
7	Fitur keluar sistem (<i>logout</i>) berjalan dengan baik	15	0
Jumlah		102	3
Persentase Jawaban		97,14%	2,86%

Berdasarkan hasil uji coba fungsional yang dilakukan pada 15 responden alumni, didapatkan jumlah total jawaban 102 dari maksimal nilai 105, maka persen-

tase jawaban adalah **97.14%** sehingga dapat dikatakan fitur-fitur pada halaman alumni dapat berjalan dengan baik dan sesuai dengan yang diharapkan.

2. Uji Coba Kebergunaan

Berikut merupakan daftar pertanyaan uji *usability* pada Alumni:

Tabel 4.8: Daftar Pertanyaan Uji *Usability* pada Alumni

No	Keterangan	STS	TS	C	S	SS
1	Fitur menyunting data diri mudah dimengerti					
2	Fitur mengelola (menambah, menyunting, dan menghapus) riwayat pekerjaan mudah dimengerti					
3	Fitur mengisi form kuesioner mudah dimengerti					
4	Fitur melihat daftar pengguna alumni beserta alumni yang bekerja mudah dimengerti					
5	Sistem ini membantu alumni dalam mendapat informasi terkait jenjang pekerjaan bidang Ilmu Komputer					
6	Sistem ini mudah untuk digunakan					
7	Sistem ini nyaman untuk digunakan					

Berikut merupakan hasil uji *usability* pada alumni yang dapat dilihat pada tabel berikut:

Pertanyaan	Jawaban Responden				
	STS	TS	C	S	SS
1	0	1	1	8	5
2	0	2	2	7	4
3	0	0	5	8	2
4	0	2	1	9	3
5	0	0	2	3	10
6	0	0	2	8	5
7	0	0	3	8	4
Jumlah	0	5	16	51	33
Jumlah Skor	0	10	48	204	165

$$\text{Nilai Total} = (0 \times 1) + (5 \times 2) + (16 \times 3) + (51 \times 4) + (33 \times 5) = 427$$

$$\text{Persentase kelayakan}(\%) = \frac{427}{525} \times 100\% = 81.33\%$$

Berdasarkan hasil uji coba kebergunaan yang telah dilakukan pada 15 responden alumni, didapatkan persentase kelayakan senilai **81.33%**. Maka dapat dikatakan halaman pada *user* alumni mendapatkan predikat sangat sesuai untuk aspek kebergunaan sistem.

4. Pengujian oleh Pengguna Alumni

User Acceptance Test pada *user* Pengguna Alumni dilakukan oleh 5 responden yang merupakan atasan dari alumni di instansi tempat alumni bekerja. Penilaian yang diberikan mengenai fungsionalitas dan kebergunaan dari halaman *user* pengguna alumni pada sistem *tracer study* Ilmu Komputer.

1. Uji Coba Fungsional

Tabel 4.9 adalah hasil uji fungsional pada pengguna alumni :

No	Pertanyaan	Jawaban Responden	
		Setuju	Tidak Setuju
1	Fitur mengisi form kuesioner pengguna alumni berjalan dengan baik	5	0
Total		5	0
Persentase Jawaban		100%	0%

Tabel 4.9: Hasil Uji Fungsional pada Pengguna Alumni

Berdasarkan hasil uji coba fungsional yang dilakukan pada *user* pengguna alumni, sistem informasi *tracer study* prodi Ilmu Komputer dapat berjalan dengan baik dan sesuai dengan yang diharapkan. Pengguna alumni dapat mengisi kuesioner secara *online*.

2. Uji Coba Kebergunaan

Tabel 4.10 merupakan daftar pertanyaan dari uji kebergunaan/*usability* pada pengguna alumni:

No	Keterangan	STS	TS	C	S	SS
1	Fitur mengisi kuesioner pengguna alumni mudah dimengerti					
2	Sistem ini mudah digunakan					
3	Sistem ini nyaman digunakan					

Tabel 4.10: Daftar Pertanyaan Uji *Usability* pada Pengguna Alumni

Berikut merupakan hasil uji *usability* pada pengguna alumni yang dapat dilihat pada tabel berikut:

Pertanyaan	Jawaban Responden				
	STS	TS	C	S	SS
1	0	0	1	3	1
2	0	0	1	2	2
3	0	0	1	3	1
Jumlah	0	0	3	8	4
Jumlah Skor	0	0	9	32	20

Tabel 4.11: Hasil Uji *Usability* pada Pengguna Alumni

$$\text{Nilai Total} = (0 \times 1) + (0 \times 2) + (3 \times 3) + (8 \times 4) + (4 \times 5) = 61$$

$$\text{Persentase kelayakan}(\%) = \frac{61}{75} \times 100\% = 81.33\%$$

Berdasarkan hasil uji coba kebergunaan yang telah dilakukan pada 5 responden pengguna alumni, didapatkan persentase kelayakan senilai **81.33%**. Maka dapat dikatakan halaman pada *user* pengguna alumni mendapatkan predikat sangat sesuai untuk aspek kebergunaan sistem.

5. Hasil Pengujian Keseluruhan Sistem

Berdasarkan hasil pengujian fungsional dan kebergunaan (*usability*) Sistem Informasi *Tracer Study* Program Studi Ilmu Komputer pada semua *user*, didapatkan bahwa fitur-fitur yang terdapat pada sistem dapat berjalan dengan baik. Selain itu, didapatkan persentase kelayakan sebagai berikut:

- Halaman *user* Admin : 80%
- Halaman *user* Koorprodi : 100%
- Halaman *user* Alumni : 81.33%
- Halaman *user* Pengguna Alumni : 81.33%

Dari persentase masing-masing *user* kemudian dihitung total persentase kelayakan yang merupakan rata-rata dari nilai persentase kelayakan semua user, dapat dilihat sebagai berikut:

$$TotalPersentaseKelayakan(\%) = \frac{80\%+100\%+81.33\%+81.33\%}{4} = 85.66\%$$

Berdasarkan perhitungan tersebut didapatkan total persentase kelayakan senilai **85.66%** berada pada rentang tafsiran 81%-100%, maka dapat dikatakan bahwa nilai kebergunaan pada keseluruhan sistem mendapatkan predikat sangat sesuai.

BAB V

KESIMPULAN DAN SARAN

A. Kesimpulan

Berdasarkan hasil implementasi dan pengujian program ini, maka didapat kesimpulan sebagai berikut:

1. Pengembangan Sistem Informasi *Tracer Study* Program Studi Ilmu Komputer berdasarkan dari analisis dan evaluasi perbandingan fitur sistem *tracer study* yang telah tersedia, yaitu Sistem Informasi *Tracer Study* Universitas Negeri Jakarta dan Sistem *Tracer Study* Dikti. Selain itu, melalui pengumpulan data yang diperoleh dari analisis kebutuhan prodi Ilmu Komputer.
2. Proses pengembangan Sistem Informasi *Tracer Study* Program Studi Ilmu Komputer menggunakan metode pengembangan perangkat lunak *System Development Life Cycle* dengan *Spiral Model* yang memiliki beberapa tahapan, yaitu pengumpulan dan analisis kebutuhan, perancangan atau desain, konstruksi atau pengkodean, dan pengujian atau evaluasi.
3. Sistem Informasi *Tracer Study* Program Studi Ilmu Komputer dikembangkan dengan bahasa pemrograman PHP dengan bantuan *framework Codeigniter*.
4. Berdasarkan hasil *User Acceptance Test* pada pengujian fungsional didapatkan bahwa fitur-fitur yang terdapat pada Sistem Informasi *Tracer Study* Program Studi Ilmu Komputer dapat berjalan dengan baik dan sesuai kebutuhan.
5. Berdasarkan hasil *User Acceptance Test* pada pengujian kebergunaan (*usability*), didapatkan total persentase kelayakan dari keseluruhan sistem adalah

85.66%. Nilai tersebut terdapat pada skor skala *likert* 81%-100%, maka nilai kebergunaan Sistem Informasi *Tracer Study* Program Studi Ilmu Komputer mendapat predikat sangat sesuai.

6. Sistem Informasi *Tracer Study* Program Studi Ilmu Komputer dikembangkan untuk dapat dimanfaatkan pihak prodi terkait pengelolaan data alumni, menghasilkan informasi hasil *tracer study* yang dibutuhkan prodi untuk meningkatkan kualitas pendidikannya dan memenuhi kebutuhan akreditasi. Selain itu, ditujukan untuk masyarakat agar dapat lebih mengenal lulusan prodi Ilmu Komputer

B. Saran

Adapun beberapa saran untuk penelitian selanjutnya adalah:

1. Ruang lingkup penelitian selanjutnya diharapkan dapat ditingkatkan menjadi rumpun maupun fakultas.
2. Desain dan fungsionalitas sistem diharapkan lebih memperhatikan pada peramban ponsel (*mobile friendly*)
3. Pada sistem diharapkan untuk menambahkan halaman yang menampilkan status pengisian kuesioner apakah pernah mengisi atau belum

DAFTAR PUSTAKA

- [1] Alim, R. S.(2019). *Perancangan dan Implementasi Sistem Informasi Penelusuran Alumni (Tracer Study)* [Skripsi]. Jakarta: Universitas Negeri Jakarta.
- [2] Anggraeni, E. Y., Irviani, R. (2017). *Pengantar Sistem Informasi*. Yogyakarta: ANDI.
- [3] S., Rosa A., Shalahuddin, M. (2014). *Rekayasa Perangkat Lunak Struktur dan Berorientasi Objek*. Bandung: Informatika.
- [4] Azis, M. F. (2005). *Object Oriented Programming dengan PHP 5*. Jakarta: Elex Media Komputindo
- [5] Boell, S. K., Cecez-Kecmanovic, D. (2015). What is an Information System?. *Hawaii International Conference on System Sciences*.
- [6] Edi, D., Betshani, S. (2009). Analisis Data dengan Menggunakan ERD dan Model Konseptual Data Warehouse. *Jurnal Informatika*, 5(1).
- [7] Fajaryati, N., Priyanto, Sukardiyono, T., Wiji, A. D., Pambudi, S., Destiana, B. (2015). Studi Penelusuran (*Tracer Study*) Terhadap Alumni. *Jurnal Electronics, Informatics, and Vocational Education (ELINVO)*, 1(1).
- [8] FMIPA Universitas Negeri Jakarta. (2015). *Pedoman Akademik 2015/2016 Fakultas Matematika dan Ilmu Pengetahuan Alam (FMIPA) Universitas Negeri Jakarta*. Jakarta: Universitas Negeri Jakarta.
- [9] Fridayanthie, E. W., Mahdiati, T. (2016). Rancang Bangun Sistem Informasi Permintaan Atk Berbasis Intranet. *Jurnal Khatulistiwa Informatika*, 2(4).

- [10] Ghozaly, A., Nurcahyawati, V., Suhandiah, S. (2016). Rancang Bangun Aplikasi *Tracer Study* Berbasis Web Pada Stikes Yayasan Rs. Dr. Soetomo Surabaya. *Jurnal Sistem Informasi dan Komputer Akuntansi*, 5(6).
- [11] Hartomo, K. D., Wellem, T., Sanjaya, A. D. (2009). Implementasi Model View Controller dan Object Relational Mapping pada Content Management System Sistem Informasi Keuangan. *Jurnal Teknologi Informasi*, 6(1).
- [12] Haviluddin. (2011). Memahami Penggunaan UML (Unified Modelling Language). *Jurnal Informatika Mulawarman*, 6(1).
- [13] Hendini, A. (2016). Pemodelan Uml Sistem Informasi Monitoring Penjualan Dan Stok. *Jurnal Khatulistiwa Informatika*, 4(2).
- [14] Hermin, F. (2017). *Desain Sistem Informasi Alumni (SILUNI) Berbasis Website*. Jakarta: Universitas Negeri Jakarta.
- [15] Hidayat, A., Surarso, B. (2012). Penerapan Arsitektur Model View Controller (MVC) dalam Rancang Bangun Sistem Kuis Online Adaptif. *Seminar Nasional Teknologi Informasi dan Komunikasi 2012 (SENTIKA 2012)*.
- [16] Ilmu Komputer FMIPA UNJ. (2017). *Borang Akreditasi Program Studi Ilmu Komputer*. Jakarta: Universitas Negeri Jakarta.
- [17] Martono, K. T. (2018). Rancang Bangun Sistem Informasi Produksi Ayam Petelur Dengan Menggunakan Framework Codeigniter. *Jurnal Sistem Komputer*, 8(1).
- [18] Maryuliana, Subroto, M. I., Chairul, S. F. (2016). Sistem Informasi Angket Pengukuran Skala Kebutuhan Materi Pembelajaran Tambahan sebagai Pendukung Pengambilan Keputusan di Sekolah Menengah Atas Menggunakan Skala Likert. *Jurnal Transistor Elektro dan Informatika*, 1(2).

- [19] Muslihudin, M., Oktafianto. (2016). *Analisis dan Perancangan Sistem Informasi*. Yogyakarta: ANDI.
- [20] Nursubiyantoro, E., Puryani. (2016). Perancangan Sistem Penelusuran Alumni (*Tracer Study*) Berbasis Web. *Jurnal Optimasi Sistem Industri (OPSI)*, 9(2).
- [21] Pressman, R. S. (2002). *Rekayasa Perangkat Lunak*. LN Harnaningrum, penerjemah: Yogyakarta: Andi. dari: *Software Engineering, a Practitioner's Approach*. McGraw Hill Company, Inc.
- [22] Raharja, I. S., SN, A. (2012). Perbandingan Proses Pengembangan Perangkat Lunak. *Seminar Nasional Informatika 2012 (semnasIF 2012)*.
- [23] Renny, Chandra, R., Ruhana, S., Sarjono, M. W. (2013). Exploring Tracer Study Service in Career Center. *International Journal of Computer Science and Information Security (IJCSIS)*, 11(3).
- [24] Ristekdikti. (2018). *Panduan Program Bantuan Pengembangan Layanan Pusat Karir Lanjutan/Tracer Study 2019* [Online]. Diambil dari <http://pkts.belmawa.ristekdikti.go.id/upload/1548740521Panduan-Bantuan-PKL-2019.pdf>
- [25] Ristekdikti. (2018). Tentang *Tracer Study* [Online]. Diambil dari <http://pkts.belmawa.ristekdikti.go.id/index.php/site/about>
- [26] Sharma, M. K. (2017). A Study of SDLC to Develop Well Engineered Software. *International Journal of Advanced Research in Computer Science*, 8(3).
- [27] Sulianta, F., Juju, D. (2010). *Data Mining*. Jakarta: Elex Media Komputindo.
- [28] Sutanta, E. (2004). *Sistem Basis Data*. Yogyakarta: Graha Ilmu.

- [29] Yanto, R. (2016). *Manajemen Basis Data Menggunakan MySQL*. Yogyakarta: Deepublish.

LAMPIRAN A

Perbandingan Fitur Sistem *Tracer Study*

No	Fitur	UNJ		Dikti	Ilmu Komputer
		Prodi	Universitas		
1	Admin melihat form kuesioner	v	v	v	v
2	Admin dapat membuat dan mengubah form kuesioner	v	v		v
3	Admin mengelola kategori kuesioner	v			v
4	Admin dapat melihat hasil <i>tracer study</i> dalam bentuk grafik dan tabel	v	v	v	v
5	Admin dapat <i>export</i> data	v	v		v
6	Admin dapat <i>upload</i> data		v	v	v
7	Admin dapat mengecek kelulusan mahasiswa PDPT			v	
8	Admin dapat melihat statistik jumlah lulusan berdasarkan tahun		v	v	v
9	Admin dapat kelola data alumni		v	v	v
10	Admin dapat melihat daftar alumni		v	v	v
11	Admin mengelola daftar prodi		v		
12	Admin dapat mengelola kata pengantar di beranda alumni				v
13	Admin dapat mengganti password	v	v	v	v
14	Alumni dapat mengisi form kuesioner	v	v		v
15	Alumni dapat mengelola data diri	v	v		v
16	Alumni dapat melihat daftar pengguna alumni				v
17	Pengguna alumni dapat mengisi kuesioner pengguna				v
18	Koorprodi dapat melihat laporan/hasil <i>tracer study</i>				v
19	Koorprodi dapat <i>export</i> data				v
20	Koorprodi dapat melihat daftar alumni				v

No	Fitur	UNJ		Dikti	Ilmu Komputer
		Prodi	Universitas		
21	Koorprodi dapat melihat daftar pengguna alumni				v
22	Pengunjung dapat melihat daftar alumni				v
23	Pengunjung dapat melihat profil alumni				v
24	Pengunjung dapat melihat data <i>tracer study</i> /statistik alumni			v	v
25	Pengunjung dapat melihat daftar pengguna alumni				v

LAMPIRAN B

Perbandingan Kuesioner Sistem *Tracer Study*

Kuesioner Alumni	Prodi	Universitas (WR3)	Dikti	Ilmu Komputer
Data Diri				
Nama Lengkap	v	v	v	v
Nomor Mahasiswa	v	v	v	v
Tempat/Tanggal Lahir	v			v
Jenis Kelamin	v	v		v
Usia		v		
Alamat Rumah	v	v		v
No Telp/HP	v	v	v	v
Email	v	v	v	v
Kode Prodi			v	v
Kode PT			v	
Riwayat Pendidikan				
Tahun Masuk	v	v		v
Tahun Lulus	v	v	v	v
IPK	v	v		v
TOEFL	v			v
Lama masa studi				v
Lama penulisan skripsi				v
Sumber dana pembiayaan kuliah			v	
Besar penekanan pada metode pembelajaran di prodi			v	v
Saran masukan untuk perbaikan prodi	v			v
Melanjutkan Pendidikan				
Tempat Melanjutkan Pendidikan	v	v		v
Alamat Universitas		v		
Program studi	v	v		v
S1 Linear dengan S2		v		v
Nama Pelatihan Kompetensi Diri	v			v
Lama Pelatihan Kompetensi Diri	v			v

Kuesioner Alumni	Siluni	Universitas (WR3)	Dikti	Ilmu Komputer
Pekerjaan				
Kapan mulai mencari pekerjaan (dalam bulan)			v	
Nama tempat bekerja pertama kali	v			v
Jenis Pekerjaan pertama kali	v			v
Jabatan pekerjaan pertama kali	v			v
Masa tunggu lulus sampai mendapat pekerjaan	v		v	v
Rata-rata pendapatan pekerjaan pertama (per bulan)	v			v
Nama tempat bekerja saat ini	v			v
Alamat bekerja saat ini		v		v
Jenis perusahaan tempat bekerja saat ini			v	v
Jenis Pekerjaan Saat Ini	v	v		v
Jabatan pekerjaan saat ini	v	v		v
Rata-rata pendapatan pekerjaan saat ini (per bulan)	v	v	v	v
Pekerjaan berhubungan dengan bidang ilmu di perguruan tinggi	v		v	v
Alasan mengambil pekerjaan yang tidak sesuai dengan bidang ilmu			v	
Hambatan dalam menyesuaikan diri dengan pekerjaan	v			v
Tingkat pendidikan yang tepat/sesuai dengan pekerjaan			v	v
Cara memperoleh pekerjaan		v	v	v
Jumlah perusahaan yang pernah dilamar sebelum memperoleh pekerjaan pertama			v	v
Jumlah perusahaan yang merespon lamaran			v	v
Jumlah perusahaan yang mengundang untuk wawancara			v	v
Alasan belum bekerja		v		
Kendala belum bekerja yang dihadapi		v		
Aktif mencari pekerjaan dalam 4 minggu terakhir			v	
Nama usaha yang didirikan (wirausaha)		v		v

Kuesioner Alumni	Prodi	Universitas (WR3)	Dikti	Ilmu Komputer
Pekerjaan				
Bidang usaha		v		v
Lama usaha berdiri		v		v
Jumlah karyawan dalam usaha tersebut		v		v
Besar pendapatan dari usaha tersebut		v		v
Kompetensi				
Pengetahuan yang dibutuhkan untuk menunjang pekerjaan	v			v
Tingkat kompetensi yang dikuasai dalam pekerjaan			v	v
Mengikuti tes profesi di bidang tertentu	v			v
Mengikuti organisasi profesi	v			v

LAMPIRAN C

Analisis Kebutuhan Sistem dengan Pihak Prodi Ilmu Komputer

1. Kapan diadakannya kegiatan tracer study pada prodi Ilmu Komputer? Kegiatan tracer study telah dilakukan sejak prodi ini memiliki lulusan pertama pada semester 2016/2017.
2. Bagaimana proses pelaksanaan tracer study pada prodi Ilmu Komputer? Pelaksanaan tracer study tersebut dilakukan melalui pengiriman kuesioner melalui pos, e-mail dan media sosial seperti whatsapp
3. Apakah terdapat kendala selama proses tracer study berlangsung? Bentuk pelaksanaan tersebut dirasa kurang efektif karena dapat membutuhkan waktu lama terhadap respon alumni
4. Apakah saat ini sudah tersedia sistem tracer study yang dapat diakses secara online? Saat ini terdapat sistem tracer study milik WR3 yang juga memiliki layanan di tingkat prodi. Karena sistem yang disediakan bersifat umum untuk semua prodi terdapat beberapa kebutuhan dari prodi Ilkom yang belum bisa dipenuhi sistem tersebut seperti belum ada pengisian kuesioner oleh pengguna lulusan/alumni. Selain itu, juga terdapat sistem tracer study milik Dikti namun untuk tingkat perguruan tinggi.
5. Mengapa diperlukan kuesioner untuk pengguna alumni? Penelitian terhadap pengguna diperlukan untuk mengetahui bagaimana penilaian pengguna terhadap kompetensi lulusan dan kurikulum yang berjalan apakah sudah mencukupi dan relevan dengan kebutuhan dunia kerja saat ini.
6. Jika akan dilakukan pengembangan sistem tracer study dikhususkan untuk pro-

di Ilmu Komputer, fasilitas apa yang diharapkan dari pengembangan sistem ini?

- Selain kuesioner untuk alumni juga disediakan fitur pengisian kuesioner secara online untuk pengguna alumni
- Koorprodi membutuhkan akses ke sistem untuk mengakses data tracer yang diperlukan bagi akreditasi maupun penjaminan mutu internal
- Disediakan open access bagi pengunjung untuk dapat melihat beberapa hasil tracer study seperti yang terdapat pada sistem tracer study Dikti.
- Bandingkan sistem tracer study UNJ dan Dikti, ambil fitur yang sekiranya dapat bermanfaat untuk prodi.

Jakarta, April 2019

Ir. Fariani Hermin Indiyah, M.T

LAMPIRAN D

Sampel Kode *Controller* Alumni pada Admin

```
<?php
defined('BASEPATH') OR exit('No direct script access allowed');

class Alumni extends CI_Controller {
    function __construct() {
        parent::__construct();
        if($_SESSION["logged_in"] != 'login') {
            redirect(base_url("login"));
        }

        $this->load->model('m_alumni');
        $this->load->model('m_master');
        $this->load->model('m_pengguna');
        $this->load->library(array('PHPExcel','PHPExcel/IOFactory'));

    }

    public function index()
    {
        $prodiID = $this->session->userdata('prodiID');
        $data = array(
            'role' => $this->session->userdata('role'),
```

```

'userID' => $this->session->userdata('userID'),
'prodiID' => $prodiID,
'alumni' => $this->m_alumni->getAlumni($prodiID),
);
$this->load->view('element/head');
$this->load->view('element/header');
$this->load->view('element/navbar', $data);
$this->load->view('admin/v_dataAlumni', $data);
$this->load->view('element/footer');
}

```

```
function exeAddAlumni()
```

```
{
```

```
$userID = 'ALU'.$this->input->post('nim');
```

```
$cek = $this->m_alumni->cekAlumni($this->input->post('nim'));
```

```
if($cek->num_rows() > 0){
```

```
$this->session->set_flashdata("gagalAddAlumni", '<div><div class="a
```

```
} else {
```

```
$data_user = array(
```

```
'userID' => $userID,
```

```
'username' => $this->input->post('nim'),
```

```
'password' => $this->input->post('nim'),
```

```
'prodiID' => $this->session->userdata('prodiID'),
```

```

        'role' => 'alumni'
    );

    $this->m_master->inputData($data_user, 'user');

    $data_alumni = array(
        'nama' => $this->input->post('nama'),
        'nim' => $this->input->post('nim'),
        'userID' => $userID,
        'jenis_kelamin' => $this->input->post('jenis_kelamin'),
        'tahun_masuk' => $this->input->post('tahun_masuk'),
        'tahun_lulus' => $this->input->post('tahun_lulus'),
        'prodiID' => $this->session->userdata('prodiID'),
    );

    $this->m_master->inputData($data_alumni, 'alumni');

    $this->session->set_flashdata("suksesAddAlumni", '<div><div class="
}

    redirect('admin/Alumni');
}

public function editProfil($id)
{
    $data = array(
        'role' => $this->session->userdata('role'),

```

```

        'userID' => $this->session->userdata('userID'),
        'prodiID' => $this->session->userdata('prodiID'),
        'profil' => $this->m_alumni->getAlumniByID($id)
    );
    $this->load->view('element/head');
    $this->load->view('element/header');
    $this->load->view('element/navbar', $data);
    $this->load->view('admin/v_editProfilAlumni', $data);
    $this->load->view('element/footer');
}

function exeEditProfil()
{

    $data_user = array(
        'username' => $this->input->post('nim')
    );
    $id = $this->input->post('id');
    $where = array('id' => $this->input->post('userID'));
    $this->m_master->updateData($where, $data_user, 'user');

    $data_alumni = array(
        'nama' => $this->input->post('nama'),
        'nim' => $this->input->post('nim'),
        'jenis_kelamin' => $this->input->post('jenis_kelamin'),
        'tahun_masuk' => $this->input->post('tahun_masuk'),
    );
}

```

```

'tahun_lulus' => $this->input->post('tahun_lulus'),
'ipk' => $this->input->post('ipk'),
'toefl' => $this->input->post('toefl'),
'alamat' => $this->input->post('alamat'),
'tempat_lahir' => $this->input->post('tempat_lahir'),
'tanggal_lahir' => $this->input->post('tanggal_lahir'),
'email' => $this->input->post('email'),
'no_telepon' => $this->input->post('no_telepon'),
);

```

```

$where = array('id' => $this->input->post('id'));
$this->m_master->updateData($where,$data_alumni,'alumni');

```

```

$this->session->set_flashdata("pesan", '<div><div class="alert alert
redirect('admin/Alumni/editProfil/'.$id);
}

```

```

function exeImport()
{
if ($_FILES['file']['name']) {
$fileName = time().$_FILES['file']['name'];

$config['upload_path'] = FCPATH .'/assets/upload/';
$config['file_name'] = $fileName;
$config['allowed_types'] = 'xls|xlsx|csv';
$config['max_size'] = 10000;

```

```

$this->load->library('upload');
$this->upload->initialize($config);

if(! $this->upload->do_upload('file') )
    $this->upload->display_errors();

$media = $this->upload->data('file');
$inputFileName = $this->upload->data('full_path');

try {
    $inputFileType = IOFactory::identify($inputFileName);
    $objReader = IOFactory::createReader($inputFileType);
    $objPHPExcel = $objReader->load($inputFileName);
} catch(Exception $e) {
    die('Error loading file "'.pathinfo($inputFileName,PATHINFO_BASENAME)
}

$sheet = $objPHPExcel->getSheet(0);
$highestRow = $sheet->getHighestRow();
$highestColumn = $sheet->getHighestColumn();

for ($row = 2; $row <= $highestRow; $row++){ // R
    $rowData = $sheet->rangeToArray('A' . $row . ':' . $highestColumn .

    $nim = $rowData[0][1];

```

```

$cek = $this->m_alumni->cekAlumni($nim);

if($cek->num_rows() < 1){

//tabel user

$data = array(

"userID"=> "ALU".$rowData[0][1],

"username"=> $rowData[0][1],

"password"=> $rowData[0][1],

"role"=> 'alumni',

"prodiID" => $this->session->userdata('prodiID')

);

$insert = $this->db->insert("user",$data);

//tabel alumni

$data = array(

"nim"=> $rowData[0][1],

"nama"=> $rowData[0][0],

"no_telepon" => $rowData[0][2],

"ipk" => $rowData[0][3],

"userID"=> "ALU".$rowData[0][1],

"jenis_kelamin"=> $rowData[0][4],

"tahun_masuk"=> $rowData[0][5],

"tahun_lulus"=> $rowData[0][6],

"prodiID" => $this->session->userdata('prodiID')

);

$insert = $this->db->insert("alumni",$data);

} /*else {

```

```

unlink($inputFileName);

}*/

if ($rowData[0][7] != NULL) {
//tabel instansi
$where = array('nama_instansi' => $rowData[0][7]);
$cek = $this->m_master->cekData("instansi",$where)->num_rows();
if ($cek == 0) {
$data = array(
"nama_instansi" => $rowData[0][7],
"prodiID" => $this->session->userdata('prodiID')
);
$insert = $this->db->insert("instansi",$data);
$id_instansi = $this->m_master->getInstansiByName($rowData[0][7])->
} else {
$id_instansi = $this->m_master->getInstansiByName($rowData[0][7])->
}

//tabel pekerjaan
$alumniID = $this->m_alumni->getAlumniByUserID("ALU".$rowData[0][1])
$where = array( 'id_alumni' => $alumniID);
$cek = $this->m_master->cekData("pekerjaan",$where)->num_rows();
if ($cek == 0) {
$firstPekerjaan = 'yes';
} else {
$firstPekerjaan = 'no';
}
}

```



```

$gajiawal = str_replace(".", "", $rowData[0][9]);
$data = array(
    "posisi" => $rowData[0][8],
    "gaji" => $gajiawal,
    "id_alumni" => $alumniID,
    'firstPekerjaan' => $firstPekerjaan,
    'id_instansi' => $id_instansi
);
$insert = $this->db->insert("pekerjaan", $data);
} // if instansi not null

}

$this->session->set_flashdata("suksesImpor", ' <div><div class="alert
redirect('admin/Alumni');
} else {
$this->session->set_flashdata("gagalImpor", ' <div><div class="alert
redirect('admin/Alumni');
}
}

function deleteAlumni($id){
    //hapus pekerjaan
    $where = array(
        'id_alumni' => $id
    );

```

```

$this->m_master->deleteData($where,'pekerjaan');

//hapus akun di tabel user
$userID = $this->m_alumni->getAlumniByID($id)->userID;
$where = array(
    'userID' => $userID
);
$this->m_master->deleteData($where,'user');

//hapus data di tabel alumni
$where = array(
    'id' => $id
);
$this->m_master->deleteData($where,'alumni');

$where = array(
    'respondenID' => $id
);
$this->m_master->deleteData($where,'notif_kuesioner');
redirect('admin/Alumni');
}

}

```

LAMPIRAN E

Sampel Kode *Model* Alumni pada Admin

```
<?php

class M_alumni extends CI_Model{

    public function __construct()
    {
        parent::__construct();
        //$this->db2 = $this->load->database('db_dua', TRUE);
    }

    function getAlumniByUserID($userid)
    {
        $this->db->select('*');
        $this->db->where('userID',$userid);
        $query = $this->db->get('alumni');
        return $query->row();
    }

    function getAlumniByID($id)
    {
        $this->db->select('*');
```

```

$this->db->where('id', $id);
$query = $this->db->get('alumni');
return $query->row();

}

function getAlumni($prodiID)
{
$this->db->select('*');
$this->db->where('status', 'aktif');
$this->db->where('prodiID', $prodiID);
$this->db->order_by('id', 'DESC');
$query = $this->db->get('alumni');
if($query->num_rows()>0)
{
return $query->result();
} else{
return $query->result();
}

}

function getAlumniByProdi($prodiID)
{
$this->db->select('*');

```

```

$this->db->where('status', 'aktif');
$this->db->where('prodiID', $prodiID);
$this->db->order_by('nama', 'ASC');
$query = $this->db->get('alumni');
if($query->num_rows()>0)
{
return $query->result();
} else{
return $query->result();
}

}

```

```

public function cekAlumni($nim){
$this->db->where('nim', $nim);
return $this->db->get('alumni');
}

```

```

function getRiwayatByAlumniID($id)
{
$this->db->select('*');
$this->db->where('id_alumni', $id);
$this->db->order_by('id', 'ASC');
$query = $this->db->get('pekerjaan');
if($query->num_rows()>0)
{

```

```

return $query->result();
} else{
return $query->result();
}

```

```

}

```

```

function getAlumniPengguna($id)
{
$this->db->select('*');
$this->db->where('id_alumni', $id);
$this->db->order_by('id', 'ASC');
$query = $this->db->get('pekerjaan');
if($query->num_rows()>0)
{
return $query->result();
} else{
return $query->result();
}
}

```

```

function hapusAlumniPengguna($id_alumni,$id_pengguna){
$this->db->where($id_alumni);
$this->db->where($id_pengguna);
$this->db->delete('alumni_pengguna');
}

```

```
}
```

```
function getPekerjaanByAlumniID($id)
{
    $this->db->select('*');
    $this->db->where('id_alumni', $id);
    $query = $this->db->get('pekerjaan');
    if($query->num_rows()>0)
    {
        return $query->result();
    } else{
        return $query->result();
    }
}
```

```
function getPekerjaanByID($id)
{
    $this->db->select('*');
    $this->db->where('id', $id);
    $query = $this->db->get('pekerjaan');
    return $query->row();
}
```

```
public function joinPekerjaanAlumniByProdiID($prodiID) {
    $this->db->select('
```

```
pekerjaan.id AS id_pekerjaan, pekerjaan.*, alumni.id AS id_alumni,
');
$this->db->join('alumni', 'pekerjaan.id_alumni = alumni.id');
$this->db->from('pekerjaan');
$this->db->where('alumni.prodiID', $prodiID);
$this->db->where('pekerjaan.gaji !=', "");
$query = $this->db->get();
return $query->result();
}

}
```


LAMPIRAN F

Kuesioner *User Acceptance Test* pada Admin



KUESIONER INSTRUMEN *USER ACCEPTANCE TEST* SISTEM INFORMASI TRACER STUDY PRODI ILMU KOMPUTER FMIPA UNJ

Kepada:

Yth. Bapak/Tbu/Sdr/I
Staf Program Studi Ilmu Komputer FMIPA Universitas Negeri Jakarta
di Tempat,

Dengan Hormat,

Saya mahasiswi Program Studi S1 Ilmu Komputer, Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Negeri Jakarta yang pada saat ini sedang melakukan penelitian dalam rangka menyusun tugas akhir (skripsi),

Nama : Saulia Karina

NIM : 3145153851

Bersama ini penulis mohon kesediaan Bapak/Tbu/Sdr/I untuk membantu mengisi kuesioner yang telah disiapkan dengan maksud untuk mengetahui seberapa jauh Sistem Informasi *Tracer Study* Program Studi Ilmu Komputer FMIPA UNJ dapat memenuhi kebutuhan dari pengguna aplikasi ini.

Sebelum memberikan penilaian pernyataan-pernyataan yang ada dalam kuesioner ini, penulis mohon agar Bapak/Tbu/Sdr/I membaca petunjuk pengisian kuesioner dengan teliti. Dimohon untuk mengisi pernyataan dengan keyakinan tinggi tanpa keraguan sesuai dengan kondisi yang ada.

Untuk menambah keyakinan Bapak/Tbu/Sdr/I dalam mengisi setiap pernyataan, diinformasikan bahwa data yang diperoleh dari kuesioner ini hanya akan digunakan untuk penelitian.

Hormat Saya,

Saulia Karina

A. Petunjuk Pengisian Kuesioner

1. Mohon untuk membaca setiap pernyataan dengan teliti.
2. Pilihlah salah satu jawaban yang paling sesuai dengan kondisi yang ada kemudian berikan tanda ceklis (✓) pada salah satu jawaban, dengan keterangan sebagai berikut Setuju (S), Tidak Setuju (TS) untuk uji fungsionalitas
Berikan tanda ceklis (✓) pada salah satu jawaban dengan keterangan point sebagai berikut : 1 (Sangat Tidak Setuju), 2 (Kurang Setuju), 3 (Cukup), 4 (Setuju), dan 5 (Sangat Setuju) untuk uji *usability*
3. Mohon untuk mengisi setiap pernyataan yang ada dalam kuesioner ini, karena penulis membutuhkan tanggapan untuk kepentingan penelitian.

B. Identitas Responden

1. Nama :
2. Jabatan :
3. Tanggal Pengisian :

1. Kuesioner *functionality* pada admin

Keterangan

S : Setuju

TS : Tidak Setuju

Tolong berikan jawaban pertanyaan berikut sesuai dengan pendapat Anda, dengan cara memberikan tanda (✓) pada kolom yang tersedia

No	Keterangan	S	TS
1	Fitur masuk ke dalam sistem (<i>login</i>) berjalan dengan baik	✓	
2	Fitur mengelola (menambah, menyunting, dan menghapus) data alumni berjalan dengan baik dan sesuai kebutuhan	✓	
3	Fitur mengelola (menyunting, dan menghapus) data pengguna alumni berjalan dengan baik dan sesuai kebutuhan	✓	
4	Fitur menyunting kata pengantar <i>tracer study</i> berjalan dengan baik sesuai kebutuhan	✓	
5	Fitur mengelola (membuat, mengedit, dan menghapus) kuesioner alumni berjalan dengan baik dan sesuai kebutuhan	✓	
5	Fitur mengelola (membuat, mengedit, dan menghapus) kuesioner pengguna alumni berjalan dengan baik sesuai kebutuhan	✓	
7	Fitur melihat laporan hasil <i>tracer study</i> berjalan dengan baik sesuai kebutuhan	✓	
8	Fitur <i>export</i> data <i>tracer</i> berjalan dengan baik dan sesuai kebutuhan	✓	
9	Fitur mengganti <i>password</i> berjalan dengan baik	✓	
10	Fitur keluar sistem (<i>logout</i>) berjalan dengan baik	✓	

2. Saran dan Masukan

.....

.....

.....

.....

1. Kuesioner *usability* admin

- 1 : Sangat Tidak Setuju
 2 : Tidak Setuju
 3 : Cukup
 4 : Setuju
 5 : Sangat Setuju

Tolong berikan jawaban pertanyaan berikut sesuai dengan pendapat Anda, dengan cara memberikan tanda (✓) pada kolom yang tersedia

No	Keterangan	1	2	3	4	5
1	Fitur mengelola data alumni mudah dimengerti				✓	
2	Fitur mengelola data pengguna alumni mudah dimengerti				✓	
3	Fitur menyunting kata pengantar <i>tracer study</i> mudah dimengerti				✓	
4	Fitur membuat kuesioner alumni dan pengguna alumni mudah dimengerti				✓	
5	Fitur melihat laporan hasil <i>tracer study</i> mudah dimengerti				✓	
6	Fitur <i>export</i> data <i>tracer</i> mudah dimengerti				✓	
7	Sistem ini membantu dalam pengelolaan data alumni				✓	
8	Sistem ini mudah untuk digunakan				✓	
9	Sistem ini nyaman untuk digunakan				✓	

2. Saran dan Masukan

.....

.....

.....

.....

.....

.....

.....

LAMPIRAN G

Kuesioner *User Acceptance Test* pada Koorprodi



KUESIONER INSTRUMEN *USER ACCEPTANCE TEST* SISTEM INFORMASI TRACER STUDY PRODI ILMU KOMPUTER FMIPA UNJ

Kepada:

Yth. Bapak/Tbu/Sdr/I
Staf Program Studi Ilmu Komputer FMIPA Universitas Negeri Jakarta
di Tempat,

Dengan Hormat,

Saya mahasiswi Program Studi S1 Ilmu Komputer, Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Negeri Jakarta yang pada saat ini sedang melakukan penelitian dalam rangka menyusun tugas akhir (skripsi),

Nama : Saulia Karina

NIM : 3145153851

Bersama ini penulis mohon kesediaan Bapak/Tbu/Sdr/I untuk membantu mengisi kuesioner yang telah disiapkan dengan maksud untuk mengetahui seberapa jauh Sistem Informasi *Tracer Study* Program Studi Ilmu Komputer FMIPA UNJ dapat memenuhi kebutuhan dari pengguna aplikasi ini.

Sebelum memberikan penilaian pernyataan-pernyataan yang ada dalam kuesioner ini, penulis mohon agar Bapak/Tbu/Sdr/I membaca petunjuk pengisian kuesioner dengan teliti. Dimohon untuk mengisi pernyataan dengan keyakinan tinggi tanpa keraguan sesuai dengan kondisi yang ada.

Untuk menambah keyakinan Bapak/Tbu/Sdr/I dalam mengisi setiap pernyataan, diinformasikan bahwa data yang diperoleh dari kuesioner ini hanya akan digunakan untuk penelitian.

Hormat Saya,

Saulia Karina

A. Petunjuk Pengisian Kuesioner

1. Mohon untuk membaca setiap pernyataan dengan teliti.
2. Pilihlah salah satu jawaban yang paling sesuai dengan kondisi yang ada kemudian berikan tanda ceklis (✓) pada salah satu jawaban, dengan keterangan sebagai berikut Setuju (S), Tidak Setuju (TS) untuk uji fungsionalitas
Berikan tanda ceklis (✓) pada salah satu jawaban dengan keterangan point sebagai berikut : 1 (Sangat Tidak Setuju), 2 (Kurang Setuju), 3 (Cukup), 4 (Setuju), dan 5 (Sangat Setuju) untuk uji *usability*
3. Mohon untuk mengisi setiap pernyataan yang ada dalam kuesioner ini, karena penulis membutuhkan tanggapan untuk kepentingan penelitian.

B. Identitas Responden

1. Nama :
2. Jabatan :
3. Tanggal Pengisian :

1. Kuesioner *functionality* pada Koorprodi

Keterangan

S : Setuju

TS : Tidak Setuju

Tolong berikan jawaban pertanyaan berikut sesuai dengan pendapat Anda, dengan cara memberikan tanda (✓) pada kolom yang tersedia

No	Keterangan	S	TS
1	Fitur masuk ke dalam sistem (<i>login</i>) berjalan dengan baik	✓	
2	Fitur melihat data alumni berjalan dengan baik dan sesuai kebutuhan	✓	
3	Fitur melihat data pengguna alumni berjalan dengan baik dan sesuai kebutuhan	✓	
4	Fitur melihat laporan hasil <i>tracer study</i> berjalan dengan baik dan sesuai kebutuhan	✓	
5	Fitur <i>export</i> data <i>tracer</i> berjalan dengan baik dan sesuai kebutuhan	✓	
6	Fitur mengganti <i>password</i> berjalan dengan baik	✓	
7	Fitur keluar sistem (<i>logout</i>) berjalan dengan baik	✓	

2. Saran dan Masukan

.....

.....

.....

.....

.....

.....

.....

1. Kuesioner *Usability* Koorprodi

- 1 : Sangat Tidak Setuju
 2 : Tidak Setuju
 3 : Cukup
 4 : Setuju
 5 : Sangat Setuju

Tolong berikan jawaban pertanyaan berikut sesuai dengan pendapat Anda, dengan cara memberikan tanda (✓) pada kolom yang tersedia

No	Keterangan	1	2	3	4	5
1	Fitur melihat data alumni mudah dimengerti					✓
2	Fitur melihat data pengguna alumni mudah dimengerti					✓
3	Fitur melihat hasil laporan <i>tracer study</i> mudah dimengerti					✓
4	Sistem ini membantu prodi dalam memenuhi kebutuhan data alumni					✓
5	Sistem ini mudah untuk digunakan					✓
6	Sistem ini nyaman untuk digunakan					✓

2. Saran dan Masukan

.....

.....

.....

.....

.....

.....

.....

LAMPIRAN H

Kuesioner *User Acceptance Test* pada Alumni



KUESIONER INSTRUMEN *USER ACCEPTANCE TEST* SISTEM INFORMASI TRACER STUDY PRODI ILMU KOMPUTER FMIPA UNJ

Kepada:

Yth. Bapak/Tbu/Sdr/I
Staf Program Studi Ilmu Komputer FMIPA Universitas Negeri Jakarta
di Tempat,

Dengan Hormat,

Saya mahasiswi Program Studi S1 Ilmu Komputer, Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Negeri Jakarta yang pada saat ini sedang melakukan penelitian dalam rangka menyusun tugas akhir (skripsi),

Nama : Saulia Karina

NIM : 3145153851

Bersama ini penulis mohon kesediaan Bapak/Tbu/Sdr/I untuk membantu mengisi kuesioner yang telah disiapkan dengan maksud untuk mengetahui seberapa jauh Sistem Informasi *Tracer Study* Program Studi Ilmu Komputer FMIPA UNJ dapat memenuhi kebutuhan dari pengguna aplikasi ini.

Sebelum memberikan penilaian pernyataan-pernyataan yang ada dalam kuesioner ini, penulis mohon agar Bapak/Tbu/Sdr/I membaca petunjuk pengisian kuesioner dengan teliti. Dimohon untuk mengisi pernyataan dengan keyakinan tinggi tanpa keraguan sesuai dengan kondisi yang ada.

Untuk menambah keyakinan Bapak/Tbu/Sdr/I dalam mengisi setiap pernyataan, diinformasikan bahwa data yang diperoleh dari kuesioner ini hanya akan digunakan untuk penelitian.

Hormat Saya,

Saulia Karina

A. Petunjuk Pengisian Kuesioner

1. Mohon untuk membaca setiap pernyataan dengan teliti.
2. Pilihlah salah satu jawaban yang paling sesuai dengan kondisi yang ada kemudian berikan tanda ceklis (✓) pada salah satu jawaban, dengan keterangan sebagai berikut Setuju (S), Tidak Setuju (TS) untuk uji fungsionalitas
Berikan tanda ceklis (✓) pada salah satu jawaban dengan keterangan point sebagai berikut : 1 (Sangat Tidak Setuju), 2 (Kurang Setuju), 3 (Cukup), 4 (Setuju), dan 5 (Sangat Setuju) untuk uji *usability*
3. Mohon untuk mengisi setiap pernyataan yang ada dalam kuesioner ini, karena penulis membutuhkan tanggapan untuk kepentingan penelitian.

B. Identitas Responden

1. Nama :
2. Tahun Lulus :
3. Tanggal Pengisian :

1. Kuesioner *functionality* pada Alumni

Keterangan

S : Setuju

TS : Tidak Setuju

Tolong berikan jawaban pertanyaan berikut sesuai dengan pendapat Anda, dengan cara memberikan tanda (✓) pada kolom yang tersedia

No	Keterangan	S	TS
1	Fitur masuk ke dalam sistem (<i>login</i>) berjalan dengan baik		
2	Fitur menyunting data diri berjalan dengan baik		
3	Fitur mengelola (menambah, mengedit, dan menghapus) riwayat pekerjaan berjalan dengan baik		
4	Fitur mengisi form kuesioner <i>tracer study</i> berjalan dengan baik		
5	Fitur melihat daftar pengguna alumni beserta alumni yang bekerja berjalan dengan baik		
6	Fitur mengganti <i>password</i> berjalan dengan baik		
7	Fitur keluar sistem (<i>logout</i>) berjalan dengan baik		

2. Saran dan Masukan

.....

.....

.....

.....

.....

.....

.....

1. Kuesioner *Usability* Alumni

- 1 : Sangat Tidak Setuju
 2 : Tidak Setuju
 3 : Cukup
 4 : Setuju
 5 : Sangat Setuju

Tolong berikan jawaban pertanyaan berikut sesuai dengan pendapat Anda, dengan cara memberikan tanda (✓) pada kolom yang tersedia

No	Keterangan	1	2	3	4	5
1	Fitur menyunting data diri mudah dimengerti					
2	Fitur mengelola (menambah, menyunting, dan menghapus) riwayat pekerjaan mudah dimengerti					
3	Fitur mengisi form kuesioner mudah dimengerti					
4	Fitur melihat daftar pengguna alumni beserta alumni yang bekerja mudah dimengerti					
5	Sistem ini membantu alumni dalam mendapat informasi terkait jenjang pekerjaan bidang Ilmu Komputer					
6	Sistem ini mudah untuk digunakan					
7	Sistem ini nyaman untuk digunakan					

2. Saran dan Masukan

.....

.....

.....

.....

.....

.....

.....

LAMPIRAN I

Kuesioner *User Acceptance Test* pada Pengguna Alumni



KUESIONER INSTRUMEN *USER ACCEPTANCE TEST* SISTEM INFORMASI TRACER STUDY PRODI ILMU KOMPUTER FMIPA UNJ

Kepada:

Yth. Bapak/Tbu/Sdr/I
Staf Program Studi Ilmu Komputer FMIPA Universitas Negeri Jakarta
di Tempat,

Dengan Hormat,

Saya mahasiswi Program Studi S1 Ilmu Komputer, Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Negeri Jakarta yang pada saat ini sedang melakukan penelitian dalam rangka menyusun tugas akhir (skripsi),

Nama : Saulia Karina

NIM : 3145153851

Bersama ini penulis mohon kesediaan Bapak/Tbu/Sdr/I untuk membantu mengisi kuesioner yang telah disiapkan dengan maksud untuk mengetahui seberapa jauh Sistem Informasi *Tracer Study* Program Studi Ilmu Komputer FMIPA UNJ dapat memenuhi kebutuhan dari pengguna aplikasi ini.

Sebelum memberikan penilaian pernyataan-pernyataan yang ada dalam kuesioner ini, penulis mohon agar Bapak/Tbu/Sdr/I membaca petunjuk pengisian kuesioner dengan teliti. Dimohon untuk mengisi pernyataan dengan keyakinan tinggi tanpa keraguan sesuai dengan kondisi yang ada.

Untuk menambah keyakinan Bapak/Tbu/Sdr/I dalam mengisi setiap pernyataan, diinformasikan bahwa data yang diperoleh dari kuesioner ini hanya akan digunakan untuk penelitian.

Hormat Saya,

Saulia Karina

A. Petunjuk Pengisian Kuesioner

1. Mohon untuk membaca setiap pernyataan dengan teliti.
2. Pilihlah salah satu jawaban yang paling sesuai dengan kondisi yang ada kemudian berikan tanda ceklis (✓) pada salah satu jawaban, dengan keterangan sebagai berikut Setuju (S), Tidak Setuju (TS) untuk uji fungsionalitas
Berikan tanda ceklis (✓) pada salah satu jawaban dengan keterangan point sebagai berikut : 1 (Sangat Tidak Setuju), 2 (Kurang Setuju), 3 (Cukup), 4 (Setuju), dan 5 (Sangat Setuju) untuk uji *usability*
3. Mohon untuk mengisi setiap pernyataan yang ada dalam kuesioner ini, karena penulis membutuhkan tanggapan untuk kepentingan penelitian.

B. Identitas Responden

1. Nama :
2. Jabatan :
3. Tanggal Pengisian :

1. Kuesioner *functionality* pada Pengguna Alumni

Keterangan

S : Setuju

TS : Tidak Setuju

Tolong berikan jawaban pertanyaan berikut sesuai dengan pendapat Anda, dengan cara memberikan tanda (✓) pada kolom yang tersedia

No	Keterangan	S	TS
1	Fitur mengisi form kuesioner pengguna alumni berjalan dengan baik		

2. Saran dan Masukan

.....

.....

.....

.....

.....

.....

.....

.....

1. Kuesioner *Usability* Pengguna Alumni

- 1 : Sangat Tidak Setuju
 2 : Tidak Setuju
 3 : Cukup
 4 : Setuju
 5 : Sangat Setuju

Tolong berikan jawaban pertanyaan berikut sesuai dengan pendapat Anda, dengan cara memberikan tanda (✓) pada kolom yang tersedia

No	Keterangan	STS	TS	C	S	SS
1	Fitur mengisi kuesioner pengguna alumni mudah dimengerti					
2	Sistem ini mudah digunakan					
3	Sistem ini nyaman digunakan					

2. Saran dan Masukan

.....

.....

.....

.....

.....

.....

.....

.....

LAMPIRAN J

Dokumentasi Pengembangan Sistem dengan Model *Spiral*

Iterasi	Analisis Kebutuhan (Communication and Planning)	Desain (Modeling)	Pengkodean (Construction)	Evaluasi (Deployment)
I	<ul style="list-style-type: none"> - Memiliki enam pengguna, yaitu admin, koorprodi, alumni, pengguna alumni dan pengunjung. - Admin : <i>login</i>, kelola data alumni dan pengguna alumni, kelola kuesioner, kelola konten beranda, menampilkan hasil <i>tracer study</i>, dan menampilkan statistik jumlah lulusan Ilmu Komputer - Koorprodi : <i>login</i>, menampilkan data alumni dan pengguna alumni, menampilkan hasil <i>tracer study</i>, dan menampilkan statistik jumlah lulusan Ilmu Komputer - Alumni : <i>login</i>, kelola data diri dan mengisi kuesioner yang telah dibuat oleh admin - Pengguna alumni : <i>login</i>, kelola profil dan mengisi form kuesioner. - Pengunjung : melakukan pencarian alumni, melihat daftar pengguna alumni, dan melihat data hasil <i>tracer</i>. - Pembuatan sistem akan menggunakan <i>framework</i> CodeIgniter, Bahasa pemrograman PHP dan tampilan dengan HTML dan <i>Bootstrap</i> 	<ul style="list-style-type: none"> - Membuat UML dan ERD - Rancangan tampilan / <i>Mock-up</i> 	<ul style="list-style-type: none"> - Membuat <i>database</i> - Membuat tampilan antar muka/<i>interface</i> 	<ul style="list-style-type: none"> - Pengguna alumni tidak perlu <i>login</i> - Pengguna alumni jadikan sub-aktor dari pengunjung - Pengguna alumni hanya mengisi kuesioner

II	<ul style="list-style-type: none"> - Memiliki lima pengguna, yaitu admin, koorprodi, alumni, dan pengunjung yang memiliki sub-aktor pengguna alumni. - Pengguna alumni dapat mengisi kuesioner 	<ul style="list-style-type: none"> - Revisi UML dan ERD - Revisi <i>Mock-up</i> 	<ul style="list-style-type: none"> - Merivisi <i>database</i> - Menghapus <i>dashboard</i> pengguna alumni - Membuat halaman pengisian kuesioner pengguna alumni - Melakukan pengkodean <i>back end (model dan controller)</i> 	<ul style="list-style-type: none"> - Pengisian semua data alumni yang terdiri dari data diri, pekerjaan, dan kuesioner lainnya dijadikan satu halaman, yaitu di <i>landing page</i>
III	<ul style="list-style-type: none"> - Mengumpulkan kuesioner <i>tracer study</i> yang dibutuhkan program studi 	<ul style="list-style-type: none"> - Merancang tampilan <i>landing page/ Beranda Alumni</i> terbaru 	<ul style="list-style-type: none"> - Merevisi <i>landing page/ beranda</i> alumni dengan menambahkan semua form pengisian data <i>tracer study</i> - Melakukan pengkodean <i>back end (model dan controller)</i> 	<ul style="list-style-type: none"> - Membuat tampilan status pengisian kuesioner alumni - Memperluas cakupan sistem yaitu sampai fakultas

DAFTAR RIWAYAT HIDUP



SAULIA KARINA. Lahir di Jakarta, 27 Juni 1997 Anak pertama dari pasangan Bapak Syarifudin dan Ibu Ana Yohana. Saat ini beralamatkan di Jl. Pisangan Baru I RT.006 RW.08 No.4, Kecamatan Matraman Jakarta Timur.

No. Ponsel : 083843549167

Email : sauliakarina@gmail.com

Riwayat Pendidikan : Penulis mengawali pendidikan di TK Nurul Iman pada tahun 2002 - 2003, dan kemudian melanjutkan pendidikan di SDN Pisangan Baru 01 Pagi pada tahun 2003 - 2009. Setelah itu, penulis melanjutkan ke SMPN 7 Jakarta hingga tahun 2012. Kemudian melanjutkan ke SMAN 54 Jakarta pada tahun 2012-2015. Di Tahun 2015 penulis melanjutkan ke Universitas Negeri Jakarta (UNJ), Program Studi Ilmu Komputer, melalui jalur PENMABA. Di awal tahun 2020 (D, dd Februari 2020) penulis telah memperoleh gelar Sarjana Komputer (S.Kom), Program Studi Ilmu Komputer, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Negeri Jakarta.

Riwayat Organisasi : Selama di bangku perkuliahan, penulis aktif di Badan Eksekutif Mahasiswa Rumpun Matematika sebagai staff di Departemen Advokasi dan Keolahragaan kemudian melanjutkan di BEM FMIPA dengan departemen yang sama. Penulis juga berpartisipasi dalam kegiatan Biner (Be Innovative and Educated Researcher) yaitu kegiatan workshop dan seminar yang diadakan oleh DEFAULT, dimana penulis tergabung sebagai anggota merangkap Bendahara.