

LAPORAN KERJA PARAKTEK

SISTEM INFORMASI PENDATAAN ALUMIN BERBASIS

WEB DI SMK MUHAMMADIYAH 2 BANDUNG

Diajukan Untuk Memenuhi Syarat Kelulusan Mata Kuliah Kerja Praktik



Disusun Oleh :
M Rizky Khoeruddin 1157050086

JURUSAN TEKNIK INFORMATIKA

FAKULTAS SAINS DAN TEKNOLOGI

UNIVERSITAS ISLAM NEGERI SUNAN GUNUNG DJATI

BANDUNG

1441 H / 2020 M

LEMBAR PERSETUJUAN

LAPORAN KERJA PRAKTIK

**SISTEM INFORMASI PENDATAAN ALUMIN BERBASIS WEB SMK
MUHAMMADIYAH 2 BANDUNG**

Disusun Oleh :

M Rizky Khoeruddin

1157050086

Telah diperiksa dan disetujui oleh pembimbing

Pada tanggal :

Pembimbing Instansi

Pembimbing Kerja Praktik



Asep Widiana., S.Kom.
NBM. 1144.275



Dena Rauda Ramdania., S.Pd.MT.,M.Eng
NIP. 198904152018012002

Mengetahui,

Ketua Jurusan Teknik Informatika



Cepv Slamet.,ST.,M.Kom
NIP. 198002252011011007

LEMBAR PENGESAHAN

LAPORAN KERJA PRAKTIK

**SISTEM INFORMASI PENDATAAN ALUMIN BERBASIS WEB SMK
MUHAMMADIYAH 2 BANDUNG**

Disusun Oleh :

M Rizky Khoeruddin

1157050086

Telah diperiksa dan disetujui oleh pembimbing

Pada tanggal :

Pembimbing Instansi



Asep Widiana., S.Kom.
NBM. 1144.275

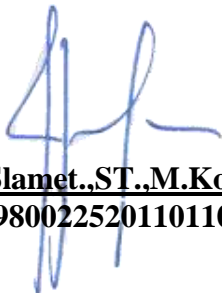
Pembimbing Kerja Praktik



Dena Rauda Ramdania., S.Pd.MT.,M.Eng
NIP. 198904152018012002

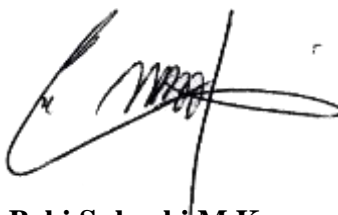
Mengetahui,

Ketua Jurusan Teknik Informatika



Cepv Slamet.,ST.,M.Kom.
NIP. 198002252011011007

Dosen Pengujian



Beki Subaeki,M.Kom.
NIP / NIDN. 0405098202

KATA PENGANTAR

Assalamu 'alaikum Wr. Wb.

Alhamdulillah puji serta syukur penulis panjatkan atas khadirat Allah SWT, karena berkat izin-Nyalah laporan Kerja Praktik ini dapat terselesaikan. Tidak lupa shalawat serta salam senantiasa semoga tercurah kepada junjunan Nabi kita, Habibana Wa Nabiyana Muhammad SAW beserta keluarga, para sahabat, dan kita selaku umatnya sampai akhir zaman.

Penulisan laporan kerja praktik ini dimaksudkan untuk melengkapi salah satu mata kuliah di Jurusan Teknik Informatika UIN Sunan Gunung Djati Bandung. Adapun judul kerja praktik yang telah diselesaikan yakni berjudul **“SISTEM INFORMASI PENDATAAN ALUMNI BERBASIS WEB DI SMK MUHAMMADIYAH 2 CIBIRU”**.

Dengan segala kerendahan hati mengucapkan terima kasih yang sebesar-besarnya kepada :

1. Allah SWT atas karunia dan rahmat-Nya
2. Nabi Muhammad SAW sebagai panutan hidup
3. Kedua orang tua yang senantiasa memberikan semua hal, baik materil maupun inmateril.
4. Yang terhormat Mohamad Irfan.,M.Kom selaku Ketua Jurusan Teknik Informatika yang senantiasa memberikan dorongan serta motivasi kepada para mahasiswanya.
5. Yang terhormat Drs Dedeng Mulyadi, M.M.Pd Selaku menerima kerja praktek dari Kepala Sekolah Di SMK MUHAMMADIYAH 2 CIBIRU.

6. Yang terhormat Iwan Setiawan ST., M.Kom sebagai dosen pembimbing kerja praktik yang telah senantiasa membantu dalam proses kerja praktik ini.
7. Yang terhormat Asep Widiani, S.Kom selaku pembimbing di SMK MUHAMMADIYAH 2 CIBIRU.
8. Kakak, teman-teman dan semua pihak yang sudah memberi dorongan serta bersedia berbagi informasi sehingga banyak ilmu dan pemikiran baru yang sebelumnya tidak tahu.

Penulis sadar bahwa di dunia ini tidak ada yang sempurna termasuk laporan kerja praktik yang disusun. Penulis masih membutuhkan masukan yang dapat membangun menjadi lebih baik di masa yang akan datang. Oleh karena itu, penulis mengharapkan kritik dan saran yang akan membuat motivasi dalam kehidupan penulis.

Wassalamu 'alaikum Wr. Wb.

Bandung, Agustus 2019

Penulis

DAFTAR ISI

Halaman Judul	i
Halaman Pengesahan	i
Kata Pengantar	ii
Daftar Isi	iv
Daftar Gambar	ix
Daftar Tabel	ix

BAB I PENDAHULUAN

1.1. Latar Belakang	1
1.2. Rumusan Masalah	3
1.3. Batasan Masalah	3
1.4. Tujuan Kerja Praktik	4
1.5. Metode Kerja Praktik	5
1.6. Tempat dan Jadwal Kerja Praktik	7
1.7. Sistematika Penulisan	7

BAB II LANDASAN TEORI

2.1. Alumni	9
2.2. Sistem Informasi	10
2.3. Administrator	10
2.4. Aplikasi	11
2.5. Website	11

2.5.1. Pengertian <i>Website</i>	12
2.5.2. Sejarah <i>Website</i>	12
2.5.3. <i>World Wide Web</i>	13
2.5.4. Komponen Pembangun <i>Website</i>	14
2.6. Portal.....	17
2.7. <i>Internet</i>	17
2.8. Medel Pembangunan Perangkat Lunak <i>Prototype</i>	17
2.9. UML (<i>Unifled Modeling Language</i>)	19
2.10. ERD (<i>Entity Relationship Diagram</i>)	39
2.11. Kamus Data	40
2.12. Struktur Database	41
2.12.1. Peranangan Database.....	41
2.13. Model Pengujian Perang Lunak <i>Black Box</i>	42
BAB III TINJAUAN UMUM SMK MUHAMMDIYAH 2 BANDUNG	
3.1. Profil Sekolah SMK Muhammadiyah	44
3.2. Visi dan Misi SMK Muhammdiyah 2 Bandung.....	44
3.2.1. Visi.....	45
3.2.2. Misi	45
3.2.3. Tujuan	46
3.3. Struktur Organisasi	46
3.4. Kondisi Umum	47
BAB IV ANALISIS DAN PERANCANGAN	
4.1. Analisa Sistem.....	48
4.1.1. Analisa Sistem yang Sedang Berjalan	48

4.1.2. Sistem yang diharapkan	49
4.2. Analisa Kebutuhan	50
4.2.1. Kebutuhan Perangkat Keras (Hardware)	50
4.2.2. Kebutuhan Perangkat Keras (Lunak).....	50
4.2.3. Kebutuhan Fungsional	50
4.3. Perancangan Sistem	51
4.4. Rancangan Fungsional (<i>Functional Design</i>).....	51
4.5. ERD (<i>Entity Relationship Diagram</i>)	72
4.6. Kamus Data.....	72
4.7. Struktur Database Website Alumni SMK Muhammadiyah 2 BANDUNG.....	73
4.7.1. Struktur Tabel.....	74
4.8. Perancangan Antarmuka (<i>Use Interface</i>).....	79
4.8.1. Halaman Home	79
4.8.2. Halaman dashboard admin	79
4.8.3. Halaman Daftar Biodata Alumni	80
4.8.4. Halaman Tampilan Daftar Alumni	84
4.8.5. Halaman Laporan Data	84
 BAB V IMPLEMENTASI	
5.1. Implementasi	86
5.1.1. Implementasi Perangkat Lunak	86
5.1.2. Implementasi Perangkat Keras (Hardware)	86
5.1.3. Konversi Sistem.....	87
5.1.4. Pengujian Sistem	87
5.1.5. Pemasangan Sistem.....	87

5.1.6. Coding	88
5.1.7. Implementasi Tabel.....	88
5.1.8. Implementasi Tabel.....	89
5.2. Implementasi Antar Muka	93
5.3. Pengujian Sistem	94
5.3.1. Rencana Pengujian Blackbox	95
5.3.2. Pengujian Antarmuka Halaman Administrator	95
 BAB VI KESIMPULAN DAN SARAN	
6.1. Kesimpulan	98
6.2. Saran	98
 DAFTAR PUSTAKA	
100	
 LAMPIRAN.....	
101	

DAFTAR GAMBAR

Gambar 2.1 Diagram UML	23
Gambar 3.2 Struktur Organisasi SMK 2 Bandung	46
Gambar 4.1 Usecase Diagram	52
Gambar 4.2 Usecase Diagram	55
Gambar 4.3 Class Diagram	58
Gambar 4.4 <i>Statechart</i> Diagram	60
Gambar 4.5 Sequence Diagram	63
Gambar 4.6 Collaboration Diagram	65
Gambar 4.7 Component Diagram	67
Gambar 4.8 Deployment Diagram	70
Gambar 4.9 ERD (<i>Entity Relationship Diagram</i>)	72
Gambar 4.10 Struktur Database	73
Gambar 4.11 Halaman Home	79
Gambar 4.12 Halaman <i>Dashboard admin</i>	80
Gambar 4.13 Halaman Tambah lengkapi siswa alumni	81
Gambar 4.14 Halaman Keluarga	81
Gambar 4.15 Halaman Orang Tua	82
Gambar 4.16 Halaman Kontak	82

Gambar 4.17 Data Terkini.....	83
Gambar 4.18 Akademik	83
Gambar 4.19 Halaman Tampilan data Anggota siswa alumni	84
Gambar 4.20 Halaman Laporan Data	85
Gambar 5.1 Database db_alumniSmk2.....	88
Gambar 5.2 Tabel Admin.....	89
Gambar 5.3 Tabel data_anggota siswa	90
Gambar 5.4 Tabel anggota Keluarga	91
Gambar 5.5 Tabel Kontak	91
Gambar 5.6 Tabel Orangtua	92
Gambar 5.7 Tabel Terkini	92
Gambar 5.8 Tabel Akademik	93
Gambar 5.9 Halaman Login	93
Gambar 5.10 Dashboard Administrator	94

DAFTAR TABEL

Tabel 2.1 Simbol-simbol Class Diagram	28
Tabel 2.2 Simbol-simbol Komponen Diagram	28
Tabel 2.3 Simbol-simbol Deployment Diagram	28
Tabel 2.4 Simbol-simbol Use Case Diagram.....	29
Tabel 2.5 Simbol-simbol Activity Diagram.....	31
Tabel 2.6 Simbol-simbol State Diagram.....	32
Tabel 2.7 Simbol-simbol Sequence Diagram	34
Tabel 2.8 Simbol-simbol Collaboration Diagram	38
Tabel 2.9 Simbol Entity Relationship Diagram	39
Tabel 3.1 Simbol Entity Relationship Diagram	40
Tabel 3.2 Data Guru SMK Muhammadiyah 2 Bandung	41
Tabel 3.3 Data Siswa Tahun Pelajaran 2019/2020.....	42
Tabel 3.4 Data Sarana dan Prasarana	43
Tabel 4.1 Kebutuhan fungsional perangkat lunak	51
Tabel 4.2 Deskripsi Aktor pada <i>Use Case Diagram</i>	52
Tabel 4.3 Deskripsi <i>Use Case</i> pada <i>Use Case Diagram</i>	53
Tabel 4.4 Deskripsi <i>Activity</i> pada <i>Activity Diagram</i>	56
Tabel 4.5 Deskripsi <i>Class</i> pada <i>Class Diagram</i>	59

Tabel 4.6 Deskripsi <i>Statechart</i> pada <i>Statechart Diagram</i>	61
Tabel 4.7 Deskripsi <i>Component</i> pada <i>Component Diagram</i>	68
Tabel 4.8 Deskripsi <i>Node</i> pada <i>Deployment Diagram</i>	71
Tabel 4.9 Struktur Table Admin	75
Tabel 4.10 Struktur Table Admin	76
Tabel 4.11 Struktur Table Admin	76
Tabel 4.12 Struktur Table Admin	77
Tabel 4.13 Struktur Table Admin	77
Tabel 4.14 Struktur Table Admin	78
Tabel 5.1 Pengujian Blackbox.....	95
Tabel 5.2 Ujian Halaman Login	95
Tabel 5.3 Uji Menu Administrator	96
Tabel 5.4 Uji Pengisian Data Alumni	96
Tabel 5.5 Uji Kelola Content (<i>insert, update, delete</i>).....	97
Tabel 5.6 Uji unduh file laporan.....	97

BAB 1

PENDAHULUAN

1.1. Latar Belakang Masalah

SMK Muhammadiyah 2 Kota Bandung merupakan sekolah menengah kejuruan yang terletak di Jl. Cilengkrang II No.7 Cibiru- Bandung. SMK Muhammadiyah 2 Kota Bandung SMK Muhammadiyah 2 Cibiru Kota Bandung merupakan salah satu SMK Swasta yang ada di Kota Bandung. Sekolah ini awalnya adalah sekolah ekonomi atas (SMEA) Muhammadiyah 2 Ujungberung yang kemudian dikembangkan menjadi SMK Muhammadiyah 2 Cibiru Kota Bandung berdasarkan peraturan yang dikeluarkan oleh Departemen Pendidikan Nasional tentang sekolah kejuruan [1].

Alumni di SMK Muhammadiyah 2 Bandung memiliki peranan yang sangat penting dalam perkembangan program keahlian Sekolah Menengah Kejuruan (SMK) itu sendiri. Pengertian alumni itu sendiri adalah orang – orang yang telah mengikuti atau tamat dari suatu sekolah atau perguruan tinggi (Tim Pustaka Phoniex, 2009: 39), dari tahun ke tahun jumlah alumni lulusan di SMK Muhammadiyah 2 Bandung semakin bertambah setiap tahunnya dari Tahun 2019 jumlah lulusannya mencapai 300 orang per tahun secara otomatis semakin tinggi juga angka kebutuhan untuk menjaringan alumninya dalam suatu wadah. Dengan database alumni penggunaan ruang penyimpanan data dapat dilakukan karena kita dapat melakukan penekanan jumlah data siswa alumni lulus. Serta informasi lainnya yang di rasa perlu untuk disajikan pada saat dibutuhkan seperti, memudahkan pencarian alumni lama atau baru. Pengembangan, pengurangan atau pencarian data akan lebih mudah [1].

Berdasarkan Hasil wawancara dengan Bapak Asep (Sebagai Ketua Jurusan Rekayasa Perangkat Lunak(RPL) SMK Muhammadiyah 2 Bandung pada tahun 2019 sekolah di SMK Muhammadiyah 2 Bandung terdiri dari berapa program keahlian Jurusan Pilihan Administrasi Perkantoran (AP), Rekayasa Perangkat Lunak (RPL), Teknik Komputer Jaringan (TKJ), Teknik Sepeda Motor (TSM), Perbankan Syariah (PBS). Sistem Informasi Pendataan Alumni SMK Muhammadiyah 2 Bandung masih dilakukan secara manual, yaitu dengan cara pengisian data pada buku besar. Pengisian data pada buku besar ini dilakukan pada saat alumni akan akan melegalisir ijazah, Setelah pengisian data selesai, selanjutnya petugas saat memindahkn satu persatu data tersebut di komputer dengan menggunakan *Miscrosoft Exel*. Hal ini memungkinkan terjadinya kekeliruan dalam penulisan data alumni pada buku besar, tidak mempermudah proses pencarian data alumni, dan data yang dicadangkan kurang terjamin kemungkinan untuk kehilangan data tersebut lebih besar, hasil dari wawancara dengan (Bapak Asep , 24 Juli 2019).

Dibutuhkan suatu sistem pendataan alumni yang mudah diakses oleh para alumninya, yang mampu memberikan berbagai informasi tentang data alumni SMK Muhammadiyah 2 Bandung yang sangat dibutuhkan oleh para alumni. Sehingga para alumni dapat melihat data alumni yang sudah diupload sebelumnya oleh admin.

Dari permasalahan di atas, maka untuk memenuhi suatu kebutuhan mengenai informasi data alumni dibutuhkan suatu pembangunan sistem informasi yang dapat mendukung semua informasi yang akan dihasilkan. Berdasarkan latar belakang masalah di atas maka penulis mengambil judul **“SISTEM INFORMASI PENDATAAN ALUMNI BERBASIS WEB DI SMK MUHAMMADIYAH 2 BANDUNG”**.

1.2. Rumusan Masalah

Berdasarkan latar belakang di atas, maka masalah yang akan dirumuskan dalam penelitian ini adalah:

- a. Bagaimana merancang dan membangun sistem informasi pendataan alumni berbasis *web* di SMK Muhammadiyah 2 Bandung ?
- b. Bagaimana menerapkan sistem informasi pendataan alumni berbasis *web* di SMK Muhammadiyah 2 Bandung ?

1.3. Batasan Masalah

Adapun batasan masalah dalam pelaksanaan laporan kerja praktek ini yaitu sebagai berikut:

1. Laporan berfokus pada pembangunan sisi *administrator* aplikasi yang akan dibuat.
2. Laporan yang di export / download berupa file *Exell*(xls).

1.4. Tujuan Kerja Praktik

Adapun tujuan penelitian yang ingin dicapai dalam pembuatan Kerja Praktik ini adalah:

- a. Membuat sistem informasi pendataan alumni berbasis *web* di SMK Muhammadiyah 2 Bandung.
- b. Menerapkan sistem informasi pendataan alumni berbasis *web* di SMK Muhammadiyah 2 Bandung.

1.5. Metode Kerja Praktik

Dalam menyelesaikan laporan kerja praktik ini menggunakan metode penelitian, diantaranya adalah:

a. Pengumpulan Data

1. Wawancara

Wawancara dilakukan dengan bertanya langsung kepada pihak sekolah

untuk mendapatkan data-data akurat yang dibutuhkan untuk pengembangan *website* alumni. Dalam proses wawancara dilakukan tanya jawab atau diskusi mengenai spesifikasi kebutuhan aplikasi yang akan dibuat. Melalui diskusi tersebut didapat kesimpulan bahwa aplikasi yang akan dibangun terdiri dari fitur yang dapat mengelola data alumni dan juga dapat memberikan informasi kepada alumni mengenai informasi terkini yang dibutuhkan oleh alumni, seperti informasi lowongan pekerjaan, informasi Keluarga, informasi mengenai kegiatan yang akan dilakukan atau dokumentasi mengenai kegiatan yang telah dilakukan oleh ikatan alumni dan laporan data siswa yang telah lulus.

2. Observasi

Observasi dilakukan dengan mengamati secara langsung cara kerja *website* sekolah yang belum ada, sistem pendataan alumni, dan bagaimana mempublikasikan informasi secara *online* yang dapat diakses melalui *internet*

3. Penelitian Kepustakaan

Penelitian kepustakaan merupakan pengumpulan data dengan cara mempelajari buku-buku literatur mengenai pembuatan *website* dengan PHP, CSS, *Javascript* serta mempelajari pendataan alumni yang biasa dilakukan dan situs-situs yang menyajikan ulasan mengenai pengembangan *website* yang membantu untuk memecahkan permasalahan yang dibahas.

b. Pengembangan Perangkat Lunak

Dalam mengembangkan *web* alumni ini pengembang menggunakan model

prototype. Model *prototype* digunakan untuk menspesifikasikan secara detail kebutuhan dari segi teknis dimana *user* kurang mengerti mengenai hal teknis tersebut.

Model *prototype* (*prototyping model*) dimulai dari mengumpulkan kebutuhan pelanggan terhadap perangkat lunak yang akan dibuat. (Shalahudin & Rosa A. S, 2013). *Prototype* menampilkan tampilan dari simulasi perangkat lunak yang belum jadi sehingga terlihat seperti perangkat lunak yang sudah jadi. *Prototype* akan dievaluasi oleh *user* apakah spesifikasi perangkat lunak sudah sesuai atau belum. Jika spesifikasinya sudah sesuai dengan keinginan *user* maka proses selanjutnya bisa dilakukan, tetapi jika spesifikasinya belum sesuai maka pengembang terus membuat spesifikasi perangkat lunak agar sesuai dengan keinginan *user*.

Berikut ini adalah tahapan dari pemodelan pengembangan perangkat lunak menggunakan model *prototype* (*prototyping model*), yaitu:

1. Pengumpulan kebutuhan

Pengembang dan pihak SMK Muhammadiyah 2 Bandung Bersama - sama mengidentifikasi, mendefinisikan, serta menspesifikasikan format *web* dan kebutuhan dari *web* yang akan dibuat.

2. Membangun *prototype*

Pengembang membuat *prototype* dengan merancang *mock-up web* yang akan dibuat.

3. Evaluasi *prototype*

Pihak SMK Muhammadiyah 2 Bandung mengevaluasi *prototype* yang sudah dibuat apakah sesuai dengan spesifikasi yang diinginkan

atau tidak. Jika spesifikasinya sudah sesuai maka pengembangan akan dilanjutkan ke tahap berikutnya, jika belum sesuai maka pengembang akan membangun *prototype* yang baru.

4. Mengkodekan aplikasi

Prototype yang sudah disepakati akan dikodekan ke dalam bahasa pemrograman yang sesuai (*coding*).

5. Menguji aplikasi

Setelah aplikasi jadi maka akan dilakukan pengujian terhadap aplikasi/*web* tersebut. Pengujian aplikasi/*web* yang sudah jadi dilakukan dengan menggunakan *black box*

6. Evaluasi aplikasi

Evaluasi aplikasi dilakukan oleh pihak SMK Muhammadiyah 2 Bandung. Evaluasi aplikasi ini untuk menguji apakah *web* yang dibangun sudah sesuai dengan yang diharapkan atau tidak. Jika sudah sesuai maka dilakukan tahap selanjutnya yaitu penggunaan aplikasi, jika tidak maka pengembangan akan kembali ke tahap pengkodean aplikasi dan pengujian aplikasi.

7. Menggunakan aplikasi

Web yang sudah dibuat siap untuk digunakan oleh pihak SMK Muhammadiyah 2 Bandung

1.6. Tempat dan Jadwal Kerja Praktik

Kerja Praktik dilaksanakan Sekolah Menengah Kejuruan Muhammadiyah 2 Cibiru Kabupaten Bandung yang beralamat lengkap di Jalan Cilengkrang II No 7

Kelurahan Palasari Kec. Cibiru, Telepon 021 7805502. Kerja Praktik dikerjakan dalam kurun waktu 1 bulan, yaitu mulai dari 26 Juni – 31 Juli 2019.

1.7. Sistematika Penulisan

Penyusunan laporan hasil kerja praktik ini terdiri dari beberapa bab dan masing-masing bab berisi uraian singkat dan penjelasan mengenai kegiatan selama kerja praktik.

Hal ini dimaksudkan agar pembahasan lebih sistematis dan spesifik sesuai dengan topik permasalahan yang dibahas. Laporan hasil kerja praktik ini terdiri dari 6 bab, diantaranya:

BAB I PENDAHULUAN

Bab ini berisi latar belakang masalah, rumusan masalah, tujuan kerja praktik, batasan masalah, metode kerja praktik, tempat dan jadwal kerja praktik, dan sistematika penulisan.

BAB II LANDASAN TEORI

Bab ini berisi tentang penjelasan mengenai teori-teori yang berhubungan dengan pembuatan halaman *website* serta teori-teori yang mendukung pengembangan *website* menggunakan PHP, *MySQL*, CSS, dan *Javascript*.

BAB III GAMBARAN UMUM INSTANSI

Bab ini berisi tentang uraian singkat mengenai latar belakang sekolah diantaranya profil sekolah, kegiatan sekolah yang meliputi aktivitas sekolah, visi dan misi sekolah, struktur organisasi sekolah, dan tugas pokok dan fungsi masing-masing seksi bagian.

BAB IV ANALISIS DAN PERANCANGAN SISTEM

Bab ini berisi mengenai modul dan komponen yang akan digunakan dan

juga menjelaskan proses perancangan dan pembuatan halaman *website*.

BAB V IMPLEMENTASI DAN PENGUJIAN SISTEM

Bab ini menjelaskan mengenai implementasi dan pengujian dari *website* alumni yang sudah dibuat.

BAB VI PENUTUP

Bab ini berisi kesimpulan dari topik yang telah dipaparkan penulis dalam laporan kerja praktik ini serta kritik dan saran yang mungkin diperlukan dalam pengembangan ilmu pengetahuan.

BAB II

LANDASAN TEORI

2. 1. Alumni

Menurut Kamus Besar Bahasa Indonesia (KBBI) versi *online*, alumni merupakan orang-orang yang telah mengikuti atau tamat dari suatu sekolah atau perguruan tinggi. Alumni merupakan bentuk jamak dari kata “alumnus”. Kata “alumni” menunjukkan banyak orang, sedangkan kata “alumnus” merupakan bentuk tunggal yang ditunjukkan untuk satu orang saja [2].

2.2. Sistem Informasi

Berdasarkan buku karangan (Abdul Kadir,2003) ada banyak pendapat tentang sistem informasi [3], yaitu sebagai berikut:

1. Menurut Alter (1992) sistem informasi adalah kombinasi antara prosedur kerja, informasi, orang, dan teknologi informasi yang diorganisasikan untuk mencapai tujuan dalam sebuah organisasi.
2. Menurut Wilkinson (1992) sistem informasi adalah kerangka kerja yang mengkoordinasikan sumber daya (manusia dan komputer) untuk mengubah masukan (input) menjadi keluaran (informasi), guna mencapai sasaran-sasaran perusahaan.
3. Menurut Hall (2001) sistem informasi adalah sebuah rangkaian prosedur formal dimana data dikelompokkan, diproses menjadi informasi dan didistribusikan kepada pemakai. Menurut penulis sistem informasi adalah suatu sistem kerja yang dibuat oleh manusia menggunakan komputer agar dapat memberikan informasi untuk mencapai tujuan yang diinginkan.

2.3. Administrator

Administrator merupakan bagian untuk mengelola isi dan tampilan *website* yang dilakukan oleh admin dimana halaman *administrator* bersifat rahasia, maka dari itu hanya admin yang dapat melihat dan mengelola. *Administrator* merupakan salah satu bagian penting dalam membangun sebuah perangkat lunak, jika tidak ada *administrator* semua content di halaman user tidak akan tampil. Perangkat lunak yang baik adalah perangkat lunak dengan *user interface* yang mudah dimengerti oleh pengguna dan bersahabat dengan penggunanya (*user friendly*). Selain itu, *administrator* juga harus memiliki kemampuan penggunaan (*usability*) yang baik. *Usability* diartikan sebagai kriteria dari kualitas suatu perangkat lunak seperti fungsionalitas, kemampuan, efisien, dan kemudahan dalam pemeliharannya [4].

2.4. Aplikasi

Menurut Ali Zaki & Smitdev Community, beliau berpendapat bahwa pengertian aplikasi adalah suatu komponen yang berguna melakukan pengolahan data maupun kegiatan-kegiatan seperti pembuatan dokumen atau pengolahan data. Aplikasi itu sendiri adalah bagian dari PC yang dapat berinteraksi secara langsung dengan user. Aplikasi yang berjalan di atas sebuah sistem operasi, sehingga aplikasi dapat diaktifkan, dan perlu untuk melakukan instalasi sistem operasi dahulu [5].

Pendapat lain dari Yuhefizar adalah aplikasi itu merupakan program yang dikembangkan untuk memenuhi kebutuhan pengguna dalam menjalankan pekerjaan tertentu [6].

2.5. Website

2.5.1 Pengertian Website

Web merupakan halaman situs berupa berkas teks (*plaintek*) yang diatur sedemikian rupa melalui instruksi berbasis HTML dan juga bahasa skrip lainnya yang saling terhubung satu sama lain. Halaman *web* berisi informasi baik berupa teks, gambar, *audio*, maupun video. Menurut (Pohan, 2010) *web* merupakan salah satu layanan yang didapat oleh pemakai komputer yang terhubung ke *internet*. Halaman *web* tersimpan dalam server *web* dan terhubung secara global dengan komputer *client* melalui jaringan *TCP/IP* sebagai protokol pertukaran paket data. Jaringan inilah yang kemudian dikenal dengan istilah *Internet* [7].

2.5.2 Sejarah Website

Sejarah terbentuknya *internet* dimulai pada tahun 1969. Pada saat itu, DARPA (*Defence Advance Research*) yang merupakan lembaga riset Departmen Pertahanan Amerika mendanai sebuah proyek untuk mengembangkan jaringan komunikasi data antar beberapa komputer. Tujuannya adalah untuk keperluan militer, yaitu untuk menghubungkan daerah-daerah penting agar bisa terhubung dalam menghadapi masalah jika terjadi serangan nuklir dan gangguan keamanan lainnya. Perkembangan jaringan tersebut menghasilkan *ARPNet*. Pada tahun 1972, organisasi ini mendemonstrasikan proyek ini di depan peserta *The First Internastional Conference on ComputerCommunication* dengan memasang 40 titik jaringan [7].

FTP (File Transfer Protokol) merupakan aplikasi *internet* pertama yang diciptakan, yang berfungsi untuk mentransfer atau mengirimkan file. Jumlah *host* di *internet* pada tahun 1984 lebih dari 1000 *host* dan pada tahun ini pula diperkenalkan *DNS (Domain Name System)* yang digunakan untuk mendefinisikan *IP (Internet Protocol)* agar lebih mudah untuk diingat [7].

Tahun 1986, U.S.NSF (*National Science Foundation*) yang merupakan Lembaga Ilmu Pengetahuan Nasional Amerika Serikat mendanai pembuatan jaringan *TCP/IP* yang didanai oleh organisasi *NSFNet*. Jaringan *TCP/IP* digunakan untuk menghubungkan universitas-universitas di Amerika Serikat dengan kecepatan transfer file sebesar 56 *Kbps*. Jaringan *TCP/IP* merupakan awal lahirnya perkembangan *internet* sampai saat ini [7].

Tahun 1987, lahirlah ISP (*Internet Service Provider*) yang diberi nama *UUNet*, yang merupakan provider utama *internet*. Tahun 1995, perkembangan *internet* semakin luas yang ditandai dengan adanya penyedia jasa *internet* (ISP). *NSFNet* yang sudah lama menjadi pusat *internet* kemudian diubah kembali menjadi jaringan untuk keperluan riset. Karena perubahan ini, maka lalu lintas data yang melalui Amerika dialihkan ke jaringan pusat ISP. Sementara itu *NSPNet* mengembangkan *VBNS (Very High Speed Backbone Network Service)*, yaitu jaringan dengan kecepatan sangat tinggi yaitu sebesar 622 *Mbps* (*OC-12*), yang menghubungkan lima komputer super. Seiring dengan berjalannya waktu, aplikasi *internet* semakin banyak dengan diciptakannya *WAIS (Wide Area Information Servers)*, *Gopher*, dan *WWW (World Wide Web)* [7].

2.5.3. World Wide Web

World Wide Web (WWW) atau yang biasa disebut *web* merupakan layanan penyajian informasi di *internet* dengan menggunakan HTML [8]. Cara kerja *www* dijelaskan sebagai berikut:

1. Informasi *web* disimpan dalam halaman *web (web pages)*. *Web pages* berisi *file-file* yang nantinya akan disimpan di komputer server (*web server*).
2. *Web server* mengirimkan *file-file* tersebut ke komputer *client*, sehingga komputer *client* dapat membaca *web pages*.

3. *Web client* dapat menampilkan halaman *web* melalui *web browser*.

2.5.4 Komponen Pembangun Website

Untuk membangun sebuah *website*, ada beberapa komponen yang harus diperhatikan [9], diantaranya:

1. HTML

HTML (*HyperText Markup Language*) merupakan bahasa pemrograman yang digunakan untuk pembuatan halaman *web*. Dokumen HTML dapat dijalankan diberbagai sistem operasi. HTML merupakan bahasa pemrograman yang tidak *case sensitive* yang artinya tidak ada perbedaan antara huruf kecil dengan huruf kapital. HTML terdiri dari tag-tag yang berisi elemen pembangun halaman *web*, seperti untuk menebalkan huruf maka digunakan tag `` atau ``.

2. PHP

PHP (*Hypertext Preprocessor*) merupakan bahasa pemrograman *server side* yang biasa digunakan oleh *developer* untuk membangun serta mengembangkan *website*. Untuk menjalankan PHP itu sendiri yaitu dengan menggunakan *web server*. Menurut Sidik, PHP merupakan bahasa pemrograman *script-script* yang membuat dokumen HTML secara *on the fly* yang dieksekusi di *server web*, dokumen HTML yang dihasilkan dari suatu aplikasi bukan dokumen HTML.

PHP yang diciptakan oleh Rasmus Lerdorf pada tahun 1994 yang awalnya singkatan dari *Personal Home Page* yang kemudian berubah menjadi *Hypertext Preprocessor*. Versi terbaru PHP saat ini yaitu versi ke 5 atau yang sering kita sebut PHP5.

3. CSS

CSS(*Cascading Style Sheet*) adalah bahasa *style sheet* untuk mengatur dokumen HTML seperti pengaturan *background*, *margin*, *font*, *border*, *teks*, gambar, atau yang lainnya sehingga membuat tampilan *web* menjadi lebih indah. Menurut Juju, CSS secara sederhana adalah sebuah metode yang digunakan untuk mempersingkat penulisan *tag* HTML, seperti *font*, *color*, *text*, dan tabel menjadi lebih ringkas sehingga tidak terjadi pengulangan penulisan (Juju, 2007). CSS bersifat lintas platform dimana *script* ini dapat dibaca oleh berbagai macam sistem operasi dan browser. CSS dikembangkan oleh W3C (*World Wide Web Consortium*), yaitu suatu organisasi yang mengembangkan teknologi *internet* pada tahun 1996.

4. Javascript

Javascript adalah bahasa pemrograman *web* berbasis java. Kode *javascript* disisipkan ke dalam halaman *web* melalui *tag-tag* yang berisi *script*. *Javascript* bersifat *case sensitive*, artinya ada perbedaan antara huruf kecil dengan huruf kapital. *Javascript* juga bersifat *client side scripting* yang eksekusinya berada di sisi *client* yang berupa *browser* seperti *Internet Explorer (IE)*, *Mozilla Firefox*, *Google Chrome*, *Metscape*, dan *Opera*.

5. JQuery

Jquery merupakan library yang berisi kumpulan kode-kode *javascript* yang berfungsi untuk mempermudah dan mempercepat dalam pembuatan kode *javascript*. Fitur yang terdapat di *jquery* diantaranya seperti fitur untuk memanipulasi *class* atau id pada CSS, efek animasi, dan yang lainnya

6. MySQL

MySQL (*My Structure Query Language*) adalah sebuah program pembuat dan pengelola *database* atau yang disebut dengan *DBMS (Database Management System)*. DBMS ini bersifat *open source*, yang artinya kita dapat memperolehnya secara gratis melalui alamat *website* <http://www.mysql.com>.

7. Apache

Apache merupakan sebuah perangkat lunak *web server* yang berfungsi untuk mengeksekusi program PHP serta melayani dan memfungsikan *website*. Untuk melayani fasilitas *web/www* pada *apache* menggunakan protokol HTTP. *Apache* dapat dijalankan di berbagai sistem operasi diantaranya *Linux*, *Windows* dan sistem operasi lainnya. Untuk menjalankan komponen *website* seperti PHP, *Apache* dan MySQL maka perlu menggunakan perangkat lunak XAMPP karena dengan menggunakan XAMPP kebutuhan paket-paket komponen *website* sudah disediakan oleh XAMPP. Pada buku Nugroho (2007), XAMPP merupakan paket PHP berbasis *open source* yang dikembangkan oleh sebuah komunitas *open source*.

2.6. Portal

Portal adalah aplikasi berbasis web yang menyediakan akses suatu titik tunggal dari informasi online terdistribusi, seperti dokumen yang didapat melalui pencarian, kanal berita, dan *link* ke situs khusus. Ada beberapa jenis portal, yang mendukung beberapa komunitas dan berbagai macam servis. Portal memiliki berbagai macam komponen teknik dan infrastruktur untuk mendukung fungsinya [10].

2.7. Internet

Internet merupakan sekumpulan komputer yang ada di dunia yang terhubung satu sama lain melalui protokol *TCP/IP* sehingga memungkinkan untuk bertukar

informasi dari komputer *server* ke komputer *client*. Menurut Juju, *internet* berasal dari kata *Interconnection Networking* yang mempunyai arti hubungan berbagai komputer dan berbagai tipe (*platform*) [11].

Komputer yang membentuk sistem jaringan yang mencakup seluruh dunia dengan melalui jalur telekomunikasi seperti telepon, *wireless*, bahkan satelit [11]. Menurut Sofana, *internet* adalah interkoneksi jaringan komputer skala besar (mirip WAN), yang dihubungkan menggunakan protokol khusus [12].

2.8. Model Pengembangan perangkat Lunak *Prototype*

Salah satu model pengembangan perangkat lunak adalah dengan model *Prototype*. Model Model *prototype* cocok digunakan untuk menggali spesifikasi kebutuhan pelanggan secara lebih detail tetapi beresiko tinggi terhadap membengkaknya biaya dan waktu proyek. [13].

Model *prototype* dimulai dengan pengumpulan kebutuhan perangkat lunak yang akan dibangun. Pengumpulan kebutuhan tersebut dapat dilakukan dengan berkomunikasi langsung antar tim pengembang dengan *client*. Komunikasi tersebut akan menghasilkan spesifikasi kebutuhan perangkat lunak yang akan dibangun. Spesifikasi kebutuhan tersebut akan diterjemahkan ke dalam rancangan *mock-up* perangkat lunak yang akan dibuat. Rancangan inilah yang merupakan konstruksi pembuatan *prototype*.

Prototype kemudian diserahkan kepada *client* untuk dievaluasi apakah sudah sesuai atau belum. Jika sudah sesuai dengan kebutuhan, maka proses pembuatan *software* akan dilanjutkan ke tahap selanjutnya, yaitu pengkodean sistem sesuai dengan *prototype* yang sudah dibangun. Jika belum sesuai dengan kebutuhan, maka proses pembuatan *software* akan kembali lagi ke tahap awal sampai didapat kesesuaian

kebutuhan yang diinginkan *client* [14].

Berikut ini adalah tahapan pengembangan perangkat lunak dengan model prototype, yaitu :

1. Pengumpulan Kebutuhan

Pada tahap ini, pengembang dan pelanggan bersama-sama mendefinisikan serta mengidentifikasi format dan semua kebutuhan perangkat lunak, beserta garis besar sistem yang akan dibuat.

2. Membangun *Prototyping*

Pembangunan prototype dengan membuat rancangan sementara yang berpusat pada penyajian kepada pelanggan (biasanya berupa *mock-up*).

3. Evaluasi *Prototyping*

Evaluasi *prototype* dilakukan oleh pelanggan dengan memeriksa apakah *prototype* yang dibangun sudah sesuai dengan kebutuhan atau belum. Jika sudah sesuai, maka tahap selanjutnya dapat dilakukan, sedangkan jika tidak maka kembali ke tahap awal.

4. Mengkodekan Sistem

Tahap ini merupakan tahap menerjemahkan *prototype* yang sudah dibangun ke dalam bahasa pemrograman yang sesuai.

5. Menguji Sistem

Setelah sistem sudah menjadi perangkat lunak yang siap pakai, maka akan dilakukan pengujian terlebih dahulu terhadap perangkat lunak tersebut. Pengujian sistem dilakukan dengan menggunakan *White Box*, *Black Box*, *Basis Path*, pengujian arsitektur, dan lain-lain.

6. Evaluasi Sistem

Evaluasi sistem dilakukan oleh pelanggan untuk memeriksa apakah sistem yang sudah jadi sesuai dengan yang diharapkan atau belum. Jika sudah maka akan

dilanjutkan ke tahap selanjutnya dan jika tidak maka akan mengulangi tahap 4 sampai 5.

7. Menggunakan Sistem

Perangkat lunak yang sudah diujian siap dipakai oleh pelanggan.

2.9. UML (*Unified Modeling Language*)

Unified Modeling Language (UML) digunakan untuk melakukan pemodelan sistem/perangkat lunak dengan menggunakan tools yang ada. Dengan pemodelan menggunakan UML, rekayasa dan pengembangan perangkat dapat dilakukan dengan fokus pengembangan dan desain perangkat lunak terhadap:

1. Tinjauan umum bagaimana arsitektur sistem secara keseluruhan
2. Penelaah bagaimana objek-objek dalam sistem saling mengirimkan pesan (message) dan saling bekerjasama satu sama lain
3. Menguji apakah sistem/perangkat lunak sudah berfungsi seperti yang seharusnya
4. Dokumentasi sistem/perangkat lunak untuk keperluan-keperluan tertentu di masa yang akan datang.

Adapun langkah-langkah penggunaan *Unified Modeling Language* (UML) diantaranya sebagai berikut:

1. Buatlah daftar *business process* dari *level* tertinggi untuk mendefinisikan aktivitas dan proses yang mungkin muncul.

2. Petakan *use case* untuk setiap *business process* untuk mendefinisikan dengan tepat fungsional yang harus disediakan oleh sistem, kemudian perhalus *use case diagram* dan lengkapi dengan *requirement*, *constraints* dan catatan-catatan lain.
3. Buatlah *deployment diagram* secara kasar untuk mendefinisikan arsitektur fisik sistem.
4. Definisikan *requirement* lain *non fungsional*, *security* dan sebagainya yang juga harus disediakan oleh sistem.
5. Berdasarkan *use case diagram*, mulailah membuat *activity diagram*.
6. Definisikan obyek-obyek level atas *package* atau *domain* dan buatlah *sequence* dan/atau *collaboration* untuk tiap alir pekerjaan, jika sebuah *use case* memiliki kemungkinan alir normal dan *error*, buat lagi satu diagram untuk masing-masing alir.
7. Buatlah rancangan *user interface model* yang menyediakan antarmuka bagi pengguna untuk menjalankan *skenario use case*.
8. Berdasarkan model-model yang sudah ada, buatlah *class diagram*. Setiap *package* atau *domain* dipecah menjadi *hirarki class* lengkap dengan *atribut* dan metodenya. Akan lebih baik jika untuk setiap *class* dibuat *unit test* untuk menguji *fungsionalitas class* dan interaksi dengan *class* lain.
9. Setelah *class diagram* dibuat, kita dapat melihat kemungkinan pengelompokkan *class* menjadi komponen-komponen karena itu buatlah *component diagram* pada tahap ini. Selain itu, definisikan *test integrasi* setiap komponen untuk meyakinkan ia dapat bereaksi dengan baik.

10. Perhalus *deployment diagram* yang sudah dibuat. Detailkan kemampuan dan *requirement* piranti lunak, sistem operasi, jaringan dan sebagainya. Petakan komponen ke dalam *node*.

11. Mulailah membangun sistem. Ada dua pendekatan yang tepat digunakan:

a. Pendekatan *use case* dengan *mengassign* setiap *use case* kepada tim pengembang tertentu untuk mengembangkan unit kode yang lengkap dengan *test*.

b. Pendekatan komponen yaitu *mengassign* setiap komponen kepada tim pengembang tertentu.

12. Lakukan **uji modul dan uji integrasi** serta perbaiki model beserta *codenya*. Model harus selalu sesuai dengan *code* yang aktual.

13. Perangkat lunak siap dirilis

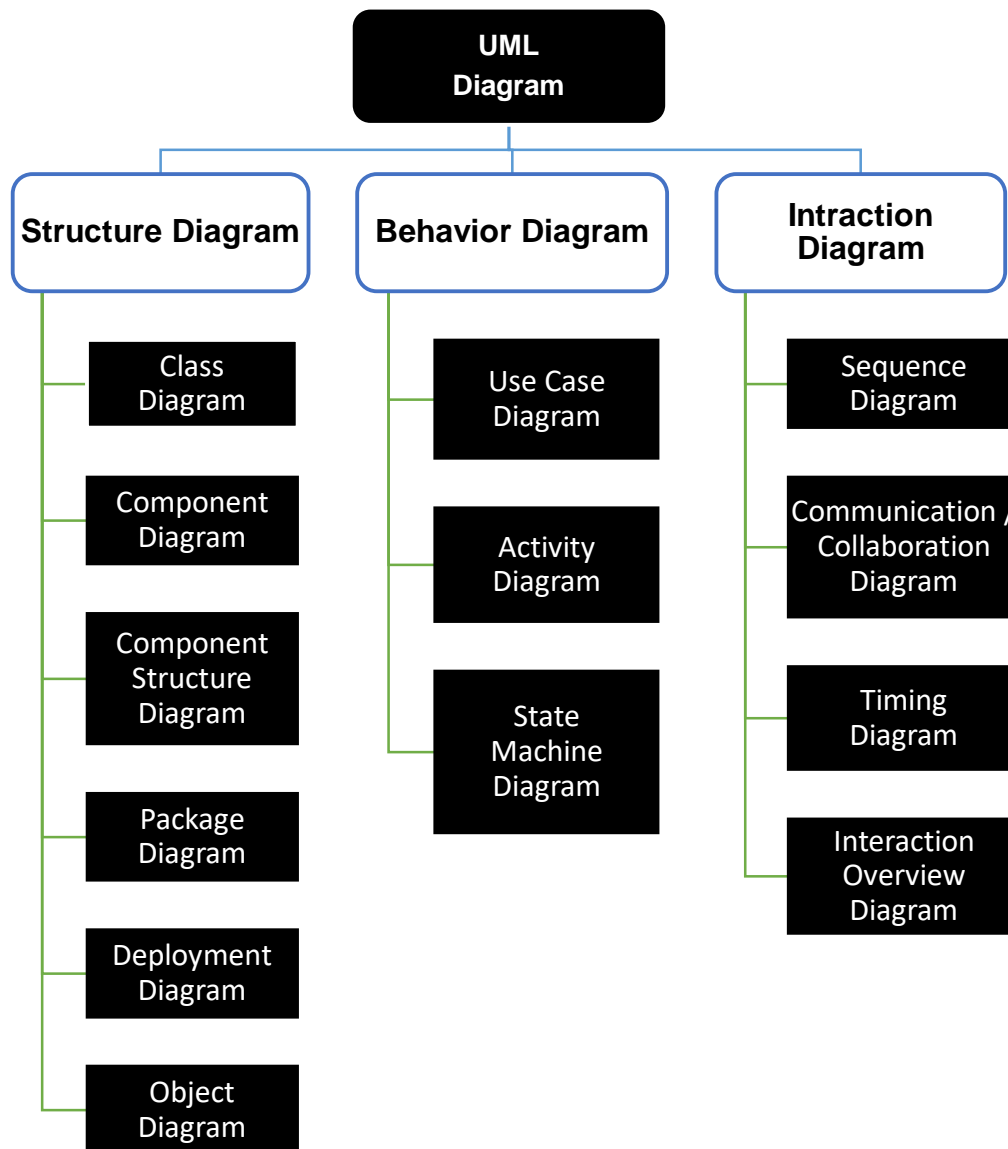
Selain itu ada Ruang Lingkup UML pada UML, dalam kerangka spesifikasi, *Unified Modeling Language (UML)* menyediakan model-model yang tepat [Adin05], tidak mendua arti (*ambigu*) serta lengkap. Secara khusus, *Unified Modeling Language (UML)* menspesifikasikan langkah-langkah penting dalam pengambilan keputusan analisis, perancangan serta implementasi dalam sistem yang sangat bernuansa perangkat lunak (*software intensive system*). Dalam hal ini, *Unified Modeling Language (UML)* bukanlah merupakan bahasa pemrograman tetapi model-model yang tercipta berhubungan langsung dengan berbagai macam bahasa pemrograman, sehingga adalah mungkin melakukan pemetaan (*mapping*) langsung dari model-model yang dibuat dengan *Unified Modeling Language (UML)* dengan bahasa-bahasa pemrograman berorientasi obyek, seperti *Java*, *Borland Delphi*, *Visual Basic*, *C++*, dan lain-lain.

Pemetaan (*mapping*) *Unified Modeling Language (UML)* bersifat dua arah yaitu :

1. Generasi kode bahasa pemrograman tertentu dari *Unified Modeling Language (UML)* *forward engineering*.
2. Generasi kode belum sesuai dengan kebutuhan dan harapan pengguna, pengembang dapat melakukan langkah balik bersifat *iterative* dari implementasi ke *Unified Modeling Language (UML)* hingga didapat sistem/peranti lunak yang sesuai dengan harapan pengguna dan pengembang.

A. Diagram UML (*Unified Modelling Language*)

Pada UML terdiri dari 13 macam diagram yang dikelompokkan dalam 3 kategori. Pembagian kategori dan macam-macam diagram tersebut dapat dilihat pada gambar dibawah.



Gambar 2.1
Diagram UML

Namun secara umum hanya 8 diagram (*class diagram, component diagram, deployment diagram, use case diagram, activity diagram, state diagram, sequence diagram, communication diagram*) saja yang biasanya digunakan oleh para pengembang.

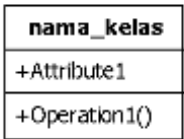


1. *Class Diagram*


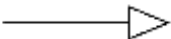


Diagram kelas atau *class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi. Atribut merupakan variabel-variabel yang dimiliki oleh suatu kelas. Sedangkan operasi atau metode adalah fungsi-fungsi yang dimiliki suatu kelas.

Diagram kelas dibuat agar pembuat program membuat kelas-kelas sesuai rancangan di dalam diagram kelas agar antara dokumentasi perancangan dan perangkat lunak sinkron. Banyak berbagai kasus, perancangan kelas yang dibuat tidak sesuai dengan kelas-kelas yang dibuat pada perangkat lunak, sehingga tidaklah ada gunanya lagi sebuah perancangan karena apa yang dirancang dan hasil jadinya tidak sesuai.

Berikut adalah simbol-simbol yang ada pada diagram kelas

Tabel 2.1
simbol-simbol *class diagram*

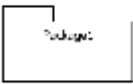
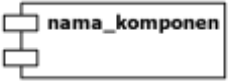
Simbol	Deskripsi
Kelas 	Kelas pada struktur sistem
Antarmuka / <i>interface</i> 	Sama dengan konsep <i>interface</i> dalam pemograman berorientasi objek
Asosiasi/ <i>association</i> 	Relasi antarkelas dengan makna umum, asosiasi biasanya juga disertai




	dengan <i>multiplicity</i>
Asosiasi berarah / <i>directed association</i> 	Relasi antarkelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga dengan <i>multiplicity</i>
Generalisasi 	Relasi antarkelas dengan makna generalisasi-spesialisasi (umum-khusus)
Kebergantungan / <i>dependency</i> 	Relasi antarkelas dengan makna kebergantungan antarkelas
Agregasi / <i>aggreation</i> 	Relasi antarkelas dengan makna semua –bagian (whole-part)

2. *Component Diagram*

Diagram komponen atau *component diagram* dibuat untuk menunjukkan organisasi dan ketergantungan diantara kumpulan komponen dalam sebuah sistem. Diagram komponen fokus pada komponen sistem yang dibutuhkan dan ada didalam sistem.

Tabel 2.2
simbol-simbol *component diagram*

Simbol	Deskripsi
<i>Package</i> 	<i>Package</i> merupakan sebuah bungkusan dari satu atau lebih komponen
Komponen 	Komponen sistem
Kebergantungan /	Kebergantungan antar komponen,

<i>dependency</i> 	arah panah mengarah pada komponen yang dipakai
Antarmuka / <i>interface</i> 	Sama dengan konsep <i>interface</i> pada pemrograman berorientasi objek, yaitu sebagai antarmuka komponen agar tidak mengakses langsung komponen
<i>Link</i> 	Relasi antar komponen


3. *Deployment Diagram*

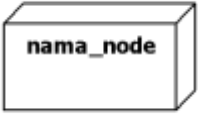


Deployment diagram menunjukkan konfigurasi komponen dalam proses eksekusi aplikasi. *Deployment diagram* juga dapat digunakan untuk memodelkan hal-hal berikut ini:

- Sistem tambahan yang menggambarkan rancangan *device*, *node*, dan *hardware*.
- Sistem terdistribusi murni.
- Rekaya ulang aplikasi.

Berikut adalah simbol-simbol yang ada pada *deployment diagram*:

Tabel 2.3
simbol-simbol *deployment diagram*

Simbol	Deskripsi
<i>Package</i> 	<i>Package</i> merupakan sebuah bungkusan dari satu atau lebih <i>node</i>
<i>Node</i>	Biasanya mengacu pada perangkat keras (<i>hardware</i>), perangkat lunak yang tidak dibuat sendiri (<i>software</i>),


	jika di dalam <i>node</i> disertakan komponen untuk mengkonsistensikan rancangan maka komponen yang diikutsertakan harus sesuai dengan komponen yang telah didefinisikan sebelumnya pada diagram komponen.
Kebergantungan / <i>dependency</i> 	Kebergantungan antar <i>node</i> , arah panah mengarah pada <i>node</i> yang dipakai
<i>Link</i> 	Relasi antar <i>node</i>





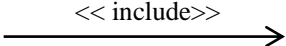
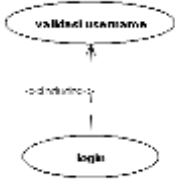
4. *Use Case Diagram*

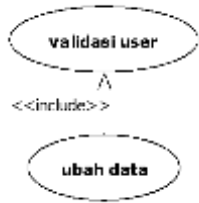
Use case diagram merupakan pemodelan untuk melakukan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu.

Berikut adalah simbol-simbol yang ada pada diagram *use case*:

Tabel 2.4
simbol-simbol *use case diagram*

Simbol	Deskripsi
<i>Use case</i> 	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja awal di awal frase nama <i>use case</i> .





<p>Aktor / <i>actor</i></p> 	<p>Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda di awal frase nama aktor.</p>
<p>Asosiasi / <i>association</i></p> 	<p>Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor</p>
<p>Generalisasi / <i>generalization</i></p> 	<p>Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya, misalnya: Arah panah mengarah pada <i>use case</i> yang menjadi generalisasinya (umum)</p> 
<p>Menggunakan / <i>include</i></p> 	<p>Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini.</p> <p>Ada dua sudut pandang yang cukup besar mengenai <i>include</i> di <i>use case</i>:</p> <ul style="list-style-type: none"> • <i>Include</i> berarti <i>use case</i> yang ditambahkan akan selalu dipanggil saat <i>use case</i> tambahan dijalankan, misal pada kasus berikut:  <ul style="list-style-type: none"> • <i>Include</i> berarti <i>use case</i> yang tambahan akan selalu melakukan


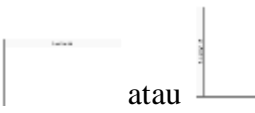
	<p>pengecekan apakah <i>use case</i> yang ditambahkan telah dijalankan sebelum <i>use case</i> tambahan dijalankan, misal pada kasus berikut :</p>  <pre> graph TD UC1([validasi user]) -.-> <<include>> UC2([ubah data]) </pre>
--	--

5. Activity Diagram

Diagram aktivitas atau *activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem

Tabel 2.5 simbol-simbol *activity diagram*

Simbol	Deskripsi
<p>status awal</p> 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
<p>Aktivitas</p> 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
<p>Percabangan / <i>decision</i></p> 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
<p>Penggabungan / <i>join</i></p> 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan



	menjadi satu.
Status akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir
Swimlane	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi. 



6. *State Diagram*

State diagram atau diagram status digunakan untuk menggambarkan perubahan status atau transaksi status dari sebuah mesin atau sistem atau objek. Jika diagram sekuen digunakan untuk interaksi antar objek maka diagram status digunakan untuk interaksi di dalam sebuah objek. Perubahan tersebut digambarkan dalam suatu graf berarah. *state diagram* cocok digunakan untuk menggambarkan alur interaksi pengguna dengan sistem.

Berikut ini komponen-komponen dasar yang ada dalam *state diagram*:

Tabel 2.6
simbol-simbol *state diagram*

Simbol	Deskripsi
<i>start</i> / status awal (<i>initial state</i>) 	<i>Start</i> atau <i>initial state</i> adalah <i>state</i> atau keadaan awal pada saat sistem mulai hidup.
<i>end</i> / status awal (<i>final state</i>) 	<i>end</i> atau <i>final state</i> adalah <i>state</i> atau keadaan akhir dari daur hidup suatu sistem.
<i>Event</i>	<i>Event</i> adalah kegiatan yang

	menyebabkan berubahnya status mesin.
<i>State</i> 	<i>State</i> atau status adalah keadaan sistem pada waktu tertentu. <i>State</i> dapat berubah jika ada <i>event</i> tertentu yang memicu perubahan tersebut.

7. *Sequence Diagram*


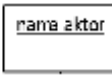

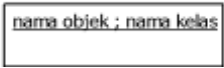


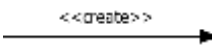
Diagram sekuen menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambar diagram sekuen maka harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu. Membuat diagram sekuen juga dibutuhkan untuk melihat skenario yang ada pada *use case*.

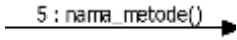

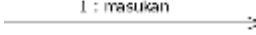


Banyaknya diagram sekuen yang harus digambar adalah minimal sebanyak pendefinisian *use case* yang memiliki proses sendiri atau yang penting semua *use case* yang telah didefinisikan interaksi jalannya pesan sudah dicakup pada diagram sekuen sehingga semakin banyak *use case* yang didefinisikan maka diagram sekuen yang harus dibuat juga semakin banyak.

Berikut adalah simbol-simbol yang ada pada diagram sekuen:

Tabel 2.7
simbol-simbol *sequence diagram*

Simbol	Deskripsi
aktor	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi

 <p>Atau</p>  <p>Tanpa waktu aktif</p>	<p>yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda di awal frase nama aktor.</p>
<p>Garis hidup / <i>lifetime</i></p> 	<p>Menyatakan kehidupan suatu objek</p>
<p>Objek</p> 	<p>Menyatakan objek yang berinteraksi pesan</p>
<p>Waktu aktif</p> 	<p>Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan di dalamnya, misalnya :</p>  <p>Maka cekstatuslogin() dan open() dilakukan di dalam metode login() aktor tidak memiliki waktu aktif</p>
<p>Pesan tipe <i>create</i></p> 	<p>Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat.</p>

<p>Pesan tipe <i>call</i></p> 	<p>Menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri.</p>  <p>Arah panah mengarah pada objek yang memiliki operasi/metode, karena ini memanggil operasi/metode maka operasi/metode yang dipanggil harus ada pada diagram kelas sesuai dengan kelas objek yang berinteraksi.</p>
<p>Pesan tipe <i>send</i></p> 	<p>Menyatakan bahwa suatu objek mengirimkan data/masukan/informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim.</p>
<p>Pesan tipe <i>return</i></p> 	<p>Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian.</p>
<p>Pesan tipe <i>destroy</i></p> 	<p>Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang diakhiri, sebaiknya jika ada <i>create</i> maka ada <i>destroy</i>.</p>

8. Collaboration/Communication Diagram

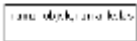

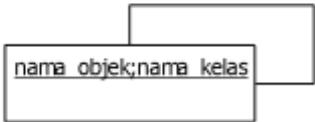
Communication diagram atau diagram komunikasi pada UML versi 2.x adalah penyederhanaan dari diagram kolaborasi (*collaboration diagram*) pada UML versi 1.x. *Collaboration diagram* sudah tidak muncul lagi pada UML versi 2.x. Diagram komunikasi sebenarnya adalah diagram kolaborasi tetapi dibuat untuk tiap sekuen.

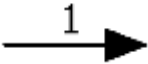
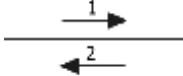
Diagram komunikasi menggambarkan interaksi antar objek/bagian dalam bentuk urutan pengiriman pesan. Diagram komunikasi merepresentasikan informasi yang diperoleh dari diagram kelas, diagram sekuen, dan diagram *use case* untuk mendeskripsikan gabungan antara struktur statis dan tingkah laku dinamis dari suatu sistem.

Diagram komunikasi mengelompokkan *message* pada kumpulan diagram sekuen menjadi sebuah diagram. Dalam diagram komunikasi yang dituliskan adalah operasi/metode yang dijalankan antara objek yang satu dan objek lainnya secara keseluruhan, oleh karena itu dapat diambil dari jalannya interaksi pada semua diagram sekuen. Penomoran metode dapat dilakukan berdasarkan urutan dijalankannya metode/operasi diantara objek yang satu dengan objek lainnya atau objek itu sendiri.

Berikut adalah simbol-simbol yang ada pada diagram komunikasi:

Tabel 2.8
simbol-simbol *collaboration diagram*

Simbol	Deskripsi
<p>objek</p> 	<p>Objek yang melakukan interaksi pesan</p>
<p><i>Link</i></p> 	<p>Relasi antar-objek yang menghubungkan objek satu dengan lainnya atau dengan dirinya sendiri</p> 

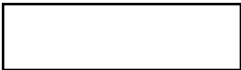

Arah pesan / stimulus 	Arah pesan yang terjadi, jika pada suatu <i>link</i> ada dua arah pesan yang berbeda maka arah juga digambarkan dua arah pada dua sisi <i>link</i> 
--	--

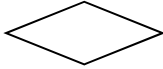

2.10 ERD (*Entity Relationship Diagram*)

ERD merupakan pemodelan pada basis data yang menghubungkan tabel atau data dalam *database*. ERD memodelkan struktur data dan hubungan antar data. Untuk menggambarkan ERD menggunakan beberapa notasi dan simbol. ERD dikembangkan berdasarkan teori himpunan dalam bidang matematika. ERD digunakan untuk pemodelan basis data relasional [15]. Simbol yang ada dalam ERD dijelaskan dalam tabel di bawah ini.

Tabel 2.2 merupakan tabel simbol *Entity Relationship Diagram*. Simbol dari ERD yaitu terdiri dari Entitas, Atribut, Relasi dan Asosiasi.

Tabel 2.9. Simbol Entity Relationship Diagram

No.	Nama	Simbol	Keterangan
1	Entitas		Entitas menunjukan data inti yang akan disimpan sebagai tabel pada <i>database</i> .
2	Atribut		<i>Field</i> atau kolom data yang disimpan dalam entitas.

3	Relasi		Relasi untuk menghubungkan antar entitas yang penamaannya dengan diawali kata kerja.
4	Asosiasi		Asosiasi merupakan penghubung antara entitas dengan relasi dimana di kedua ujungnyamemilikie beragaman kemungkinan jumlah pemakaian.

2.11 Kamus Data

Kamus data merupakan katalog yang berisi mengenai fakta tentang data dan kebutuhan suatu sistem. Kamus data adalah kumpulan daftar elemen data yang mengalir pada sistem perangkat lunak sehingga masukan (*input*) dan keluaran (*output*) [16]. Kamus data dibuat berdasarkan aliran data pada *Unified Modeling Language* (UML) digunakan untuk melakukan pemodelan sistem/perangkat lunak dengan menggunakan tools yang ada. Melalui kamus data, analisis sistem dapat mendefinisikan aliran data pada sistem dengan lengkap. Pada tahap analisis sistem, kamus data digunakan sebagai alat komunikasi antara *system analist* dengan *user* atau pemakai sistem mengenai aliran data dalam sistem, yaitu tentang data yang masuk ke sistem dan tentang informasi yang dihasilkan untuk kebutuhan pemakai sistem. Pada tahap perancangan sistem, kamus data digunakan untuk merancang inputan, merancang laporan dan *database*.

2.12. Struktur Database

Struktur *database* adalah pengaturan *field-field* tabel pada *database*. Struktur *database* diantaranya:

1. Nama *field*, digunakan sebagai identitas atau keterangan pada *field*.
2. *Type data*, digunakan untuk pemberian *type data* pada *field* sesuai dengan identitas yang telah ditentukan.
3. Ukuran data, digunakan untuk pemberian panjang atau batas banyaknya *bit* yang diinputkan ke *database*.
4. Keterangan, digunakan untuk pemberian deskripsi pada *field*.

2.12.1 Perancangan Database

Perancangan *database* terdiri dari 6 tahap, yaitu:

1. Pengumpulan data dan analisa

- a. Menentukan pengguna beserta bidang aplikasinya
- b. Meninjau dokumentasi yang sudah ada
- c. Analisis lingkungan operasi dan proses data
- d. Daftar pertanyaan dan wawancara

2. Perancangan *database* secara konseptual

- a. Perancangan skema yang konseptual
- b. Perancangan relasi yang akan terjadi dalam *database*

4. Pemilihan DBMS

a. Faktor teknis

Yang termasuk dalam faktor teknis diantaranya seperti tipe model data, struktur penyimpanan dan jalur pengaksesan yang didukung oleh *DBMS* (*Database Management System*), tipe *interface* dan *programmer*, serta tipe *query*.

b. Faktor ekonomi dan politik organisasi

- Faktor ekonomi meliputi biaya penyediaan *hardware* dan *software*, biaya pembuatan *database*, serta biaya SDM.
- Faktor organisasi contohnya seperti *database* penyimpanan buku di perpustakaan.

5. Perancangan *database* secara *logic* (*data model mapping*)

a. Pemetaan (transformasi data)

Pada tahap ini, transformasi data tidak mempertimbangkan karakteristik yang akan diaplikasikan pada sistem manajemen *database* dengan spesifik atau khusus.

b. Penyesuaian skema ke *DBMS*

Skema *database* dihasilkan dari tahap pemetaan data. Pemetaan dikonfirmasi ke dalam bentuk implementasi yang lebih spesifik dari model data yang digunakan oleh sistem manajemen *database* yang dipilih

6. Perancangan *database* secara fisik

a. *Response time*

Response time merupakan waktu transaksi *database* yang dibutuhkan selama eksekusi data untuk menerima respon.

b. *Space utility*

Space utility merupakan jumlah ruang penyimpanan yang digunakan oleh sistem *database* dan struktur jalur akses *database*.

7. Implementasi sistem *database*

Implementasi sistem *database* merupakan pelaksanaan pembuatan sistem *database* sesuai dengan perancangan yang sebelumnya sudah dibuat.

Implementasi *database* dilakukan oleh seorang *programmer database*.

2.13. Model Pengujian Perangkat Lunak Black Box

Pengujian perangkat lunak *black box* merupakan pengujian terhadap fungsionalitas aplikasi yang bertentangan dengan struktur internal. *Black box* fokus pada pengujian terhadap kebutuhan fungsional sistem. Pengujian dibangun di sekitar spesifikasi dan persyaratan, seperti aplikasi seperti apa yang seharusnya dilakukan.

Pengujian *black box* diterapkan pada semua tingkat pengujian perangkat lunak, seperti unit, integrasi, fungsional, sistem, dan penerimaan [17]. Pengujian *black box* mencoba untuk menemukan kesalahan dalam beberapa kategori seperti fungsi-fungsi yang salah atau hilang, kesalahan pada *interface*, kesalahan pada struktur data atau akses *database* eksternal kesalahan pada performa, serta kesalahan inisialisasi dan terminasi.

BAB III
TINJAUAN UMUM
SMK MUHAMMADIYAH 2 BANDUNG

3.1. Profil Sekolah Smk muhammadiyah

SMK Muhammadiyah 2 Kota Bandung merupakan sekolah menengah kejuruan yang terletak di Jl. Cilengkrang II No.7 Cibiru- Bandung. SMK Muhammadiyah 2 Kota Bandung SMK Muhammadiyah 2 Cibiru Kota Bandung merupakan salah satu SMK Swasta yang ada di Kota Bandung. Sekolah ini awalnya adalah sekolah ekonomi atas (SMEA) Muhammadiyah 2 Ujungberung yang kemudian dikembangkan menjadi SMK Muhammadiyah 2 Cibiru Kota Bandung berdasarkan peraturan yang dikeluarkan oleh Departemen Pendidikan Nasional tentang sekolah kejuruan. Sekolah ini mulai beroperasi sejak tahun 1968, tetapi baru terdaftar pada Dinas Pendidikan Kota Bandung pada tahun 1973 dengan izin operasional berdasarkan surat keputusan No.48/S/1973 tanggal 3 April 1973.

SMK Muhammadiyah 2 Cibiru Kota Bandung berada dibawah pengelolaan Yayasan Majelis Pendidikan Dasar dan Menengah (Dikdasmen) Pimpinan Cabang Muhammadiyah Ujungberung yang kemudian beralih ke PCM Cibiru karena pemekaran wilayah Kota Bandung.

SMK ini berada dalam satu kompleks dengan SMA Muhammadiyah 4. Lokasi tersebut berada di wilayah Kecamatan Cibiru yang berbatasan dengan Kecamatan Ujungberung di sebelah barat, Kecamatan Cileunyi di sebelah timur, Kecamatan Panyileukan di sebelah Selatan dan Kecamatan Cilengkrang di sebelah utara.

Kecamatan Cibiru berada dikawasan Bandung bagian timur yang nota bene merupakan kawasan prioritas pengembangan pembangunan Kota Bandung sekarang ini.

Dengan demikian SMK ini memiliki prospek yang cukup baik, ditambah lagi dengan program pemerintah yang ingin meningkatkan minat masyarakat terhadap SMK dengan tujuan menciptakan kemampuan keterampilan masyarakat sehingga siap untuk memasuki dunia kerja. SMK Muhammadiyah 2 Bandung menyelenggarakan program studi Bisnis dan Manajemen dengan jurusan yang dibuka adalah Administrasi Perkantoran (dulu bernama sekretaris), Rekayasa Perangkat Lunak, Teknik Sepeda Motor dan Perbankan Syariah.

3.2. Visi dan Misi SMK MUHAMMADIYAH 2 BANDUNG

3.2.1. Visi

Menyiapkan tenaga kerja terdidik, terampil, mandiri, beriman dan berakhlak mulia

3.2.2. Misi

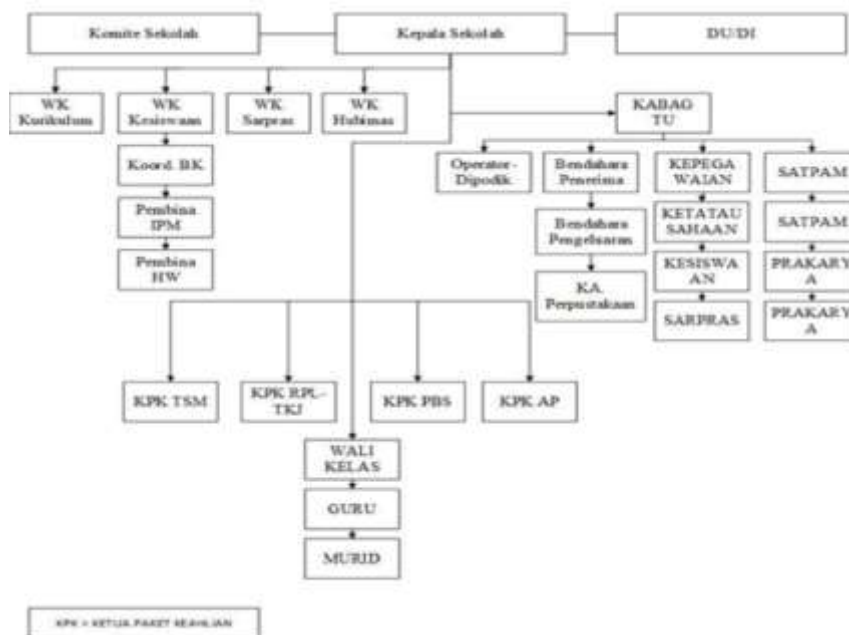
1. Mewujudkan lingkungan yang menjunjung tinggi nilai – nilai ketertiban, kedisiplin, didasarkan atas nilai – nilai keislaman.
2. Menumbuhkan kembangkan budaya semangat belajar sebagai upaya mewujudkan manusia terampil dan produktif sesuai tuntutan dunia usaha dan industri
3. Menumbuhkan semangat keunggulan dan peningkatan sumber daya insani sesuai dengan lajunya IPTEK dan IMTAK.
4. Membekali peserta didik keterampilan dasar berbahasa Inggris sebagai bekal untuk bekal untuk bekerja di dunia internasional.
5. Mewujudkan lingkungan fisik yang BERIMAN
(Bersih, Rindang, Indah, Aman, dan Nyaman).

6. Menumbuhkan pemahaman, penghayatan, dan pengamalan ajaran islam secara benar sebagai dasar dan sumber terbentuknya akhlak yang mulia.

3.2.3 Tujuan

1. Mempersiapkan peserta didik agar menjadi manusia produktif, mampu bekerja mandiri, mengisi lowongan pekerjaan yang ada di DU/DI sebagai tenaga kerja tingkat menengah, sesuai dengan kompetensi dalam program keahlian pilihannya
2. Membekali peserta didik agar mampu memilih karir, ulet dan gigih dalam berkompetisi, beradaptasi di lingkungan kerja dan mengembangkan sikap profesional dalam bidang keahlian yang diminatinya
3. Membekali peserta didik dengan ilmu pengetahuan, teknologi dan seni agar mampu mengembangkan diri dikemudian hari baik secara mandiri maupun melalui jenjang pendidikan yang lebih tinggi

3.3. Struktur Organisasi



Gambar 3.1: Struktur Organisasi SMKM 2 BANDUNG

3.4. Kondisi Umum

1. Keadaan Guru

Tabel 3.1 : Data Guru SMK Muhammadiyah 2 Bandung

Ijazah Tertinggi	Jumlah			
	Guru DPK	Guru Tetap	GTT	Guru Bantu
S2/S3	3	3	-	-
S1	6	35	15	-
D3	1	2	2	-
D1/D2/SLTA	-	-	-	-
Jumlah	10	40	17	-

2. Keadaan Tenaga Adminitrasi /TU

Tabel 3.2 : Data Tenaga Administrasi/TU

Ijazah Tertinggi		
S2/S3	-	-
S1	2	-
D3	1	-
D1/D2/SLTA	6	1
Jumlah	9	1

3. Keadaan siswa Tahun Pelajaran 2019/2020

Tabel 3.2 :Data Siswa Tahun Pelajaran 2019/2020

Program Keahlian	Kelas	Laki-laki	Perempuan	Jumlah
Administrasi Perkantoran	I	23	104	127
	II	32	126	158
	III	37	140	177
Rekayasa Perangkat Lunak	I	11	34	45
	II	36	18	54
	III	53	11	64
Teknik Kerja Jaringan	I	45	13	58
	II	33	5	38
	III	39	11	50
Teknik Sepeda Motor	I	39	-	39
	II	43	-	43
	III	36	-	36
Perbankan Syariah	I	-	33	33
	II	3	22	25
	II	-	28	28
Jumlah		430	545	975

4. Keadaan Sarana dan Prasarana

Tabel 3.3 :Data Sarana dan Prasarana

No.	Jenis	Jumlah	Keterangan
1	Ruang Belajar/Kelas	26	Baik
2	Ruang Kepala Madrasah/Ruang PKM	1	Baik
3	Ruang Tata Usaha	1	Baik
4	Ruang Guru	1	Baik
5	Ruang Perpustakaan	1	Baik
6	Ruang Laboratorium IPA	1	Baik
7	Ruang Komputer/Multi Media	1	Baik
8	BP	1	Baik
9	Ruang UKS	1	Baik
10	Ruang Koperasi dan Kantin	1	Baik
11	Ruang IPM	1	Baik
12	Gudang	1	Baik
13	WC Guru	2	Baik
14	WC Siswa	10	Baik
15	Musholla	1	Baik
16	Lapang Olah Raga/Upacara	1	Baik
17	Tempat paker	1	Baik
18	Bengkel/Lab TSM	1	Baik
19	Laboratorium Komputer	1	Baik
20	Mini Bank	1	Baik
21	Pos Satpam	1	Baik
22	Ruang Bendahara	1	Baik
23	Komputer	31	Baik
24	Printer	2	Baik
25	Sound Portable	2	Baik
26	Audio	1	Baik
27	Brankas	1	Baik
28	Lemari/F. Kabinet	2	Baik
29	Meja TU	3	Baik
30	Kursi TU	3	Baik
31	Meja Guru	23	Baik
32	Kursi Guru	23	Baik
33	Meja Siswa	> 200	Baik
34	Kursi Siswa	> 200	Baik
35	Stand Mic	1	Baik
36	Papan Tulis	19	Baik
37	LCD	1	Baik

4.1. Program Pembinaan Dan Pengembangan Peserta Didik

1. Kegiatan Pembelajaran

Kegiatan pembelajaran di SMK Muhammadiyah 2 Kota Bandung dilaksanakan pada hari Senin sampai Sabtu dan di bagi menjadi dua sif, yaitu sif pertama dilaksanakan mulai pukul 07:00 sampai dengan 12:00 WIB, dan sif kedua dilaksanakan mulai pukul 13:00 sampai dengan 18:00 WIB. Pembagian dua sif ini didasar karenaa tidak berbanding nya jumlah rombongan belajr dengan jumlah ruang kelas. Kelas X dan XII masuk pad sif pertama dan Kelas XII masuk pad sif kedua.

2. Kegiatan Keagamaan

Kegiatan keagamaan di SMK Muhammadiyah 2 Cibiru Kota Bandung telah dilaksanakan dan terus ditingkatkan dengan cara melaksanakan Shalat Dhuha berjamaah jam 07.00 WIB, setelah melaksanakan shalat dhuha perwakilan siswa akan menyampaikan ceramah selama 7 menit (Kultum). Selain itu, pembiasaan berdoa dan membaca surat-surat pendek dalam Al-Quran pada waktu sebelum dan sesudah belajar, juga sekolah mengadakan pengajian rutin setiap malam rabu yang membahas mengenai amalan *fiqih muhammadiyah*. Setiap seminggu sekali siswa dan guru mengadakan pengajian keliling di rumah siswa dan kegiatan tersebut dilakukan secara bergiliran, waktunya biasanya siang hari.

Selain itu, SMK Muhammadiyah 2 Kota Bandung menyelenggarakan program BTQ. Latar belakang diadakannya BTQ adalah banyak siswa siswi SMK Muhammadiyah yang belum bisa baca tulis Al-Quran sehingga sekolah membuat kegiatan BTQ. Pelaksanaan pembinaan BTQ dilaksanakan menggunakan pola diniyah di sekolah ada jam pembelajaran intrakulikuler PAI, yakni dua jam pelajaran 2 x 45 menit tiap kelas. Tempat yang digunakan

adalah ruang kelas dan musholla sekolah. kegiatan BTQ ini adalah membaca, menulis dan menghafal Al-Quran. Tujuan diadakannya kegiatan BTQ ini adalah untuk mengetahui tingkat kemajuan siswa dalam membaca, menulis dan menghafal Al-Quran. Pada kegiatan BTQ ini diharapkan siswa bisa menghafal doa-doa shalat, *juz amma* dan doa-doa yang sering di amalkan di kehidupan sehari-hari. Namun di kegiatan BTQ ini masih ada kekurangan-kekurangan yaitu kurangnya guru yang melatih kegiatan tersebut dan kurangnya minat siswa dalam belajar BTQ

3. Kegiatan Intrakulikuler

Program kegiatan intrakulikuler siswa yang ada di SMK Muhammadiyah 2 Kota Bandung adalah Ikatan Pelajar Muhammadiyah (IPM). Program kegiatan ini merupakan kegiatan semacam Organisasi Siswa Intra Sekolah (OSIS) di tingkat sekolah negeri, tetapi jika di sekolah Muhammadiyah, IPM ini lah sebagai Organisasi Siswa Intra Sekolah nya.

Adanya IPM di sekolah Muhammadiyah adalah untuk mewadahi kreativitas, pikiran, masukan, kritikan, serta membawahi berbagai ekstrakulikuler yang ada di sekolah.

1. Kegiatan Ekstrakuler

a. Hizbul Wathon

Kegiatan ekstrakurikuler Kepanduan Hizbul Wathan SMK Muhammadiyah 2 Cibiru Bandung telah dilaksanakan dan terus ditingkatkan dengan cara memberdayakan tenaga guru dan juga melibatkan alumni sekolah yang sudah mahir dalam kegiatan hizbul wathan, sehingga sampai dengan saat ini ekstra kurikuler Kepramukaan terus berjalan.

Kegiatan hizbul wathan dilaksanakan setiap hari sabtu siang tepatnya jam 13.00-15.00. peserta hizbul wathan adalah kelas X dari semua jurusan yang ada di SMK Muhammadiyah. Di kepanduan hizbul wathan ini ada organisasinya, kepengurusan hizbul wathan di pegang oleh kelas XI. Di awal tahun pelajaran, kepanduan hizbul wathan mengadakan kegiatan PERGARI yang bertujuan untuk memperkenalkan kepanduan hizbul wathan smk muhammadiyah pada siswa baru (kelas X). kegiatan ini melibatkan seluruh tenaga pendidik dan kependidikan serta siswa kelas X dan XI.

b. Keputrian

Kegiatan Keputrian merujuk pada salah satu kegiatan ekstrakurikuler yang dilaksanakan dan dikhususkan untuk para siswi (perempuan). Kegiatan ini bertujuan untuk memberdayakan para siswi dan menyalurkan bakat dan minat mereka yang berkaitan dengan pekerjaan domestik perempuan seperti belajaran merajut, merangkai bunga, memasak dan belajar menjahit. Namun kegiatan keputri tidak berjalan dengan lancar karena tidak adanya pendanaan yang cukup untuk melancarkan kegiatan ini. Hendaknya pihak sekolah harus mendukung penuh kegiatan tersebut khususnya dalam urusan pendanaan karena tidak mungkin semua pembiayaan dalam pelaksanaan kegiatan keputrian ditanggung oleh siswi. Selain itu, kegiatan program ini hendaknya diatur sedemikian rupa oleh pembina agar terjadwal dan terorganisir, sehingga dalam setiap pertemuan para siswi bisa berkarya, meskipun sederhana tetapi kaya manfaat

c. Lingkungan Hidup

Kegiatan ekstrakurikuler lingkungan hidup dilaksanakan setiap hari sabtu dan kegiatannya seperti kegiatan pembelajaran di kelas. Kegiatan ini bertujuan agar siswa siswi SMK Muhammadiyah 2 cibiru memiliki wawasan, peduli dan berbudaya lingkungan. Dengan adanya kegiatan ini diharapkan siswa siswi.

d. Kesenian dan olahraga

Kegiatan ekstra kurikuler Kesenian di SMK Muhammadiyah 2 BANDUNG dilaksanakan secara temporer, misalnya pada waktu akan melaksanakan pentas seni di akhir tahun pelajaran, Acara Akreditasi ataupun ada acara perlombaan. Pelatihan tari tidak dilaksanakan di sekolah melainkan di sanggar seni dikarenakan tidak adanya tempat yang memadai.

e. PASKIBRA

Kegiatan Paskibra SMK Muhammadiyah 2 BANDUNG masih dibawah naungan kepanduan Hizbul Wathan. Kegiatan paskibra ini dilaksanakan setiap hari sabtu siang. Kegiatan ini sudah mendapatkan beberapa penghargaan. Pada kegiatan ini didominasi oleh para perempuan, jumlah laki-lakinya sangat minim.

f. Tapak Suci

Kegiatan ekstrakurikuler tapak suci dilaksanakan pada hari senin jam 13.00 WIB. Kegiatan ini adalah kegiatan yang diwajibkan oleh sekolah SMK Muhammadiyah. Kegiatan tapak suci sudah mendapatkan beberapa penghargaan ketika mengikuti perlombaan.

BAB IV

ANALISI DAN PERACANGAN SISTEM

4.1 Analisis Sistem

Analisis sistem ini merupakan penguraian dari suatu aplikasi bantu dengan menggunakan objek yang utuh ke dalam bagian – bagian komponennya dengan maksud untuk mengidentifikasi dan mengevaluasi permasalahan – permasalahan yang ada.

Tahap analisis ini merupakan tahap untuk mendefinisikan kebutuhan – kebutuhan dan yang dibutuhkan untuk membuat suatu aplikasi bantu sistem informasi pendataan alumni SMK Muhammadiyah 2 Bandung berbasis web.

4.1.1 Analisis Sistem yang Sedang Berjalan

1. Deskripsi Masalah

Media yang ada sebelumnya di SMK Muhammadiyah 2 Bandung adalah *website* sekolah yang hanya memberikan informasi umum mengenai sekolah tersebut. Belum adanya sistem yang berfokus pada pemberian informasi mengenai alumni. Selain itu juga minimnya pengontrolan terhadap pendataan alumni yang masih menggunakan sistem manual. Dengan begitu muncul permasalahan dalam pengelolaan data alumni yang kurang efektif dan juga kurangnya informasi dari pihak sekolah yang diterima alumni.

1. Pemecahan Masalah

Melalui pendeskripsian permasalahan diatas, maka solusi untuk pemecahan masalah yang akan dilakukan adalah sebagai berikut:

- a. Sekolah harus memiliki sistem yang dapat membantu dalam pengelolaan data

alumni.

- b. Sekolah harus memiliki sistem tersendiri yang dapat memberikan informasi khusus yang berhubungan dengan alumni.

2. Fitur yang Diharapkan

Fitur-fitur yang diharapkan dari pembuatan *website* alumni ini adalah:

- a. Fitur untuk *input* data alumni oleh admin.
- b. Fitur untuk melihat informasi mengenai hal-hal yang dibutuhkan oleh alumni, seperti jumlah data alumni per program keahlian kejuruan SMK, pencarian alumnus berdasarkan nama, tahun masuk, tahun kelulusan, angkatan, dan kota tempat alumni
- c. Fitur untuk unduh *file* yang dibutuhkan alumni yang baru bisa dilakukan melalui admin.
- d. Fitur untuk *menginputkan* oleh admin berkenaan dengan informasi almuni seperti informasi dasar, keluarga, Orangtua, kontak, data terkini, akademik, *download file*
- e. Fitur untuk melihat/unduh file laporan oleh *admin*

4.1.2. Sistem yang Diharapkan

Setelah menguraikan beberapa pemecahan masalah maka diharapkan sistem yang akan dibangun akan dapat memenuhi kebutuhan *user*, diantaranya:

- a. *Web* ini menjadi salah satu media yang dapat membantu pengelolaan data alumni.
- b. *Web* ini menjadi salah satu media bagi alumni untuk mendapatkan informasi yang dibutuhkan para alumni.
- c. *Web* ini dapat membantu alumni untuk mendapatkan *file-file* yang

dibutuhkan oleh para alumni maupun pihak sekolah.

4.2. Analisis Kebutuhan

4.2.1. Kebutuhan Perangkat Keras (*Hardware*)

Perangkat keras yang dibutuhkan dalam pembuatan aplikasi portal alumni ini adalah:

1. Minimal *processor Intel Pentium*
2. Minimal *Harddisk* 20 GB
3. Minimal RAM 515 MB
4. *Monitor, Keyboard, Mouse* sebagai peralatan antarmuka.

4.2.2. Kebutuhan Perangkat Lunak (*Software*)

Perangkat lunak yang dibutuhkan dalam pembuatan aplikasi portal alumni ini

adalah:

1. *Browser Internet*, seperti *Google Chrome, Mozilla Firefox*, atau *Opera*
2. *Software editor*, seperti *notepad++, Sublime Text 2*.
3. Aplikasi *Web Server*, seperti *XAMPP v1.7.3*
4. Perangkat lunak pemodelan sistem, seperti *Power Designer v16.1.0*
5. Perangkat lunak pemodelan *user interface*, seperti *Balsamiq Mockups*

4.2.3. Kebutuhan Fungsional

Kebutuhan yang harus ada dalam *Administrator* aplikasi Sistem Informasi Pendataan Alumni SMK Muhammadiyah 2 Bandung terdiri dari fitur, komponen, serta fungsi. Berikut ini spesifikasi kebutuhan fungsional *Administrator diagram* Alumni SMK Muhammadiyah 2 Bandung di jelaskan pada tabel 4.1 di bawah ini

Tabel 4.1 Kebutuhan fungsional perangkat lunak

Kode	Fungsi	Deskripsi
Req01	Login	Merupakan proses untuk melakukan <i>login</i> admin <i>website</i> Alumni
Req02	Pengisian data alumni	Proses pendataan alumni
Req03	Kelola <i>Content</i>	Merupakan proses untuk <i>insert</i> , <i>update</i> , <i>delete</i>
Req04	Unduh <i>File</i> Laporan	Proses untuk mengunduh file laporan yang siap dicetak

4.3. Perancangan Sistem

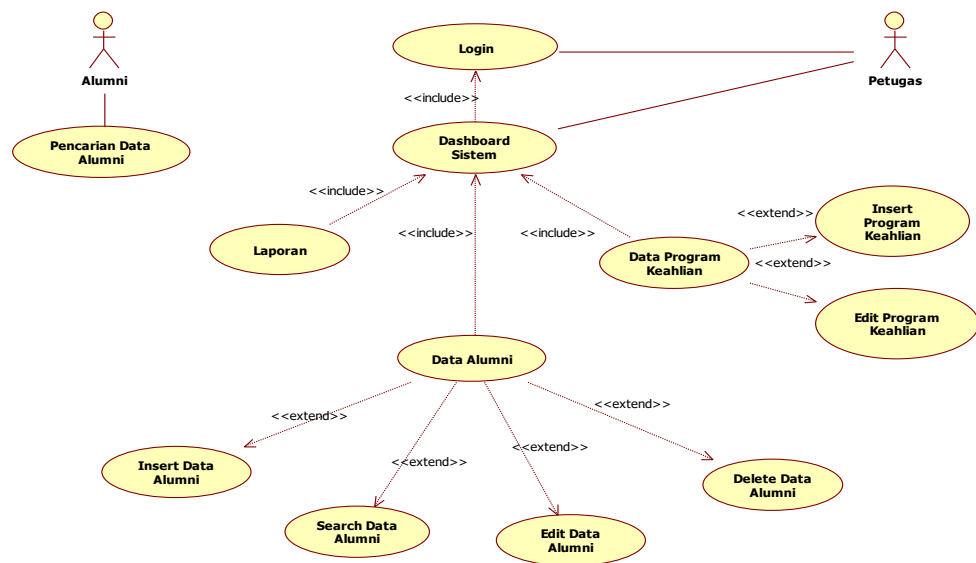
Hasil analisis terhadap sistem yang telah ada pada SMK Muhammadiyah 2 Bandung, maka harus ditambahkan sistem informasi yang berfokus pada pengelolaan informasi alumni. Dengan begitu, maka diusulkan beberapa pendekatan terhadap sistem yang akan dibangun dalam bentuk diagram agar mendapatkan data yang akurat dan dapat memudahkan dalam pembacaan aplikasi.

4.4. Rancangan Fungsional (*Functional Design*)

Rancangan yang dibuat penulis menggunakan metode *Unified Modelling Language* (UML) adalah bahasa pemodelan untuk sistem atau perangkat lunak dengan pendekatan berorientasi objek. Tahapannya adalah sebagai berikut:

1. *Use Case Diagram*

Use case diagram merupakan pemodelan untuk melakukan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu.



Gambar 4.1 Use Case Diagram

a. Deskripsi Aktor

Berikut ini adalah deskripsi aktor pada *use case diagram* Sistem informasi Alumni SMK Muhammadiyah 2 Bandung yang diusulkan oleh penulis.

No	Aktor	Deskripsi
1.	Alumni	Lulusan dari SMK Muhammadiyah 2 Cibiru yang memiliki hak akses untuk melihat dan mencari data para alumni tanpa perlu masuk ke <i>dashboard</i> .
2.	Petugas	User yang bertugas untuk mengolah data para alumni baik menambahkan, merubah maupun data yang ada di sistem. Serta, melakukan pencetakan/perekapan laporan yang telah ada di sistem.

Tabel 4.2 Deskripsi Aktor pada Use Case Diagram

b. Deskripsi Use Case

Berikut ini adalah deskripsi *use case* pada *use case diagram* Sistem informasi Alumni SMK Muhammadiyah 2 Cibiru Bandung yang diusulkan oleh penulis.

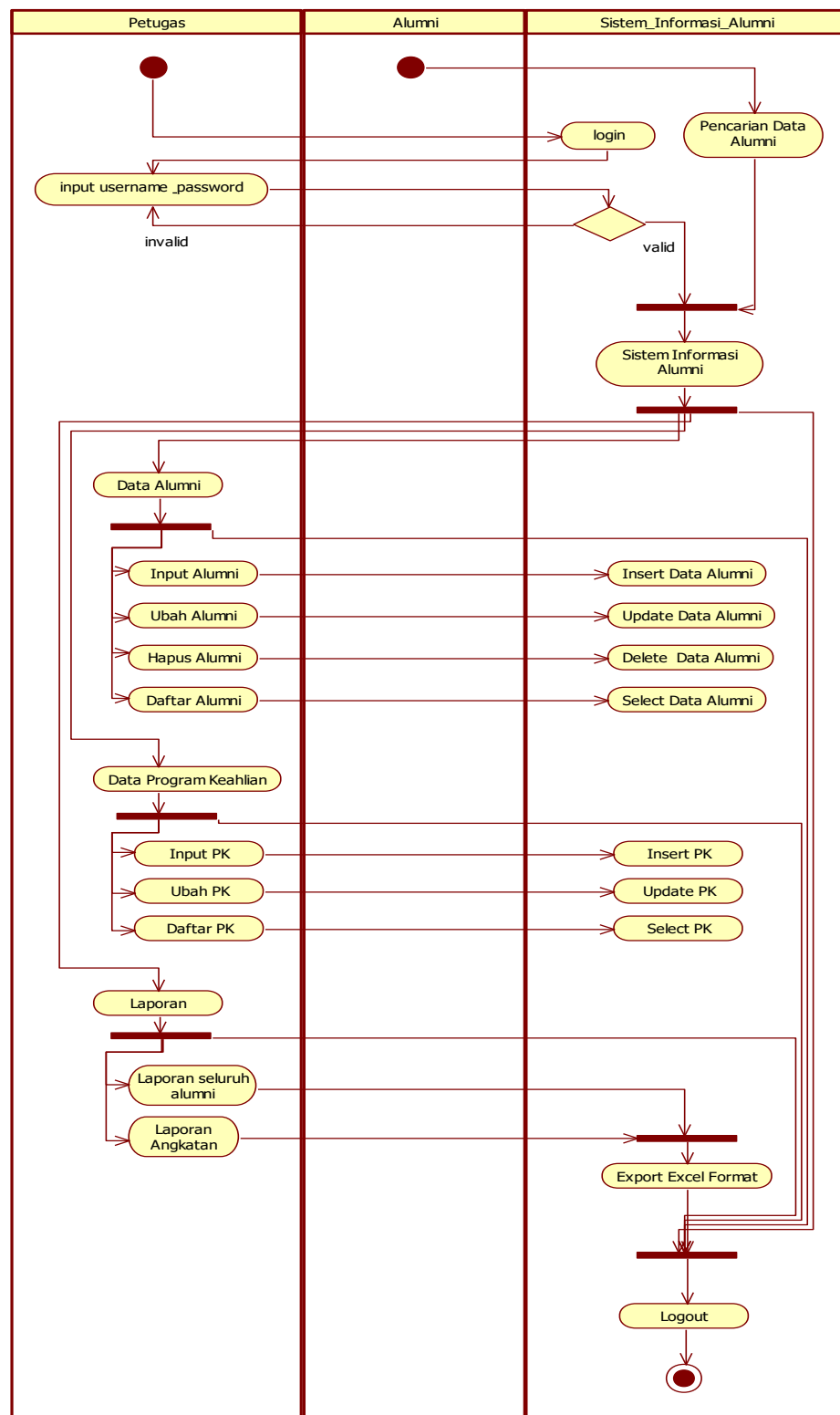
Tabel 4.3 Deskripsi *Use Case* pada *Use Case Diagram*

No	<i>Use Case</i>	Deskripsi
1.	Pencarian data alumni	Alumni dapat mencari dan melihat data-data terkait alumni yang telah lulus.
2.	<i>Login</i>	Petugas melakukan login ke sistem. Sistem akan melakukan validasi login. Apabila diterima oleh sistem maka akan masuk ke halaman utaman. Jika tidak, maka akan muncul pesan untuk login kembali.
3.	<i>Insert data</i> alumni	Petugas melakukan input data ke sistem terkait dengan data para alumni seperti identitas, kegiatan saat ini dan sebagainya.
4.	<i>Insert data</i> program keahlian	Petugas melakukan input data ke sistem terkait dengan data program keahlian seperti kode prorgam keahlian, deskripsi dan sebagainya.
5.	<i>Edit data</i> alumni	Petugas melakukan input data ke sistem terhadap data-data alumni yang akan diperbaharui sesuai dengan kondisi terbaru.
6.	<i>Edit data</i> program keahlian	Petugas melakukan input data ke sistem terhadap data-data program keahlian yang akan diperbaharui sesuai dengan kondisi terbaru.
7.	<i>Delete data</i> alumni	Petugas menghapus data alumni sesuai dengan kondisi tertentu.
8.	<i>Search data</i> alumni	Petugas dapat mencari data-data alumni sesuai dengan filter yang telah ada di dalam sistem.
9.	Laporan	Petugas dapat mencetak data-data alumni sebagai

		laporan dengan dibuatkan format excel oleh sistem.
--	--	--

2. *Activity Diagram*

Diagram aktivitas atau *activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak.



Gambar 4.2
Use Case Diagram

Berikut ini adalah deskripsi *activity* pada *activity diagram* Sistem Informasi Alumni SMK Muhammadiyah 2 Bandung yang diusulkan oleh penulis.

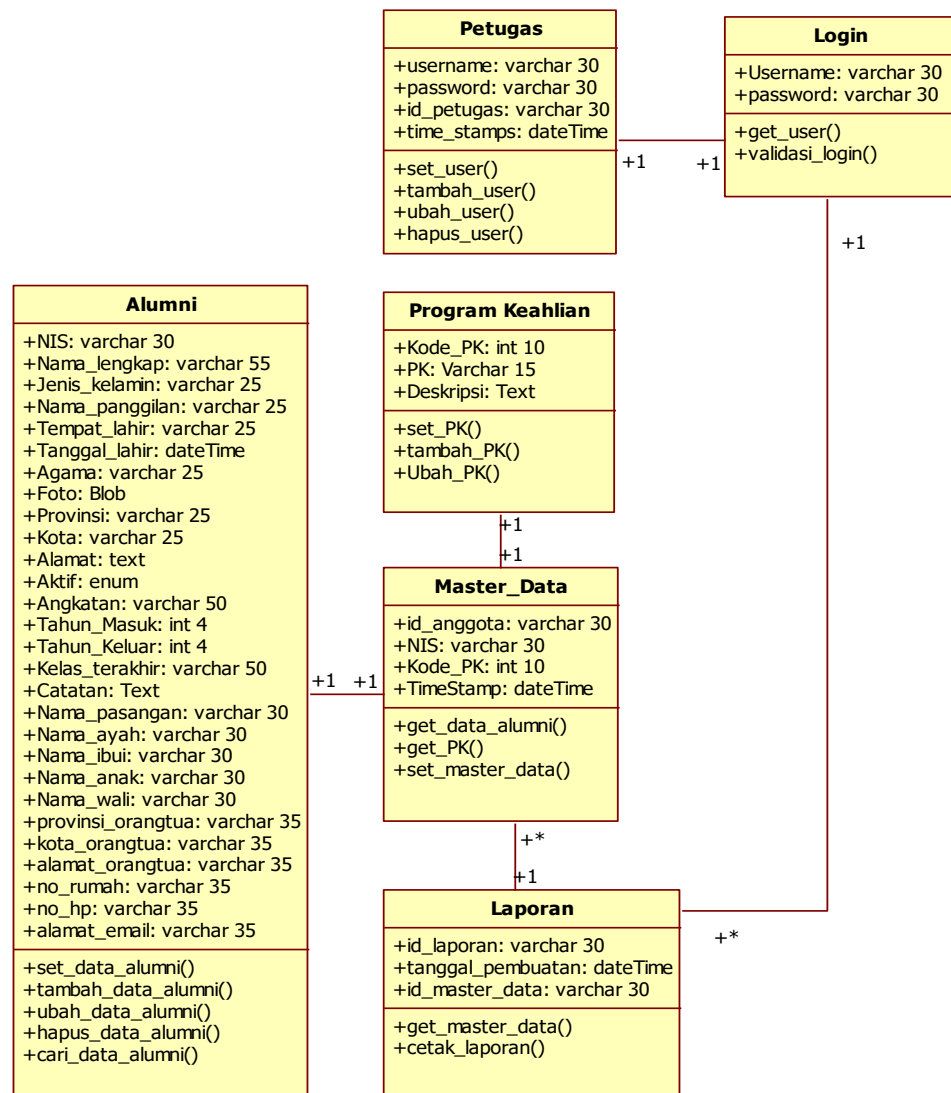
Tabel 4.4 Deskripsi *Activity* pada *Activity Diagram*

No	Petugas	Alumni	Sistem
1.	Petugas melakukan login dengan memasukkan <i>username</i> dan <i>password</i> .	Alumni melakukan pencarian data-data alumni	
2.			Kemudian sistem akan melakukan validasi <i>login</i> terhadap inputan. Apabila valid, sistem akan mengarahkan admin, operator, atau pengguna ke <i>dashboard</i> . Apabila tidak valid maka akan diminta untuk memasukan kembali <i>username</i> atau <i>password</i> yang sesuai.
3.	Jika petugas memilih data Alumni		Maka sistem akan menampilkan data yang terkait dengan alumni dan memberikan hak akses untuk dapat merubah atau menghapus sesuai dengan kebutuhan.
4.	Jika petugas memilih data Program keahlian		Maka sistem akan menampilkan data yang terkait dengan program keahlian dan dapat merubah sesuai dengan kebutuhan.
5.	Jika petugas memilih laporan		Maka sistem akan menampilkan filter untuk membuat laporan dan

			akan membuat laporan berdasarkan filter ke dalam format excel.
6.			Sistem akan melakukan <i>logout</i> jika petugas telah selesai menyelesaikan urusannya dan memilih untuk <i>logout</i> .

3. *Class Diagram*

Diagram kelas atau *class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi. Atribut merupakan variabel-variabel yang dimiliki oleh suatu kelas. Sedangkan operasi atau metode adalah fungsi-fungsi yang dimiliki suatu kelas.



Gambar 4.3 Class Diagram

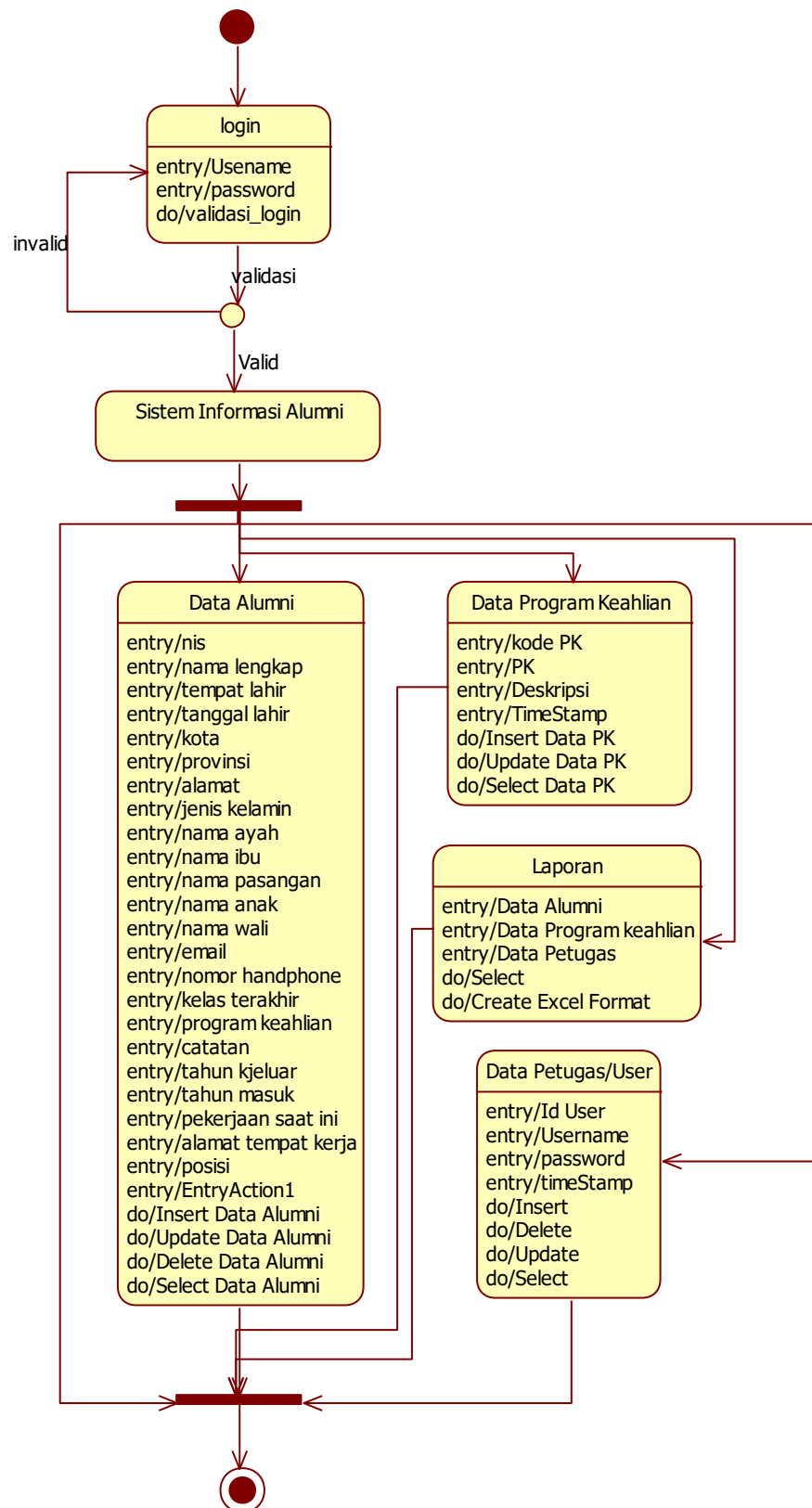
Berikut ini adalah deskripsi *Class* pada *Class Diagram* Sistem informasi Alumni SMK Muhammadiyah 2 Bandung yang diusulkan oleh penulis.

Tabel 4.5 Deskripsi Class pada Class Diagram

No	Class	Deskripsi
1	<i>login</i>	Kelas proses yang menggambarkan struktur dan deskripsi pada proses <i>login</i> untuk memverifikasi pengguna yang dapat mempunyai akses dalam pengolahan sistem.
2	Petugas	Kelas data yang menggambarkan struktur dan deskripsi objek pada pengguna sistem.
3	Alumni	Kelas data yang menggambarkan struktur dan deskripsi Alumni pada pengguna sistem.
4	Program keahlian	Kelas data yang menggambarkan struktur dan deskripsi Program Keahlian pada pengguna sistem.
5	Master Data	Kelas proses yang menggambarkan struktur dan deskripsi pada proses pengolahan data alumni untuk memudahkan pengguna dalam mengolah data.
6	Laporan	Kelas data yang menggambarkan struktur dan deskripsi pada laporan yang ada di dalam sistem.

4. Statechart Diagram

State diagram atau diagram status digunakan untuk menggambarkan perubahan status atau transaksi status dari sebuah mesin atau sistem atau objek. Jika diagram sekuen digunakan untuk interaksi antar objek maka diagram status digunakan untuk interaksi di dalam sebuah objek. Perubahan tersebut digambarkan dalam suatu graf berarah. *state diagram* cocok digunakan untuk menggambarkan alur interaksi pengguna dengan sistem.



Gambar 4.4 Statechart Diagram

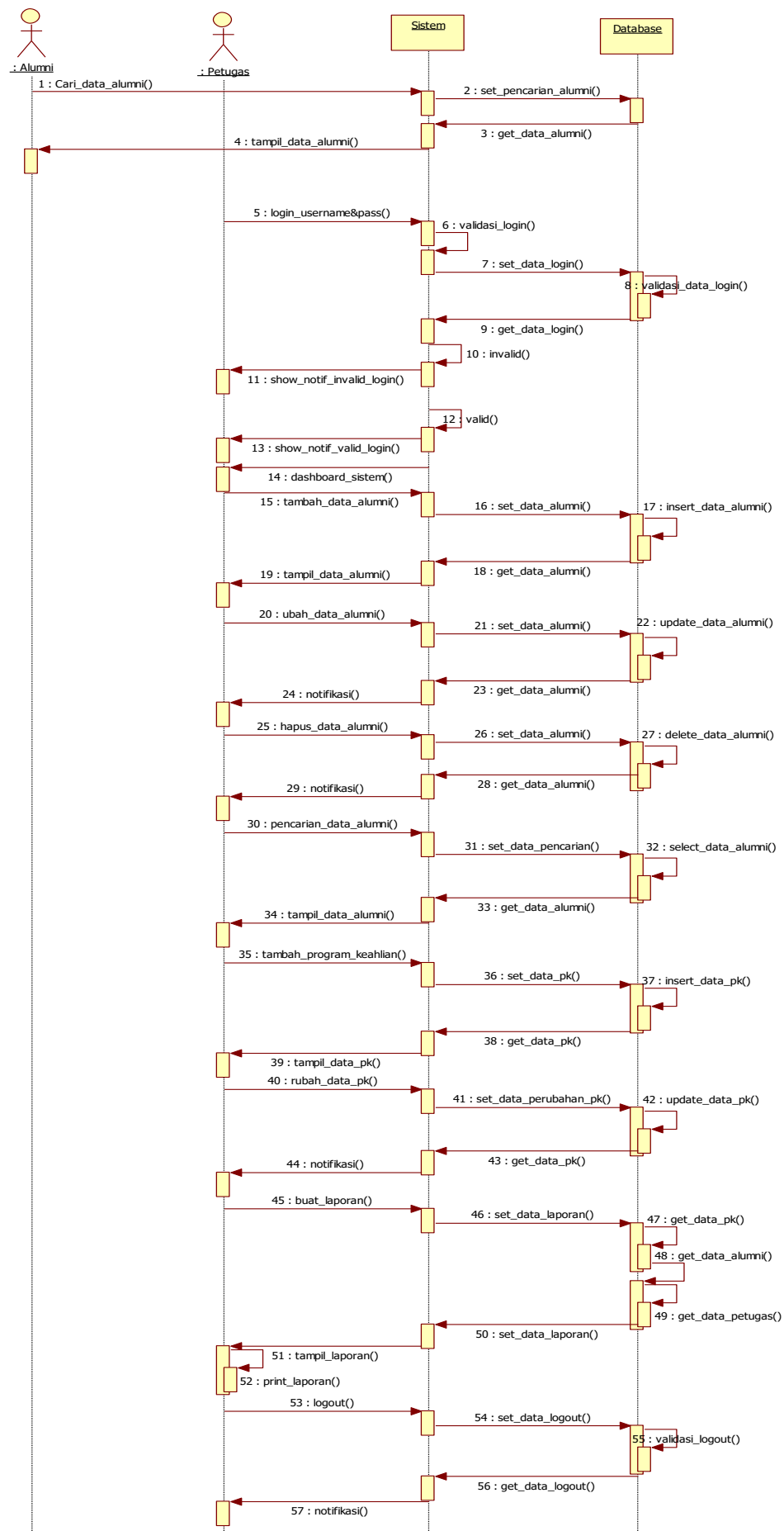
Berikut ini adalah deskripsi *Statechart* pada *Class Diagram* Sistem Informasi Pendataan Alumni SMK Muhammadiyah 2 Bandung yang diusulkan oleh penulis.

Tabel 4.6 Deskripsi *Statechart* pada *Statechart Diagram*

No	State	Deskripsi
1	<i>login</i>	Pada <i>state</i> ini petugas melakukan <i>login</i> dengan memasukan <i>username</i> dan <i>password</i> . Sistem melakukan validasi dan akan masuk ke halaman utama sistem informasi pelayanan administrasi kependudukan.
2	Sistem Informasi Pelayanan Alumni	Pada <i>state</i> ini menampilkan halaman utama pada <i>website</i> sistem informasi Alumni.
3	Data Alumni	Pada <i>state</i> ini, petugas dapat melakukan proses CRUD (<i>Create, Read, Delete, Update</i>) terhadap data alumni sesuai dengan kebutuhan.
4	Data Program keahlian	Pada <i>state</i> ini, petugas dapat melakukan proses CRUD (<i>Create, Read, Delete, Update</i>) terhadap data program keahlian sesuai dengan kebutuhan.
5	Data Petugas	Pada <i>state</i> ini, petugas dapat melakukan proses CRUD (<i>Create, Read, Delete, Update</i>) terhadap data petugas sesuai dengan kebutuhan.
6	Laporan	Pada <i>state</i> ini, petugas dapat mencetak rekpa data alumni yang ada dengan memilih filter laporan dan sistem akan membuat laporan dalam format Excel.

5. *Sequence Diagram*

Diagram sekuen menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambar diagram sekuen maka harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu. Membuat diagram sekuen juga dibutuhkan untuk melihat skenario yang ada pada *use case*.



Gambar 4.5 *Sequence Diagram*

Berikut ini adalah deskripsi *Sequence* pada *Sequence Diagram* Sistem informasi Pendataan Alumni SMK Muhammadiyah 2 Bandung yang diusulkan oleh penulis.

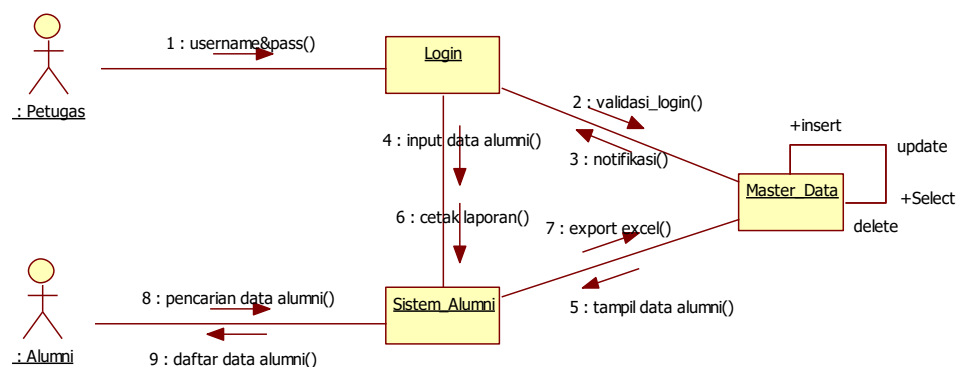
1. Alumni dapat melakukan proses pencarian data seluruh alumni sesuai dengan *filter* yang dipilih. Lalu, sistem akan menampilkan data alumni dari database sesuai dengan *filter* masukan.
2. Petugas melakukan *login* pada sistem.
3. Sistem akan mengecek dan mencocokkan data *username* dan *password* yang diinputkan dengan tabel user. Jika sesuai maka pengguna dapat mengakses sistem informasi pelayanan administrasi kependudukan.
4. Jika *invalid*, maka sistem akan mengarah ke tahap awal. Jika *valid*, maka sistem akan mengarahkan ke tahap selanjutnya.
5. Petugas dapat melakukan *Input*, *Update*, *Delete*, dan *Select* terhadap data alumni kemudian sistem akan melakukan koneksi dan eksekusi terhadap database sesuai dengan perintah CRUD yang di berikan oleh petugas.
6. Petugas dapat melakukan *Input*, *Update*, *Delete*, dan *Select* terhadap data program keahlian kemudian sistem akan melakukan koneksi dan eksekusi terhadap database sesuai dengan perintah CRUD yang di berikan oleh petugas.
7. Petugas dapat melakukan *Input*, *Update*, *Delete*, dan *Select* terhadap data alumni kemudian sistem akan melakukan koneksi dan eksekusi terhadap database sesuai dengan perintah CRUD yang di berikan oleh petugas.

8. Petugas dapat mencetak data alumni yang ada dengan format excel yang dibuat oleh sistem secara otomatis sesuai dengan filter laporan. Kemudian, petugas dapat mengunduh file tersebut untuk dijadikan rekap data maupun dicetak.
9. Petugas melakukan *logout* pada sistem jika telah selesai melakukan tugasnya.

6. Collaboration Diagram

Communication diagram atau diagram komunikasi pada UML versi 2.x adalah penyederhanaan dari diagram kolaborasi (*collaboration diagram*) pada UML versi 1.x. *Collaboration diagram* sudah tidak muncul lagi pada UML versi 2.x. Diagram komunikasi sebenarnya adalah diagram kolaborasi tetapi dibuat untuk tiap sekuen.

Diagram komunikasi menggambarkan interaksi antar objek/bagian dalam bentuk urutan pengiriman pesan. Diagram komunikasi merepresentasikan informasi yang diperoleh dari diagram kelas, diagram sekuen, dan diagram *use case* untuk mendeskripsikan gabungan antara struktur statis dan tingkah laku dinamis dari suatu sistem.



Gambar 4.6 ***Collaboration Diagram***

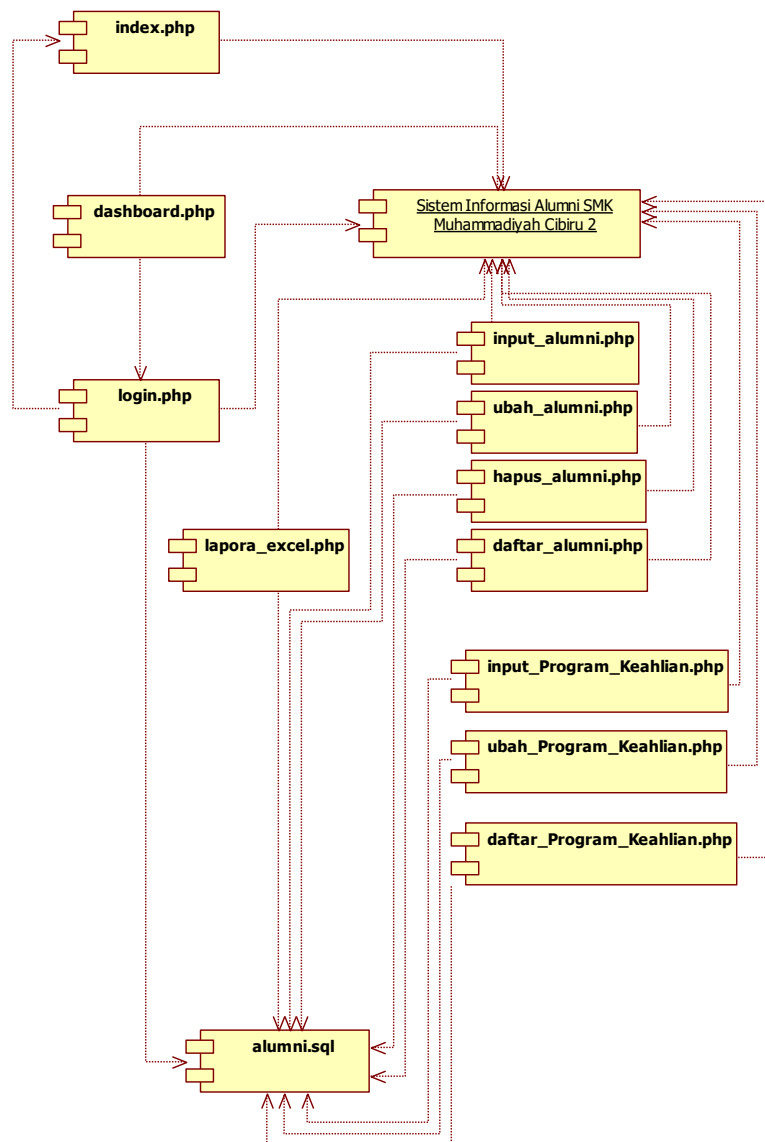
Berikut ini adalah deskripsi *Collaboration* pada *Collaboration Diagram* Sistem informasi Alumni SMK Muhammadiyah 2 Bandung yang diusulkan oleh penulis.

1. Alumni dapat melihat data alumni tanpa perlu melakukan *login* terlebih dahulu.
2. Petugas melakukan *login* pada sistem.
3. Sistem akan mengecek dan mencocokkan data *username* dan *password* yang diinputkan dengan tabel user. Jika sesuai maka pengguna dapat mengakses sistem informasi pelayanan administrasi kependudukan.
4. Petugas memasukan data alumni pada tabel alumni. Kemudian, sistem akan menyimpannya ke dalam database dan dapat ditampilkan dalam bentuk daftar alumni.
5. Petugas memasukan data program keahlian pada tabel alumni program keahlian. Kemudian, sistem akan menyimpannya ke dalam database dan dapat ditampilkan dalam bentuk daftar program keahlian.
6. Petugas dapat melihat data alumni yang telah dimasukan. Kemudian, petugas dapat melakukan perubahan terhadap data alumni ataupun menghapusnya sesuai dengan kebutuhan.
7. Petugas dapat melihat data program keahlian yang telah dimasukan. Kemudian, petugas dapat melakukan perubahan terhadap data program keahlian ataupun menghapusnya sesuai dengan kebutuhan.

8. Petugas dapat membuat laporan alumni berdasarkan tahun angkatan, domisili, dan sebagainya. Kemudian sistem akan merubah ke dalam format excel untuk dapat dicetak atau diarsipkan oleh petugas.

7. *Component Diagram*

Diagram komponen atau *component diagram* dibuat untuk menunjukkan organisasi dan ketergantungan diantara kumpulan komponen dalam sebuah sistem. Diagram komponen fokus pada komponen sistem yang dibutuhkan dan ada didalam sistem.



Gambar 4.7 Component Diagram

Berikut ini adalah deskripsi *Component* pada *Component Diagram* Sistem informasi Alumni SMK Muhammadiyah 2 Bandung yang diusulkan oleh penulis.

Tabel 4.7
Deskripsi Component pada Component Diagram

No	State	Deskripsi
1	<i>Login.php</i>	Komponen untuk antar muka <i>form login</i> .
2	Sistem Informasi Pendata alumni SMK Muhammadiyah 2 Bandung	Komponen utama Sistem Informasi Pendata alumni SMK Muhammadiyah 2 Bandung
3	Dashboard.php	Komponen antarmuka untuk petugas dalam melakukan proses olah data.
4	Index.php	Komponen untuk antarmuka halaman awal saat pertama kali sistem informasi di akses.
5	Input_alumni.php	Komponen antarmuka halaman untuk menampilkan <i>form input</i> alumni.
6	Input_alumni.php	Komponen antarmuka halaman untuk menampilkan <i>form input</i> program keahlian.
7	Daftar_alumni.php	Komponen antarmuka halaman untuk menampilkan daftar alumni dalam bentuk tabel.
8	Daftar_program_keahlian.php	Komponen antarmuka halaman untuk menampilkan daftar program keahlian dalam bentuk tabel.
9	Ubah_alumni.php	Komponen antarmuka halaman untuk menampilkan <i>form update</i> alumni.
10	Ubah_program_keahlian.php	Komponen antarmuka halaman untuk menampilkan <i>form update</i> program keahlian.
11	Hapus_alumni.php	Komponen antarmuka halaman untuk menampilkan <i>form delete</i> alumni.
12	Laporan.php	Komponen antarmuka halaman untuk menampilkan filter pembuatan laporan dan

		menampilkan laporan tersebut..
13	Alumni.sql	Komponen basis data untuk menampung data-data yang ada dalam Sistem Informasi Alumni.

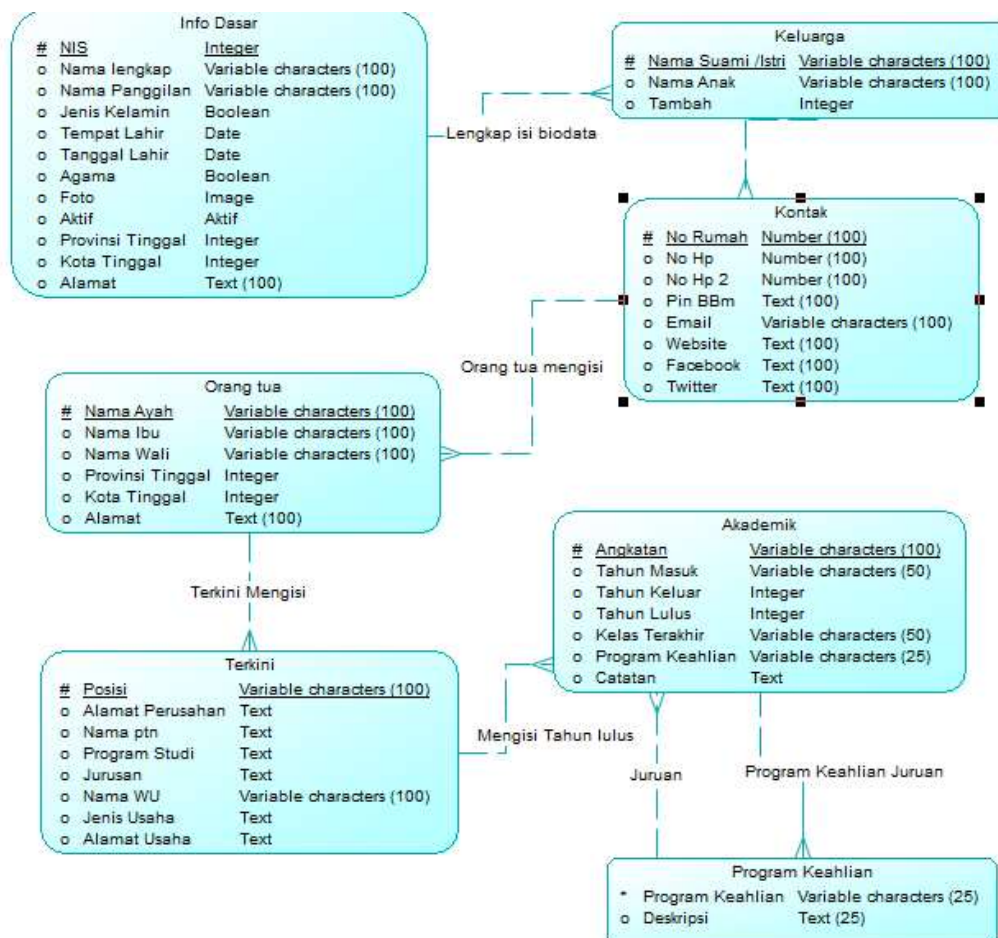
8. *Deployment Diagram*

Deployment diagram menunjukkan konfigurasi komponen dalam proses eksekusi aplikasi. *Deployment diagram* juga dapat digunakan untuk memodelkan hal-hal berikut ini:

- Sistem tambahan yang menggambarkan rancangan *device*, *node*, dan *hardware*.
- Sistem teridistribusi murni.
- Rekayasa ulang aplikasi

	pemdataan alumni SMK Muhammadiyah 2 Cibiru	yang akan dirancang pada sistem informasi alumni SMK Muhammadiyah 2 Cibiru bandung.
2	MySQL Database	Node yang berisi database yang digunakan dalam sistem.
3	<<device>> Printer PC	Node yang berisi perangkat keras yang digunakan untuk mengakses sistem dan mencetak laporan.

4.5 ERD (Entity Relationship Diagram)



Gambar 4.8 di bawah ini menjelaskan mengenai Entity Relationship Diagram aplikasi yang akan dibangun.

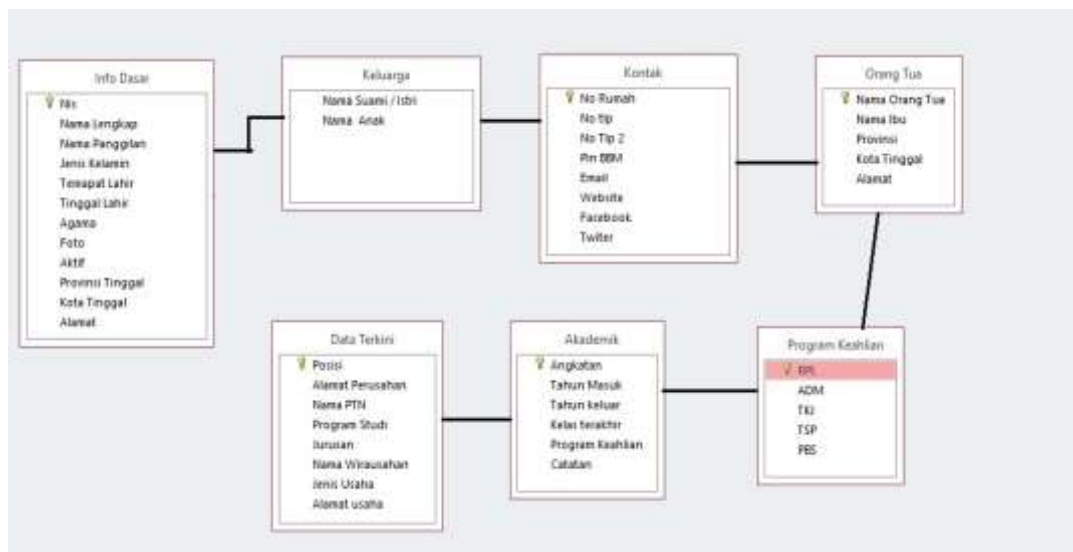
Gambar 4.9 ERD (Entity Relationship Diagram)

4.6 Kamus Data

1. Info_data_alumni = @id_nis + Nama_lengkap + Nama_panggilan + Jenis_Kelamin + Tempat_Lahir + Agama + foto + Aktif_Y/T + Provinsi_Tinggal + Kota_Tinggal + Alamat
2. keluarga = @id_Nama_suami/Istri + Nama_Anak
3. Orang = @id_Nama_Ayah + Nama_ibu + Provinsi_Tinggal + Kota_Tinggal + Alamat
4. Kontak = @id_No_Rumah + No_hp1 + No_hp2 + Pin_Blackberry + email + Website + Facebook + Twwitter
5. terkini = @id_posisi + alamat_Perusahaan + Nama_pt + Program_studi + Jurusan + Nama_Wirusaha + Jenis_Usaha + Alamat_Usaha
6. Akademik = @id_angkata + Tahun_masuk + Tahun_lulus + Kelas_Terakhir + Program Keahlian_pilihan + Catatat

4.7 Struktur Database Website Alumni SMK Muhammadiyah 2 BANDUNG

Struktur *database* merupakan kumpulan tabel-tabel yang saling berelasi. Struktur *database website* alumni SMK Muhammadiyah 2 BANDUNG juga terdiri dari tabel-tabel yang saling berelasi. Relasi atau hubungan tabel-tabel tersebut di jelaskan pada gambar di bawah ini.



Gambar 4.10 Stuktur *Database*

4.7.1 Struktur Tabel

Struktur tabel merupakan kumpulan tabel-tabel yang menyusun basis data. Tabel tersusun atas sejumlah *record* dimana *record* tersebut diisi oleh sejumlah *field*. Sebuah *field* tersimpan pada kumpulan *bit*. Struktur tabel *website* alumni SMK Muhammadiyah 2 BANDUNG yaitu:

1. Tabel Informasi alumni data

Nama Tabel : Informasi alumni data

Fungsi : untuk menyimpan biodata siswa alumni yang inputkan oleh admin

Primary Key :-

Foreign key :-

No	Nama Field	Type	Panjang
1	NIS	Int	100
2	Nama lengkap	Varchar	100

3	Nama Panggilan	Varchar	50
4	Jenis Kelamin	Enum	l/p
5	Tempat Lahir	Date	-
6	Tanggal Lahir	Date	-
7	Agama	Varchar	30
8	Foto	Image	-
9	Aktif	Enum	ya/tidak
10	Provinsi Tinggal	Varchar	30
11	Kota Tinggal	Varchar	30
12	Alamat	Text	100

Tabel 4.9. di atas menjelaskan mengenai struktur table admin yang terdapat dalam *database* sistem aplikasi portal pendataan alumni.

2. Tabel Keluarga

Nama Tabel : Keluarga data

Fungsi : untuk menyimpan keluarga alumni yang inputkan oleh admin

Primary Key :-

Foreign key :-

No	Nama Field	Type	Panjang
1	Nama Suami/Istri	Varchar	100
2	Nama anak	Varchar	100

Tabel 4.10. di atas menjelaskan mengenai struktur table admin yang terdapat dalam *database* sistem aplikasi portal pendataan alumni.

3. Tabel Orang Tua

Nama Tabel : Orang tua data

Fungsi : untuk menyimpan Orang Tua siswa alumni yang inputkan oleh admin

Primary Key :-

Foreign key :-

No	Nama Field	Type	Panjang
1	Nama Ayah	Int	100
2	Nama Ibu	Varchar	100
3	Nama wali	Varchar	100
4	Provinsi Tinggal	Integer	100
5	Kota Tinggal	Integer	100
6	Alamat	Text	100

Tabel 4.11. di atas menjelaskan mengenai struktur table admin yang terdapat dalam *database* sistem aplikasi portal pendataan alumni.

4. Tabel Kontak

Nama Tabel : Kontak data

Fungsi : untuk menyimpan kontak siswa alumni yang inputkan oleh admin

Primary Key :-

Foreign key :-

No	Nama Field	Type	Panjang
1	no rumah	Number	100
2	no hp	Number	100
3	pin BBM	Varchar	50
4	email	Text	100
5	website	Text	100

Tabel 4.12. di atas menjelaskan mengenai struktur table admin yang terdapat dalam database sistem aplikasi portal pendataan alumni.

5. Tabel Data Terkini

Nama Tabel : Keluarga data

Fungsi : untuk menyimpan terkini nama perusahaan, ptn, usaha siswa alumni yang inputkan oleh admin

Primary Key :-

Foreign key :-

No	Nama Field	Type	Panjang
1	posisi	varchar	26
2	alamat perusahaan	text	100
3	nama ptn	varchar	50
4	program std	varchar	100
5	jurusan	varchar	52

6	nama usahan	varchar	52
7	jenis usaha	varchar	52
8	alamat usahan	text	100

Tabel 4.13. di atas menjelaskan mengenai struktur table admin yang terdapat dalam *database* sistem aplikasi portal pendataan alumni.

6. Tabel Akademik

Nama Tabel : Akademik data

Fungsi : untuk menyimpan Akademik data tahun setelah lulus alumni yang inputkan oleh admin

Primary Key :-

Foreign key :-

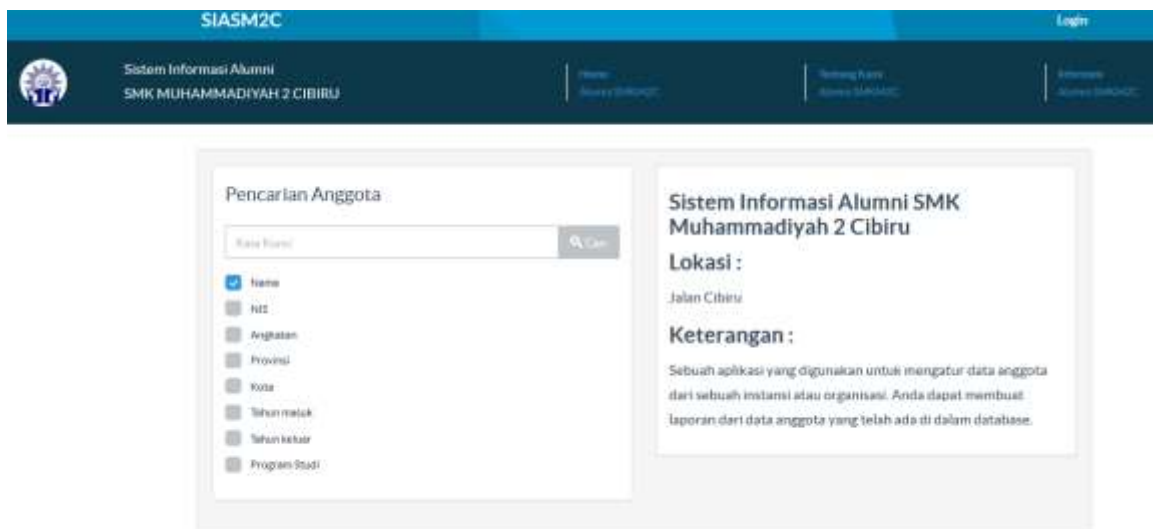
No	Nama Field	Type	Panjang
1	Angkatan	Varchar	50
2	Tahun Masuk	Varchar	100
3	Tahun Keluar	varchar	50
4	Kelas Terakhir	varchar	100
5	Program Keahlian	Varchar	50
6	Catatan	Text	-

Tabel 4.14. di atas menjelaskan mengenai struktur table admin yang terdapat dalam *database* sistem aplikasi portal pendataan alumni.

4.8. Perancangan Antarmuka (*Use Interface*)

4.8.1 Halaman *Home*

Halaman Home merupakan halaman awal website alumni ini. Halaman home ini dapat melakukan pencarian data anggota alumni dengan pencarian nama, NIS, angkatan, provinsi, kota, tahun masuk dan tahun keluar.

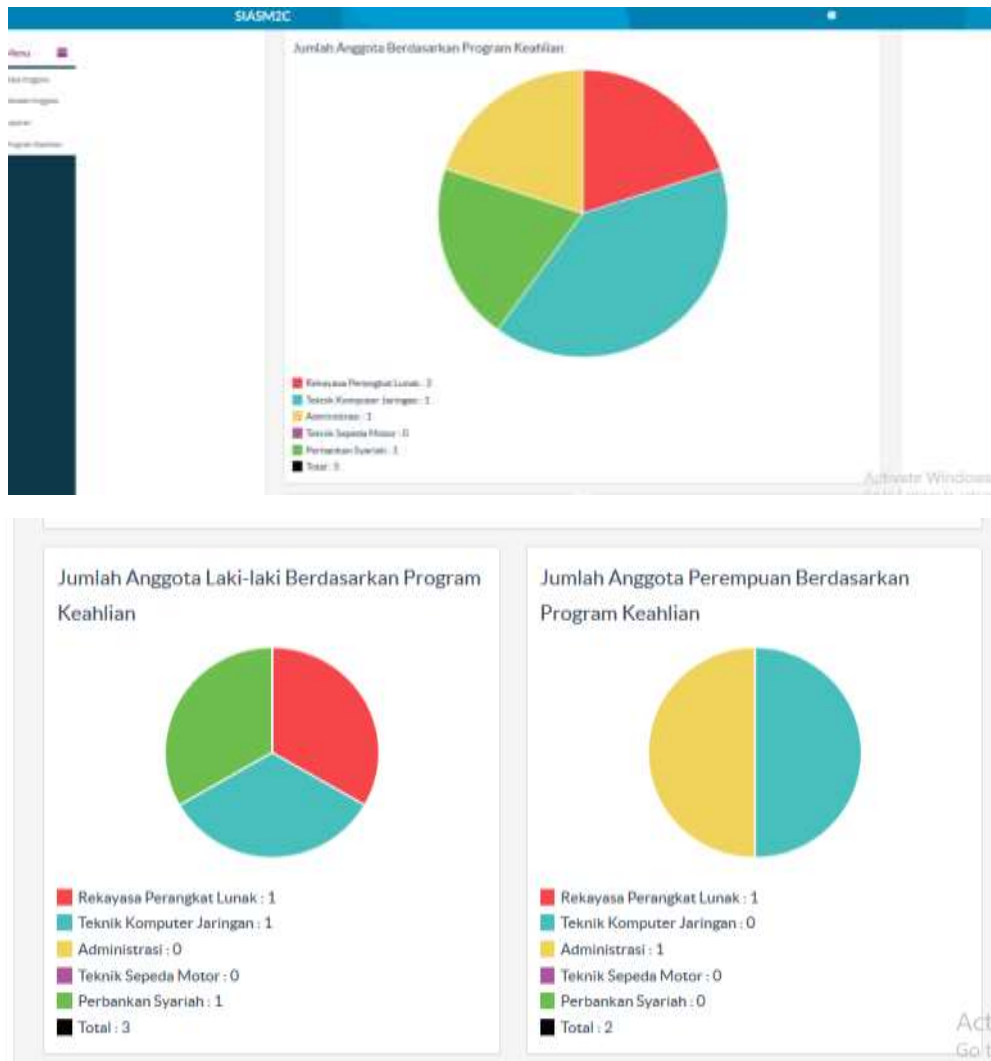


Keterangan Gambar 4.11. :

1. kolom “kata kunci” digunakan untuk pencarian cek data alumni
2. klik “Login” untuk login ke halaman backend oleh admin

4.8.2 Halaman *dashboard admin*

Halaman *Dashboard admin* gambar 4.10 ini menjelaskan jumlah chart diagram berkenaan dengan program keahlian jurusan, serta jumlah laki-laki dan perempuan per program keahlian.



Gambar 4.12 Halaman *Dashboard admin*

4.8.3. Halaman Daftar Biodata Alumni

Halaman daftar biodata alumni gambar 4.13 sampai 4.18 di bawah ini berisi biodata informasi mengenai alumni. Terdapat kolom data NIS, Nama lengkap siswa, keluarga , orang tua, kontak siswa, data terkini, akademik , data dapat diinput oleh admin.

Tambah Anggota

[Info Dasar](#)
[Keluarga](#)
[Orang Tua](#)
[Kontak](#)
[Data Terkini](#)
[Akademik](#)
[Aksi](#)

NIS

Nama lengkap

Nama panggilan

Nama panggilan harus diisi dan harus ada karena akan digunakan sebagai username untuk dapat masuk ke dalam sistem ini.

Jenis Kelamin ☒ Laki-laki ☐ Perempuan

Tempat Lahir

Tanggal Lahir

Agama

Foto

Format harus jpg, png atau gif. Maksimal ukuran 2 Mb.

Aktif ☒ Ya ☐ Tidak

Actis Go to

Gambar 4.13 Halaman Tambah lengkapi siswa alumni

Tambah Anggota

[Info Dasar](#)
[Keluarga](#)
[Orang Tua](#)
[Kontak](#)
[Data Terkini](#)
[Akademik](#)
[Aksi](#)

Nama Suami / Istri

Nama Anak

Gambar 4.14 Halaman Keluarga

Info Dasar Keluarga Orang Tua Kontak Data Terkini Akademik Akun

Nama Ayah

Nama Ibu

Nama Wali

Provinsi Tinggal

Kota Tinggal

Alamat

File Edit Insert View Format Tools

Gambar 4.15 Halaman Orang Tua

Tambah Anggota

Info Dasar Keluarga Orang Tua Kontak Data Terkini Akademik Akun

No Rumah

No Handphone

No Handphone 2

Pin Blackberry

Email

Website

Facebook

Twitter

Gambar 4.16 Halaman Kontak

Info Dasar
Keluarga
Orang Tua
Kontak
Data Terkini
Akademik
Akun

Posisi

Posisi

Alamat Perusahaan

Alamat Perusahaan

Nama Perguruan Tinggi

Nama Perguruan Tinggi

Program Studi

Program Studi

Jurusan

Jurusan

Nama Wirausaha

Nama Wirausaha

Jenis Usaha

Jenis Usaha

Alamat Usaha

Alamat Usaha

Simpan

Gambar 4.17. Data Terkini

Tambah Anggota

Info Dasar
Keluarga
Orang Tua
Kontak
Data Terkini
Akademik
Akun

Angkatan

Angkatan (contoh : 12)

Tahun Masuk

Tahun Masuk (contoh : 1990)

Tahun Lulus

Tahun Lulus (contoh : 1996)

Kelas Terakhir

Kelas Terakhir

Program Keahlian

- Pilih Program Keahlian -

Catatan

File
Edit
Insert
View
Format
Tools

p

Simpan

Gambar 4.18. Akademik

4.8.4. Halaman Tampilan Daftar Alumni

Halaman data anggota gambar 4.15 di bawah ini menampilkan summary data yang telah diinput oleh admin, dan admin bisa melakukan pengeditan kembali jika masih ada data yang mau ditambahkan atau diganti. Maupun Admin dapat menghapus data yang sudah diinput.

Daftar Anggota

Tampilkan entri Cari:

No	Nama Lengkap	Panggilan	JK	No Handphone	Angkatan	Program Keahlian	#
1	Adi Muhamad Malik	adimm	laki-laki		2016	RPL	<div><div></div><div></div><div></div></div>
2	Aini Dinah Umaroh	ainido	perempuan		2016	RPL	<div><div></div><div></div><div></div></div>
3	Aisha Nur Hasya	aishanh	perempuan		2016	ADM	<div><div></div><div></div><div></div></div>
4	Wahyudin	wahyudin	laki-laki		2016	PBS	<div><div></div><div></div><div></div></div>




Menampilkan 1 sampai 4 dari 4 entri Sebelumnya **1** Selanjutnya

Gambar 4.19 Halaman Tampilan data Anggota siswa alumni

4.8.5 Halaman Laporan Data

Halaman Laporan Data gambar 4.16 di bawah ini halaman laporan data summary laporan keseluruhan, laporan per angkatan, laporan lokasi provinsi, data dapat diunduh download gunakan excel langsung cek data siswa alumni.

Laporan

 <p>Laporan Keseluruhan Proses download laporan keseluruhan data anggota.</p> <p>Download</p>	 <p>Laporan Per Angkatan Proses download laporan berdasarkan angkatan anggota.</p> <p>- Pilih Angkatan -</p> <p>Download</p>	 <p>Laporan Per Provinsi Proses download laporan berdasarkan provinsi tinggal anggota.</p> <p>- Pilih Provinsi -</p> <p>Download</p>
---	--	--

Gambar 4.20 Halaman Laporan Data

BAB V

IMPLEMENTASI DAN PENGUJIAN SYSTEM

5.1. Implementasi

5.1.1 Implementasi Perangkat Lunak

Langkah pertama yang harus dilakukan dalam mengimplementasikan perangkat lunak yaitu dengan mempersiapkan sistem terlebih dahulu. Sistem yang harus dipersiapkan untuk mendukung kinerja *Website* Alumni SMK Muhammadiyah 2 Cibiru diantaranya:

- a. Sistem Operasi Windows 10
- b. Program XAMPP versi 3.2.1
- c. Apache versi 2.4.3
- d. MySQL versi 5.5.27
- e. PHP versi 5.4.7
- f. Browser (Mozilla Firefox atau Google Chrome)
- g. Text Editor (Notepad++, Sublime Text 3, visual studio)

5.1.2 Implementasi Perangkat Keras (Hardware)

Perangkat Keras (hardware) memiliki peran penting dalam pembangunan program pengolahan data. Untuk mengimplementasikan sistem informasi yang sudah dibuat diperlukan perangkat keras yang sesuai dengan sistem informasi yang diusulkan.

Adapun perangkat keras yang dibutuhkan diantaranya:

- h. Processor intel inside pentium
- i. Harddisk 200 GB
- j. RAM 200 MB

k. Monitor,keyboard,Mouse ,sebagai peralatan antar muka

5.1.3. Konversi Sistem

Konversi sistem dilakukan setelah program selesai dibuat, serta sarana pendukung sudah dipersiapkan dengan baik. Konversi sistem yaitu mengiputkan data – data yang diperlukan pada sistem baru, sehingga sistem yang dibangun yaitu Sistem Informasi alumni Pendataan SMK Muhammadiyah 2 Bandung harus dilakukan pengujian terlebih dahulu sebelum siap dioperasikan.

5.1.4. Pengujian Sistem

Pengujian sistem merupakan tahap dilakukannya pengujian terhadap sistem yang dibangun. Pengujian sistem ini bertujuan untuk memastikan bahwa sistem dapat beroperasi secara optimal. Tujuan utamanya yaitu untuk memastikan komponen-komponen sistem telah berfungsi sesuai dengan yang diharapkan. Pengujian sistem ini memeriksa fungsi-fungsi antar komponen sistem serta pengujian program secara keseluruhan. Pengujian sistem sangat penting untuk dilakukan, karena hal ini dibutuhkan untuk mengetahui apakah sistem dapat menerima *input*, memproses, dan memberikan *output* sesuai dengan yang diharapkan

5.1.5. Pemasangan Sistem

Pemasangan sistem yaitu proses *hosting* terhadap *website* yang sudah siap untuk dioperasikan. *Hosting* ini bertujuan agar *website* yang dibangun dapat diakses dan dioperasikan melalui *internet*. Untuk dapat menjalankan *Website* Alumni SMK Muhammadiyah 2 Bandung, maka diperlukan *hosting* terhadap *website* tersebut sebagai tempat disimpannya *website* tersebut sehingga dapat diakses melalui *internet*.

5.1.6. Coding

Pembuatan *website* menggunakan bahasa pemrograman *web* PHP. Untuk perancangan desain *website* menggunakan CSS/*Bootstrap*, HTML, *Jquery*, dan *Javascript*.

5.1.7. Implementasi Tabel

Implementasi basis data diambil berdasarkan perancangan basis data yang telah dibuat sebelumnya. Implementasi basis data menggunakan MySQL. Nama *Database* yang dibuat yaitu db_alumniSmk2.

Pada Gambar 5.1 merupakan struktur *database* db_alumniSmk2 pada Sistem Informasi Pendataan Alumni SMK Muhammadiyah 2 Bandung.

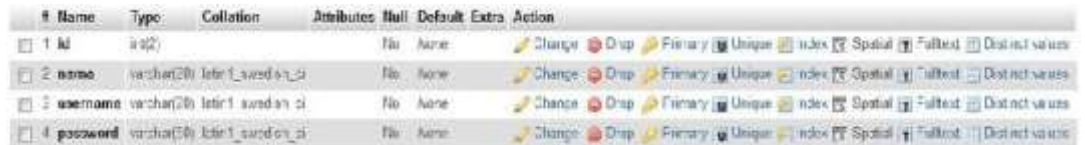
Table	Action	Rows	Type	Collation	Size	Overhead
<input type="checkbox"/> alumni_anggota_akademik	★ Browse Structure Search Insert Empty Drop	2	InnoDB	latin1_swedish_ci	16 B	-
<input type="checkbox"/> alumni_anggota_v <i>(Left Indent)</i>	★ Browse Structure Search Insert Empty Drop	2	InnoDB	latin1_swedish_ci	16 B	-
<input type="checkbox"/> alumni_anggota_keluarga	★ Browse Structure Search Insert Empty Drop	2	InnoDB	latin1_swedish_ci	16 B	-
<input type="checkbox"/> alumni_anggota_kontak	★ Browse Structure Search Insert Empty Drop	2	InnoDB	latin1_swedish_ci	16 B	-
<input type="checkbox"/> alumni_anggota_orang_tua	★ Browse Structure Search Insert Empty Drop	2	InnoDB	latin1_swedish_ci	16 B	-
<input type="checkbox"/> alumni_anggota_terkini	★ Browse Structure Search Insert Empty Drop	0	InnoDB	latin1_swedish_ci	16 B	-
<input type="checkbox"/> alumni_data_view	★ Browse Structure Search Insert Drop	-0	View	—	—	—
<input type="checkbox"/> alumni_m_agama	★ Browse Structure Search Insert Empty Drop	8	InnoDB	latin1_swedish_ci	16 B	-
<input type="checkbox"/> alumni_m_kota	★ Browse Structure Search Insert Empty Drop	430	InnoDB	latin1_swedish_ci	84 B	-
<input type="checkbox"/> alumni_m_modul	★ Browse Structure Search Insert Empty Drop	5	InnoDB	latin1_swedish_ci	16 B	-
<input type="checkbox"/> alumni_m_provinsi	★ Browse Structure Search Insert Empty Drop	24	InnoDB	latin1_swedish_ci	16 B	-
<input type="checkbox"/> alumni_pengaturan	★ Browse Structure Search Insert Empty Drop	2	InnoDB	latin1_swedish_ci	16 B	-
<input type="checkbox"/> alumni_pengguna	★ Browse Structure Search Insert Empty Drop	5	InnoDB	latin1_swedish_ci	16 B	-
<input type="checkbox"/> alumni_pengguna_hak_akses	★ Browse Structure Search Insert Empty Drop	1	InnoDB	latin1_swedish_ci	16 B	-
<input type="checkbox"/> alumni_pengguna_status_password	★ Browse Structure Search Insert Empty Drop	5	InnoDB	latin1_swedish_ci	16 B	-
<input type="checkbox"/> alumni_program_kahlian	★ Browse Structure Search Insert Empty Drop	5	InnoDB	latin1_swedish_ci	16 B	-
<input type="checkbox"/> alumni_v_anggota_lengkap	★ Browse Structure Search Insert Drop	-0	View	—	—	—
<input type="checkbox"/> alumni_v_anggota_mentah	★ Browse Structure Search Insert Drop	-0	View	—	—	—
<input type="checkbox"/> alumni_v_data	★ Browse Structure Search Insert Drop	-0	View	—	—	—
<input type="checkbox"/> alumni_v_pengguna	★ Browse Structure Search Insert Drop	-0	View	—	—	—

Gambar 5.1 *database* db_alumniSmk2

5.1.8. Implementasi Tabel

1. Tabel Admin

Tabel admin pada Gambar 5.2 merupakan tempat menyimpan informasi akun *administrator*. Tabel admin berisi informasi akun *administrator* untuk *login* ke halaman *administrator*.



#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	id	int(2)			No	None		Change Drop Primary Unique Index Spatial Fulltext Distinct values
2	name	varchar(20)	latin1_swedish_ci		No	None		Change Drop Primary Unique Index Spatial Fulltext Distinct values
3	username	varchar(20)	latin1_swedish_ci		No	None		Change Drop Primary Unique Index Spatial Fulltext Distinct values
4	password	varchar(50)	latin1_swedish_ci		No	None		Change Drop Primary Unique Index Spatial Fulltext Distinct values

Gambar 5.2. Tabel *admin*

2. Tabel Tambah alumni

Tabel tambah alumni pada Gambar 5.3 merupakan tabel yang menyimpan data alumni SMK Muhammadiyah 2 Bandung. Dalam tabel ini, semua data alumni SMK Muhammadiyah 2 Bandung tercatat dan tersimpan untuk dijadikan informasi bagi pengunjung *website*.

<input type="checkbox"/>	4	jenis_kelamin	enum('laki-laki', 'perempuan')	latin1_swedish_ci	No	None
<input type="checkbox"/>	5	tempat_lahir	varchar(50)	latin1_swedish_ci	No	None
<input type="checkbox"/>	6	tanggal_lahir	date		No	None
<input type="checkbox"/>	7	agama	varchar(30)	latin1_swedish_ci	Yes	NULL
<input type="checkbox"/>	8	foto	blob		Yes	NULL
<input type="checkbox"/>	9	provinsi	varchar(50)	latin1_swedish_ci	Yes	NULL
<input type="checkbox"/>	10	kota	varchar(50)	latin1_swedish_ci	Yes	NULL
<input type="checkbox"/>	11	alamat	text	latin1_swedish_ci	Yes	NULL
<input type="checkbox"/>	12	aktif	enum('ya', 'tidak')	latin1_swedish_ci	No	ya
<input type="checkbox"/>	13	angkatan	varchar(50)	latin1_swedish_ci	Yes	NULL
<input type="checkbox"/>	14	tahun_masuk	int(4)		No	None
<input type="checkbox"/>	15	tahun_keluar	int(4)		Yes	NULL
<input type="checkbox"/>	16	kelas_terakhir	varchar(100)	latin1_swedish_ci	Yes	NULL
<input type="checkbox"/>	17	catatan	text	latin1_swedish_ci	Yes	NULL
<input type="checkbox"/>	18	nama_pasangan	varchar(100)	latin1_swedish_ci	Yes	NULL
<input type="checkbox"/>	19	nama_anak	varchar(500)	latin1_swedish_ci	Yes	NULL
<input type="checkbox"/>	20	nama_ayah	varchar(100)	latin1_swedish_ci	Yes	NULL
<input type="checkbox"/>	21	nama_ibu	varchar(100)	latin1_swedish_ci	Yes	NULL
<input type="checkbox"/>	22	nama_wali	varchar(100)	latin1_swedish_ci	Yes	NULL
<input type="checkbox"/>	23	provinsi_orang_tua	varchar(50)	latin1_swedish_ci	Yes	NULL
<input type="checkbox"/>	24	kota_orang_tua	varchar(50)	latin1_swedish_ci	Yes	NULL
<input type="checkbox"/>	25	alamat_orang_tua	text	latin1_swedish_ci	Yes	NULL
<input type="checkbox"/>	26	no_rumah	varchar(15)	latin1_swedish_ci	Yes	NULL
<input type="checkbox"/>	27	no_handphone	varchar(15)	latin1_swedish_ci	Yes	NULL
<input type="checkbox"/>	28	no_handphone2	varchar(15)	latin1_swedish_ci	Yes	NULL

Gambar 5.3. Tabel data_anggota siswa

3. Tabel Anggota Keluarga

Tabel Anggota keluarga pada Gambar 5.4 merupakan tabel yang menyimpan data mengenai Anggota keluarga. Dalam tabel ini, semua data Anggota keluarga tercatat dan tersimpan untuk dijadikan informasi bagi pengunjung *website*.

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	id_anggota	int(12)			No	None		Change Drop Primary Unique Index Spatial Fulltext Distinct values
2	nama_pasangan	varchar(100)	latin1_swedish_ci		Yes	NULL		Change Drop Primary Unique Index Spatial Fulltext Distinct values
3	nama_anak	varchar(500)	latin1_swedish_ci		Yes	NULL		Change Drop Primary Unique Index Spatial Fulltext Distinct values
4	created_by	varchar(30)	latin1_swedish_ci		No	None		Change Drop Primary Unique Index Spatial Fulltext Distinct values
5	created_date	datetime			No	None		Change Drop Primary Unique Index Spatial Fulltext Distinct values
6	updated_by	varchar(30)	latin1_swedish_ci		No	None		Change Drop Primary Unique Index Spatial Fulltext Distinct values
7	updated_date	datetime			No	None		Change Drop Primary Unique Index Spatial Fulltext Distinct values
8	revisi	tinyint(2)			No	0		Change Drop Primary Unique Index Spatial Fulltext Distinct values

Gambar 5.4 Tabel anggota Keluarga

4. Tabel Kontak

Tabel Kontak pada Gambar 5.5 merupakan penyimpanan untuk data Kontak. Data dalam tabel ini digunakan untuk keperluan pendataan alumni. Tabel ini akan direlasikan ke tabel data alumni.

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	id_anggota	int(12)			No	None		Change Drop Primary Unique Index Spatial Fulltext Distinct values
2	no_rumah	varchar(15)	latin1_swedish_ci		Yes	NULL		Change Drop Primary Unique Index Spatial Fulltext Distinct values
3	no_handphone	varchar(15)	latin1_swedish_ci		Yes	NULL		Change Drop Primary Unique Index Spatial Fulltext Distinct values
4	no_handphone2	varchar(15)	latin1_swedish_ci		Yes	NULL		Change Drop Primary Unique Index Spatial Fulltext Distinct values
5	pin_blackberry	varchar(10)	latin1_swedish_ci		Yes	NULL		Change Drop Primary Unique Index Spatial Fulltext Distinct values
6	alamat_email	varchar(50)	latin1_swedish_ci		Yes	NULL		Change Drop Primary Unique Index Spatial Fulltext Distinct values
7	alamat_website	varchar(50)	latin1_swedish_ci		Yes	NULL		Change Drop Primary Unique Index Spatial Fulltext Distinct values
8	facebook	varchar(100)	latin1_swedish_ci		Yes	NULL		Change Drop Primary Unique Index Spatial Fulltext Distinct values
9	twitter	varchar(50)	latin1_swedish_ci		Yes	NULL		Change Drop Primary Unique Index Spatial Fulltext Distinct values
10	created_by	varchar(30)	latin1_swedish_ci		No	None		Change Drop Primary Unique Index Spatial Fulltext Distinct values
11	created_date	datetime			No	None		Change Drop Primary Unique Index Spatial Fulltext Distinct values
12	updated_by	varchar(30)	latin1_swedish_ci		No	None		Change Drop Primary Unique Index Spatial Fulltext Distinct values
13	updated_date	datetime			No	None		Change Drop Primary Unique Index Spatial Fulltext Distinct values
14	revisi	tinyint(2)			No	0		Change Drop Primary Unique Index Spatial Fulltext Distinct values

Gambar 5.5. Tabel Kontak

5. Tabel Orang Tua

Tabel orang tua pada Gambar 5.6 merupakan penyimpanan untuk data orang tua. Data dalam table ini digunakan untuk keperluan pendataan alumni. Tabel ini akan direlasikan ke table data alumni.

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	id_anggota	int(12)			No	None		Change, Drop, Primary, Unique, Index, Spatial, Fulltext, Distinct values
2	nama_sayah	varchar(100)	latin1_swedish_ci		Yes	NULL		Change, Drop, Primary, Unique, Index, Spatial, Fulltext, Distinct values
3	nama_ibu	varchar(100)	latin1_swedish_ci		Yes	NULL		Change, Drop, Primary, Unique, Index, Spatial, Fulltext, Distinct values
4	nama_sali	varchar(100)	latin1_swedish_ci		Yes	NULL		Change, Drop, Primary, Unique, Index, Spatial, Fulltext, Distinct values
5	id_provinsi	int(4)			Yes	NULL		Change, Drop, Primary, Unique, Index, Spatial, Fulltext, Distinct values
6	id_kota	int(4)			Yes	NULL		Change, Drop, Primary, Unique, Index, Spatial, Fulltext, Distinct values
7	alamat_orang tua	text	latin1_swedish_ci		Yes	NULL		Change, Drop, Primary, Unique, Index, Spatial, Fulltext, Distinct values
8	created_by	varchar(30)	latin1_swedish_ci		No	None		Change, Drop, Primary, Unique, Index, Spatial, Fulltext, Distinct values
9	created_date	datetime			No	None		Change, Drop, Primary, Unique, Index, Spatial, Fulltext, Distinct values
10	updated_by	varchar(30)	latin1_swedish_ci		No	None		Change, Drop, Primary, Unique, Index, Spatial, Fulltext, Distinct values
11	updated_date	datetime			No	None		Change, Drop, Primary, Unique, Index, Spatial, Fulltext, Distinct values
12	revisi	tinyint(2)			No	0		Change, Drop, Primary, Unique, Index, Spatial, Fulltext, Distinct values

Gambar 5.6 Orang Tua

6. Tabel Anggota Terkini

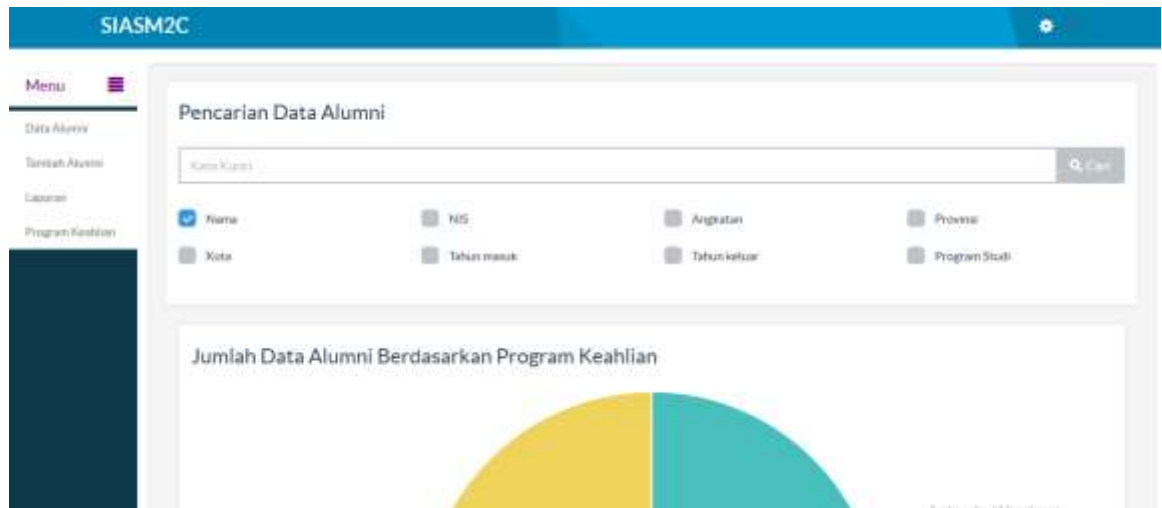
Tabel anggota terkini pada Gambar 5.7 merupakan penyimpanan untuk data setelah lulus sekolah. Data dalam tabel ini digunakan untuk keperluan pendataan alumni. Tabel ini akan direlasikan ke tabel data alumni.

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	id_anggota	int(11)			No	None		Change, Drop, Primary, Unique, Index, Spatial, Fulltext, Distinct values
2	posisi	varchar(20)	latin1_swedish_ci		No	None		Change, Drop, Primary, Unique, Index, Spatial, Fulltext, Distinct values
3	alamat_pemrosahan	text	latin1_swedish_ci		No	None		Change, Drop, Primary, Unique, Index, Spatial, Fulltext, Distinct values
4	nama_gin	varchar(50)	latin1_swedish_ci		No	None		Change, Drop, Primary, Unique, Index, Spatial, Fulltext, Distinct values
5	program_sdi	varchar(52)	latin1_swedish_ci		No	None		Change, Drop, Primary, Unique, Index, Spatial, Fulltext, Distinct values
6	jurusan	varchar(52)	latin1_swedish_ci		No	None		Change, Drop, Primary, Unique, Index, Spatial, Fulltext, Distinct values
7	nama_usaha	varchar(52)	latin1_swedish_ci		No	None		Change, Drop, Primary, Unique, Index, Spatial, Fulltext, Distinct values
8	jenis_usaha	varchar(52)	latin1_swedish_ci		No	None		Change, Drop, Primary, Unique, Index, Spatial, Fulltext, Distinct values
9	alamat_usaha	text	latin1_swedish_ci		No	None		Change, Drop, Primary, Unique, Index, Spatial, Fulltext, Distinct values

Gambar 5.7 Anggota Terkini

7. Tabel Akademik

Tabel Akademik pada Gambar 5.8 merupakan penyimpanan untuk data masuk dan keluar setelah lulus sekolah dari angkata. Data dalam tabel ini digunakan untuk keperluan pendataan alumni. Tabel ini akan direlasikan ke tabel data alumni.



Gambar 5.10 *dashboard administrator*

5.3 Pengujian Sistem

Pengujian sistem merupakan bagian penting yang harus dilakukan dalam pembangunan sebuah sistem atau perangkat lunak. Pengujian sistem ditunjukan untuk menemukan kesalahan-kesalahan pada sistem serta memastikan sistem yang dibangun sesuai dengan rancangan yang sudah direncanakan sebelumnya. Pengujian sistem dilakukan untuk menjamin kualitas sistem, bahwa sistem yang dibangun memiliki kualitas yang baik, yaitu mampu mempresentasikan kajian pokok dari spesifikasi analisi, perancangan, dan pengkodean sistem. Selain itu, melalui pengujian sistem ini juga untuk mengetahui kelemahan dari sistem itu sendiri.

5.3.1

Pengujian sistem ini menggunakan metode pengujian blackbox. Pengujian black fokus pada bersyarat fungsional sistem yang dibuat.

Kelas Uji	Butir Uji	Jenis Pengujian
<i>Login</i>	Verifikasi <i>login</i>	<i>Black box</i>
Menu <i>Administrator</i>	Memilih menu	<i>Black box</i>
Pengisian data	Pengisian data alumni	<i>Black box</i>
Kelola isi <i>content</i>	<i>Insert, update, delete</i>	<i>Black box</i>
Unduh <i>file</i>	Unduh <i>file</i> laporan	<i>Blackbox</i>

Tabel 5.1. Tabel Pengujian Blackbox

5.3.2 Pengujian Antarmuka Halaman Administrator

1. Ujian Halaman *Login*

Tabel 5.2 dibawah ini merupakan tabel pengujian halaman login sebelum masuk ke halaman *administrator* yang terdapat dalam *database* dengan menggunakan *blackbox*.

Kode	Skenario	Hasil		Keterangan
		Sukses	Gagal	
Req01	<i>Login</i> dengan mengisi <i>username</i> dan <i>password</i> yang benar	√		<i>Login</i> Sukses dan <i>Admin</i> berhasil masuk <i>Dashboard Administrator</i>
Req02	<i>Login</i> dengan mengisi <i>username</i> dan <i>password</i> yang salah	√		Sistem menampilkan pesan “anda gagal <i>login</i> , silahkan <u><i>login kembali</i></u> ”
Req03	Mengisi form login dengan <i>username</i> dan <i>password</i> kosong.	√		Sistem menampilkan pesan “ <i>please fill out this field</i> ”, yang berarti tidak boleh kosong

Tabel 5.2. Ujian Halaman *Login*

2. Uji Menu Administrator

Pada Tabel 5.3 merupakan tabel pengujian menu *administrator* untuk menampilkan semua menu dan link.

Kode	Skenario	Hasil		Keterangan
		Sukses	Gagal	
Req04	Klik semua menu	√		Berhasil dibuka.
Req05	Klik semua link	√		Berhasil berpindah halaman.

Tabel 5.3. Uji Menu *Administrator*

3. Ujian Pengisian Data Alumni

Pada Tabel 5.4 merupakan table pengujian input data alumni yang akan disimpan di *database*.

Kode	Skenario	Hasil		Keterangan
		Sukses	Gagal	
Req06	Mengisi data alumni dengan lengkap.	√		Berhasil menambahkan data alumni
Req07	<i>Form</i> tidak diisi dengan lengkap	√		Tampil pesan “ <i>please fill out this field</i> ” yang berarti harus diisi
Req08	Menambahkan alamat email tidak sesuai dengan format email	√		Tampil pesan untuk menambahkan alamat email dengan tanda “@”

Tabel 5.4. Uji Pengisian Data Alumni

4. Uji Kelola Content (*inset,update,delete*)

Pada Tabel 5.5 merupakan tabel pengujian kelola *content* seperti *insert*, *update*, dan *delete* yang akan disimpan di *database*.

Kode	Skenario	Hasil		Keterangan
		Sukses	Gagal	
Req09	Memasukan data baru	√		Berhasil tersimpan
Req010	Mengubah data yang ada	√		Berhasil diubah
Req011	Menghapus data	√		Berhasil terhapus

Tabel 5.5. Tabel Uji Kelola *Content* (*insert, update, delete*)

5. Uji Unduh File Laporan

Pada Tabel 5.6 merupakan tabel pengujian unduh *file* laporan data alumni pertahun berupa table *excel*.

Kode	Skenario	Hasil		Keterangan
		Sukses	Gagal	
Req012	Klik tombol <i>export/download</i>	√		Berhasil mengunduh <i>file</i>

Tabel 5.6. Uji unduh file laporan

BAB VI

PENUTUP

6.1. Kesimpulan

Proses pembangunan Aplikasi Sistem Informasi Pendataan berbasis web Alumni SMK Muhammadiyah 2 Bandung melalui beberapa tahapan-tahapan yang dapat ditarik kesimpulan. Kesimpulan tersebut antara lain:

1. Sisi User Sistem Informasi Alumni Pendataan SMK Muhammadiyah 2 Bandung dibangun menggunakan bahasa pemrograman php,css,javascript,jquery dan database MySQL.
2. Sistem Informasi Alumni Pendataan SMK Muhammadiyah 2 Bandung dibangun untuk memberikan informasi yang dibutuhkan bagi pihak sekolah maupun *user*.
3. Konten-konten yang terdapat dalam halaman *Administrator* Sistem Informasi Alumni Pendataan SMK Muhammadiyah 2 Bandung disesuaikan dengan kebutuhan pihak sekolah dan *user*, sehingga dapat digunakan sesuai dengan fungsinya.

6.2 Saran

Sesuai dengan judul awal Sistem Informasi Alumni Pendataan SMK Muhammadiyah 2 Bandung yaitu aplikasi ini mampu membantu pihak sekolah dalam menyajikan informasi mengenai kegiatan alumni serta membantu dalam menyajikan informasi yang terkait dengan kebutuhan *user* dengan efektif dan efisien. Mempertimbangkan beberapa manfaat dari *website* ini, baiknya proyek ini

terus dikembangkan secara berkelanjutan dan juga diperlukannya perawatan.

Beberapa saran untuk pengembangan aplikasi kedepannya diantaranya:

1. Penambahan lengkap biodata data siswa pada tampilan alumni oleh admin.
2. *Username* dan *password* sudah disiapkan dari pihak sekolah, agar yang *login* pada halaman user hanya Sistem Informasi Pendataan Alumni SMK Muhammadiyah 2 Bandung dan tanpa registrasi terlebih dulu.
3. Alumni dapat menambahkan aktivitas yang pernah dilakukan dan yang sedang dilakukan.
4. Perlu ditingkatkan kembali kerja sama antara pihak sekolah dengan pihak pengembang. Pengangkatan *staff* khusus untuk mengelola *website* ini juga dianggap perlu agar *website* ini dapat berjalan sebagaimana mestinya.

Daftar Pustaka

- [1] Anarchyta D. (2015, Agustus 19). Diambil Kembali dari Sagga-US:
<http://www.saggaus.net/2015/08/pengertian-aplikasi-menurut-ahli.html>
- [2] Jaya (2010). *CSS untuk Orang Awan*. Palembang : Maxikom
- [3] Jogyanto (1989). *Analisis & Design*. Yogyakarta: Andi.
- [4] Nugroho, B. (2009). *Latihan Membuat Aplikasi Web PHP dan MySQL dengan Dreamweaver MX*. Yogyakarta: Gava Media.
- [6] Permatasari Novi Endah, dkk. (2014). *Laporan Praktek Kerja Nyata (Perancangan Sistem Informasi Berbasis Website di SMK Bina Wiyata Karang Malang Sragh*. Surakarta.
- [7] Salahudin, M., & Rosa, A. (2013). *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*. Bandung: Informatika Bandung.

Lampiran