

SISTEM INFORMASI E-LEARNING STMIK MUHAMMADIYAH JAKARTA

SKRIPSI

Ditulis untuk memenuhi sebagian persyaratan dalam
menempuh gelar sarjana Komputer
Jenjang Pendidikan Strata – 1



RUBI CHANDRAPUTRA

17.1.0.0004

**PROGRAM STUDI SISTEM INFORMASI
SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER
MUHAMMADIYAH JAKARTA
2019**

HALAMAN PERSETUJUAN

SKRIPSI

Sistem Informasi E-Learning STMIK Muhammadiyah Jakarta

RUBI CHANDRAPUTRA

NIM: 17.1.0.0004

Jakarta, 22 Juni 2019

Menyetujui	Menyetujui
Ka. Prodi Sistem Informasi	Dosen Pembimbing
<u>M.Arief Sutisna, S.Kom, M.Kom</u>	<u>M.Arief Sutisna, S.Kom, M.Kom</u>
NBM: 841.536	NBM: 841.536

Mengetahui
Wakil Ketua Bidang Akademik
STMIK Muhammadiyah Jakarta

M.Zen Fitriansyah, S,Kom
NBM:841537

KATA PENGANTAR

Puji syukur penulis panjatkan kehadiran Allah SWT yang telah berkenan memberikan rahmat dan hidayah-Nya, sehingga penulisan Skripsi dapat diselesaikan dengan tepat waktu. Penulis menyadari bahwa tanpa bimbingan dan dorongan dari semua pihak, penulisan Skripsi ini tidak dapat terselesaikan dengan baik. Oleh karena itu pada kesempatan ini penulis menyampaikan rasa terima kasih kepada:

1. Bapak Faiz Rafdhi CH, M.Kom selaku Ketua STMIK Muhammadiyah Jakarta.
2. Bapak M. Zen Fitriansyah, S.Kom selaku Wakil Ketua Bidang Akademik.
3. Bapak M. Arief Sutisna, S.Kom selaku Wakil Ketua Bidang Keuangan.
4. Bapak Himawan Dwiatmodjo.SH,LL.MM selaku Wakil Ketua Bidang Kemahasiswaan.
5. Bapak M. Arief Sutisna, S.Kom selaku Kepala Jurusan Sistem Informasi.
6. Bapak M. Arief Sutisna, S.Kom selaku pembimbing dalam penulisan skripsi ini.
7. Ibu saya dan ayah saya beserta keluarga yang telah memberikan dukungan terbaik bagi penulis.
8. Seluruh staff dan dosen di lingkungan STMIK Muhammadiyah Jakarta.
9. Sahabat, teman-teman dan rekan-rekan mahasiswa STMIK Muhammadiyah Jakarta yang telah memberikan dorongan semangat dalam pembuatan sproposal ini.

Penulis tidak mampu membalas jasa baiknya, hanya berdoa semoga Allah SWT memberikan ganjaran yang setimpal dengan amal perbuatannya. Dan semoga tulisan ini dapat berguna bagi penulis dan pembaca dalam rangka pengembangan ilmu pengetahuan.

Jakarta, 22 Juni 2019

Rubi Chandraputra

ABSTRAK

Rubi Chandraputra, 17.1.0.0004. **Sistem Informasi *E-Learning* STMIK Muhammadiyah Jakarta**. Skripsi. Jakarta: Program Studi Sistem Informasi Sekolah Tinggi Manajemen Informatika dan Komputer Muhammadiyah Jakarta. 2019

STMIK Muhammadiyah Jakarta menawarkan banyak keuntungan bagi seluruh mahasiswanya yang sedang bekerja. Dengan menawarkan jam perkuliahan yang luwes, STMIK Muhammadiyah menjadi salah satu sekolah tinggi yang dapat dijadikan pilihan utama bagi mereka yang sedang bekerja, namun ingin menempuh jenjang perkuliahan. Namun keuntungan yang ditawarkan tersebut, bukanlah menjadi sebuah solusi terhadap kurang efektif dan efisiennya sistem perkuliahan di dalam kelas. Maka dari itu, diciptakan sebuah Sistem Informasi *E-Learning* yang dapat meminimalisir kekurangan tersebut. Sistem Informasi *E-Learning* yang dirancang khusus bagi seluruh mahasiswa STMIK Muhammadiyah, berfungsi sebagai media yang memberikan fasilitas berupa kemudahan belajar di tengah-tengah kesibukan. Sistem Informasi ini juga memberikan fungsi tambahan berupa ketersediaan materi panduan (*Tutorial*) yang dipaparkan dengan jelas. Materi panduan tersebut dilengkapi dengan latihan uji tiap beberapa sesi, yang merupakan proyek kecil untuk mengimplementasikan semua yang telah dipelajari dalam bentuk program. Dosen juga dapat *memposting* tugas-tugas yang harus diselesaikan oleh mahasiswa.

Sistem Informasi ini dibangun menggunakan sistem rekayasa SDLC (*System Development Life Cycle*) dengan perangkat permodelan UML (*Unified Modelling Language*).

Dibuatnya Sistem Informasi *E-Learning* STMIK Muhammadiyah Jakarta adalah sebagai bentuk upaya perbaikan sarana belajar di lingkungan STMIK Muhammadiyah Jakarta, dengan mengurangi kurangnya efisiensi dan efektivitas waktu perkuliahan di dalam kelas, demi menciptakan lulusan yang berkompeten di bidangnya.

Kata Kunci: *Sistem Informasi, E-Learning, CRUD, Aplikasi, Web Developer.*

DAFTAR ISI

SISTEM INFORMASI E-LEARNING STMIK MUHAMMADIYAH JAKARTA.....	i
HALAMAN PERSETUJUAN	ii
KATA PENGANTAR	iii
ABSTRAK.....	iv
DAFTAR ISI	v
BAB I.....	1
1.1 Latar Belakang.....	1
BAB I Pendahuluan.....	3
BAB II Landasan Teori	4
BAB III Metodologi Penelitian.....	4
BAB IV Hasil dan Pembahasan	4
BAB V Penutup	4
BAB II	5
2.1 Penelitian Sebelumnya.....	5
2.2 Sistem	7
2.3 Informasi.....	10
2.4 Sistem Informasi.....	11
2.5 Perancangan Sistem	12
2.6 E-Learning	12
2.7 Sistem Informasi E-Learning.....	13
2.7.1 Ciri-ciri Sistem Informasi E-Learning.....	13
2.8 Internet.....	14
2.8.1 Pengertian Internet.....	14
2.8.2 HyperText Transfer Protocol (HTTP).....	14
2.8.3 Uniform Resource Locator (URL).....	15

2.8.4	Web Browser	15
2.8.5	Email Gateway.....	16
2.9	DATABASE	16
2.9.1	Pengertian <i>Database</i>	16
2.9.2	Database Management System(DBMS)	16
2.10	PERANGKAT LUNAK IMPLEMENTASI SISTEM	17
2.10.1	Web Application.....	17
2.10.2	Web Server	17
2.10.3	PHP	18
2.10.4	MySQL	18
2.10.5	APACHE	19
2.10.6	XAMPP.....	19
2.10.7	Composer.....	20
2.10.8	Node.js	21
2.10.9	Laravel	21
2.11	TOOLS PERANCANGAN SISTEM	22
2.11.1	UML(<i>Unified Modeling Language</i>).....	22
2.11.2	Class Diagram.....	23
2.11.3	Use Case Diagram.....	24
2.11.4	Activity Diagram	27
2.11.5	Sequence Diagram	29
BAB III	32
3.1	Tujuan Dan Manfaat Penelitian	32
3.1.1	Tujuan Penelitian	32
3.1.2	Manfaat Penelitian	32
3.2	Tempat Dan Waktu Penelitian.....	33
3.2.1	Gambaran Umum Instansi Pendidikan	33
3.2.1.1	Sejarah	33
3.3	Jadwal Penelitian	34

3.4	Objek Penelitian.....	35
3.5	Metode Pengumpulan Data.....	35
3.5.1	Obervasi Dan Wawancara.....	35
3.6	Metodologi Pengembangan Perangkat Lunak	37
3.6.1	<i>Waterfall Model</i>	37
3.6.2	Kelebihan dan kekurangan <i>Waterfall Model</i>	39
3.7	Analisa Sistem Yang Berjalan	Error! Bookmark not defined.
3.7.1	Activity Diagram Sistem Perkuliahan	51
DAFTAR PUSTAKA		86

BAB I

PENDAHULUAN

1.1 Latar Belakang

Sebagai salah satu sekolah tinggi yang menyediakan program studi Sistem Informasi dan Teknik Informatika, STMIK Muhammadiyah Jakarta menjadi salah satu pilihan sekolah dengan jenjang Strata satu(S1) yang relevan. Dengan biaya yang lebih terjangkau, kurikulum yang terbaru, dan jam kuliah yang relatif terbilang ‘luwes’, STMIK Muhammadiyah menjadi sekolah tinggi yang dapat menjadi pilihan utama bagi para kaum *Milennial* yang berstatus lulusan SMA/SMK maupun karyawan.

Sekolah tinggi yang berpusat di Ciracas dengan 2(dua) cabang lain yang berlokasi di Matraman dan Bekasi ini, menyediakan 3(tiga) jenis waktu perkuliahan yaitu reguler(khusus ciracas), reguler malam dan karyawan. Jam kuliah yang disesuaikan dengan kebutuhan mahasiswa, sangat menjadi dambaan bagi karyawan yang sibuk bekerja, namun ingin berkuliah. Sarana ini tentu menjadi hal positif bagi mahasiswa, namun menjadi kekurangan tersendiri bagi efektifnya jam belajar.

Ada beberapa kekurangan yang saya lihat dan saya alami sendiri ketika mengikuti kelas regular malam di kampus matraman. Efektifnya jam kampus merupakan permasalahan yang paling utama. Faktor kelelahan, *deadline*, tugas keluar kota sudah menjadi hal wajar bagi mahasiswa sebagai alasan untuk tidak menghadiri perkuliahan. Hal ini mengakibatkan kurang baiknya mahasiswa dalam menguasai seluruh materi pembelajaran. Yang akan berujung pada terlukanya instansi perguruan tinggi karena pada akhirnya, berpotensi untuk meluluskan mahasiswa-mahasiswa yang kurang kompeten di bidangnya.

Faktor-faktor di luar perkuliahan, yang menyebabkan kurang baiknya proses belajar, seperti dosen maupun mahasiswa lelah setelah bekerja, sibuk dengan urusan kantor, dapat mengurangi intensitas interaksi antara dosen dan mahasiswa di dalam kelas. Tidak bisa kita salahkan pula faktor-faktor tersebut terhadap

kurang baiknya sistem pembelajaran, karena pada dasarnya, seluruh mahasiswa bekerja untuk membiayai kuliah dan kebutuhan hidup yang lain.

Maka dari itu perlu diciptakan sebuah solusi yang dapat menjawab segala kekurangan pada sistem pembelajaran STMIK Muhammadiyah Jakarta. Tidak untuk mengganti sistem yang sudah berjalan, tapi melengkapinya dengan sistem baru yang menyediakan seluruh bahan ajar yang sesuai ke dalam *platform* yang lebih fleksibel. Dimana mahasiswa dapat mengakses materi-materi yang dosen berikan, dimanapun dan kapanpun, dengan cara yang tidak membosankan, dan lebih dari sekedar membaca.

Berdasarkan permasalahan-permasalahan yang dijelaskan diatas, maka sangat dibutuhkan sekali sistem informasi yang mendukung proses pembelajaran terutama dalam hal mengantisipasi faktor-faktor di luar perkuliahan. Oleh karena itu, penelitian ini merencanakan “*Sistem Informasi E-Learning STMIK Muhammadiyah Jakarta*”. Sistem yang berbasis web dapat diakses dimanapun dan kapanpun dengan cara-cara pembelajaran yang efektif sebagai solusi belajar di luar kampus, sehingga memudahkan mahasiswa untuk tetap belajar di sela-sela kesibukan mereka.

1.2 Identifikasi Masalah

Melihat adanya latar belakang diatas, maka identifikasi masalah dalam penelitian ini adalah sebagai berikut:

1. Sistem informasi *E-Learning* yang diperuntukan bagi STMIK Muhammadiyah belum terealisasi.
2. Kurangnya efektivitas jam belajar pada kelas reguler malam dan kelas karyawan yang mengakibatkan kurangnya kemampuan mahasiswa dalam menguasai materi kuliah.
3. Belum adanya sebuah sistem informasi yang memudahkan mahasiswa STMIK Muhammadiyah, untuk mempelajari kembali materi kuliah.

1.3 Rumusan Masalah

Guna memperjelas permasalahan yang timbul, dalam hal ini diformulasikan ke dalam sebuah bentuk rumusan masalah. Adapun rumusan masalahnya:

1. Bagaimana merancang sistem informasi *E-Learning* STMIK Muhammadiyah Jakarta yang dengan tepat menjawab segala kebutuhan mahasiswa sebagai sebuah solusi dari kurangnya efektivitas jam belajar di dalam kelas.
2. Bagaimana Sistem Informasi *E-Learning* STMIK Muhammadiyah Jakarta ini dapat diimplementasikan ke dalam sistem perkuliahan yang sesuai dengan SKS dan kebutuhan mahasiswa sehingga dapat mengoptimalkan efektivitas jam belajar bagi mahasiswa.

1.4 Batasan Masalah

Untuk lebih mengarahkan pembuatan rencana skripsi ini dan untuk mempermudah pembahasan maka pembatasan masalahnya adalah:

- a. Pembahasan masalah ini hanya dibatasi pada lingkup STMIK Muhammadiyah Jakarta
- b. Aplikasi yang dibangun diperuntukkan sebagai repository dari seluruh Modul yang dibuat oleh dosen maupun admin.
- c. Untuk memenuhi kebutuhan, perancangan sistem informasi *E-Learning* STMIK Muhammadiyah Jakarta ini dengan bahasa pemrograman PHP dan database menggunakan MySQL.

1.5 Sistematika Penulisan

BAB I Pendahuluan

Pada bab I ini berisi latar belakang masalah, tujuan, batasan masalah, metodologi penelitian, sistematika penulisan dan kerangka pemikiran.

BAB II Landasan Teori

Bab II dari penulisan buku ini menjelaskan tentang tinjauan pustaka yaitu teori-teori bagaimana tentang sistem informasi. Selain itu juga menjelaskan mekanisme belajar dan mengajar yang berjalan di STMIK Muhammadiyah Jakarta, menjelaskan bagaimana itu metode SDLC Model *Fountain* dan hal lain yang berhubungan dengan penelitian ini.

BAB III Metodologi Penelitian

Pada Bab III menjelaskan gambaran bagaimana akan dibangun Sistem Informasi *E-Learning* STMIK Muhammadiyah Jakarta untuk membantu mahasiswa untuk menguasai materi-materi yang belum dipahami dalam segala kesibukan, agar tidak terjadi penurunan kualitas lulusan yang tidak berkompeten. Perancangan sistem yang digunakan untuk mengolah data, antara lain; APL(Arsitektur Perangkat Lunak), pemodelan system UML *United Model Language* diantaranya *actifity diagram*, *usescase diagram*, *class diagram*, *sequence diagram*.

BAB IV Hasil dan Pembahasan

Bab ini menjelaskan tentang bagaimana Sistem Informasi *E-Learning* STMIK Muhammadiyah Jakarta diimplementasikan mulai dengan tahapan pemrograman, spesifikasi perangkat keras dan perangkat lunak, implementasi basis data, proses *input*, *output* serta menu-menu yang mewakili modul yang ada.

BAB V Penutup

Didalam bab V ini berisi uraian singkat atau kesimpulan dan saran dari penelitian ini.

BAB II

LANDASAN TEORI

2.1 Penelitian Sebelumnya

Penelitian sebelumnya merupakan daftar referensi dari semua jenis referensi seperti buku, *jurnal papers*, artikel, disertasi, tesis, skripsi, *hand outs*, *laboratory manuals*, dan karya ilmiah lainnya yang dikutip di dalam penulisan proposal. Semua referensi yang tertulis dalam kajian pustaka harus dirujuk di dalamnya. Referensi ditulisurut menurut abjad huruf awal dari nama akhir/keluarga penulis pertama dan tahun penerbitan (yang terbaru ditulis lebih dahulu).

Dalam Penelitian biasanya diawali dengan ide-ide atau gagasan dan konsep-konsep yang dihubungkan satu sama lain melalui hipotesis tentang hubungan yang diharapkan. Ide-ide dan konsep-konsep untuk penelitian dapat bersumber dari gagasan peneliti sendiri dan dapat juga bersumber dari sejumlah kumpulan pengetahuan hasil kerja sebelumnya yang kita kenal juga sebagai literatur atau pustaka. Literatur atau bahan pustaka ini kemudian kita jadikan sebagai referensi atau landasan teoritis dalam penelitian.

Sebagai bahan perbandingan dalam penelitian yang dilakukan, disini diambil beberapa contoh jurnal serta skripsi yang pernah dibuat oleh lulusan dari kampus maupun dari luar kampus STMIK Muhammadiyah Jakarta.

Tabel 2.1 *Daftar Penelitian Sebelumnya*

No	Judul Penelitian	Th	Peneliti	Pembahasan
1	Sistem Informasi Akademik Berbasis Web Pada MTS Al-Muawanah Kecamatan Curug Kabupaten Tangerang	2010	Ahmad Khairul Rijal	Perancangan Sistem Informasi akademik dengan menggunakan metode RAD.
2	Sistem Informasi Akademik Berbasis Web SMA Negeri 1 Bandar	2013	M. Hery Setiawan	Membahas tentang pembuatan sistem informasi akademik bagi SMA Negeri 1 Bandar dengan PHP dan MySql
3	Pembuatan Aplikasi Sistem Informasi Monitoring Kegiatan Mahasiswa Berbasis Web dan Android Client.	2015	Erdi Risnandar	Penelitian Ini membahas tentang monitoring kegiatan mahasiswa.
4	Rancang Bangun Sistem Informasi Akademik dan e-Learning Terpadu	2016	M. Muzakhi	Membahas tentang perancangan dan pembuatan sistem informasi akademik yang dipadukan dengan e-learning
5	Perancangan Sistem Monitoring Pengerjaan Skripsi pada STMIK Stikom Bali Berbasis Web	2017	Putu Ramayasa, Ida Bagus Ketut Suryana Arnawa	Membahas tentang monitoring skripsi yang terjadi akibat kendala-kendala yang ada dalam penulisan skripsi baik kendala dari sisi mahasiswa maupun dari sisi dosen pembimbing

2.2 Sistem

Simpulan dari beberapa ahli yang mengerti dengan sistem, bahwasanya sistem adalah prosedur-prosedur yang saling berkaitan membentuk suatu kerangka kerja yang utuh untuk melaksanakan suatu perintah atau pekerjaan sehingga dalam pelaksanaannya antara satu elemen dengan elemen yang lain saling berinteraksi demi mencapai suatu tujuan yang sudah di tentukan.¹

Sedangkan pandangan beberapa ahli yang digunakan sebagai bahan referensi adalah sebagai berikut:

Menurut Azhar Susanto: **“Sistem adalah kumpulan/group dari sub sistem/bagian/komponen/apapun baik phisik ataupun non phisik yang saling berhubungan satu sama lain dan bekerja sama secara harmonis untuk mencapai tujuan tertentu”.**²

Menurut Sutarman: **“Sistem adalah kumpulan elemen yang saling berinteraksi dalam suatu kesatuan untuk menjalankan suatu proses pencapaian suatu tujuan utama.”**³

Dari beberapa pandangan yang ada dari para ahli, ada yang mengatakan bahwa sistem erat kaitanya dengan prosedur. Keduanya tidak bisa dipisahkan satu sama lain. Suatu sistem bisa berjalan apabila di dalamnya terdapat beberapa prosedur yang berjalan.

Menurut Mulyadi: **“Sistem Adalah jaringan prosedur yang dibuat menurut pola yang terpadu untuk melaksanakan kegiatan-kegiatan pokok perusahaan sedangkan prosedur adalah urutan kegiatan klerikal, biasanya melibatkan beberapa orang dalam satu department atau lebih yang dibuat untuk menjamin penanganan secara seragam transaksi perusahaan yang terjadi secara berulang-ulang.”**⁴

¹ Bambang Harianto, *Dasar informatika dan Komputer*(Yogyakarta: Graha Ilmu, 2008), h27-28

² Susanto Azhar, *Sistem Informasi Akuntansi*, (Bandung :Linggar Jaya, 2013), h.22

³ Sutarman, *Pengantar teknologi informasi*, (Jakarta: Bumi Aksara, 2009), h.5

⁴ Mulyadi, *Sistem Akuntansi*, (Jakarta :Salemba Empat, 2010), h.5

Dari pendapat para ahli diatas dapat diambil kesimpulan bahwa sebenarnya sistem adalah sekelompok unsur yang hubungannya saling berkaitan satu dengan yang lain, difungsikan bersama demi mencapai suatu tujuan tertentu. Maka dengan kesimpulan diatas dapat dirinci dalam pengertian sistem yang lebih global sebagai berikut:⁵

1. Setiap sistem terdiri dari unsur-unsur
2. Dari unsur-unsur yang merupakan bagian sistem yang terpadu dan saling bersangkutan.
3. Unsur sistem yang saling bersangkutan bekerja sama untuk mencapai tujuan tertentu.
4. Suatu sistem merupakan bagian dari sistem yang lain yang lebih besar.

Suatu Sistem baru bisa dikatakan sebuah sistem informasi apabila didalamnya mengandung dan memenuhi karakteristik utama dari sebuah sistem informasi. Karakteristik utama tersebut menunjukkan bahwa sistem yang dibuat benar-benar memberikan solusi dari informasi awal yang didapatkan dalam sistem tersebut dengan alur mulai dari host sampai dengan user atau penerimanya. Beberapa karakteristik sistem yang ada sebagai berikut:

1. Komponen Sistem(*Component*)

Suatu sistem yang ada, terdiri dari kumpulan-kumpulan komponen yang saling berinteraksi. Setiap komponen yang ada berfungsi untuk menjalankan fungsi-fungsi tertentu dari sebuah sistem. Komponen tersebut sering kita kenal dengan sebutan subsistem. Apabila salah satu komponen tersebut tidak bekerja sebagaimana mestinya, maka sistem yang ada tidak dapat bekerja secara optimal.

2. Batasan Sistem(*Boundary*)

Karakteristik sistem yang selanjutnya adalah bahwa sistem informasi memiliki batasan, maksudnya disini sistem informasi harus memiliki pembatas dengan sistem informasi yang lain. Sehingga tugas dari masing-masing sistem

⁵ Edhy Sutanta ST, *Sistem Informasi Manajemen*,(Yogyakarta: Graha Ilmu, 2003)h.4-6

terbagi secara jelas, tidak saling tumpang tindih dan tidak saling mengacaukan peranan sistem yang lain. Hal tersebut karena ruang lingkup sebuah sistem itu sudah ada pembagian tersendiri, tidak bisa dicampuradukkan.

3. Lingkungan Luar Sistem (*Environment*)

Selanjutnya, karakteristik dari sistem informasi bahwa sebuah sistem informasi haruslah memiliki lingkungan luar. Environment bisa dikatakan keseluruhan sistem dan juga lingkungan yang berada diluar batasan atau boundary dari sebuah sistem informasi. Tanpa adanya environment, sistem yang ada tidak akan berarti apa-apa.

4. Penghubung Sistem(*Interface*)

Suatu interface atau antarmuka, merupakan karakteristik sebuah sistem yang harus dimiliki. Karakteristik tersebut mengacu pada user yang nanti akan menggunakan sistem tersebut bahwa tujuan dibuatnya sistem adalah agar user yang awam sekalipun dapat dengan mudah mengoperasikannya. Interface juga bisa dikatakan sebuah komponen khusus yang ada pada sistem, apabila interface suatu sistem tidak ada, maka sistem informasi tidak dapat berjalan secara optimal.

5. Masukan sistem (*Input*)

Karakteristik selanjutnya adalah input, bisa dikatakan suatu energy yang dimasukkan kedalam sistem. Masukan tersebut dapat berupa masukan perawatan (*maintenance*) agar sistem dapat beroperasi dan masukan signal (*signal input*) yaitu energy yang diproses melalui sistem informasi menuju output.

6. Pengolahan Sistem (*Proses*)

Pengolahan sistem merupakan karakteristik dari sistem informasi yang memiliki tugas utama untuk memproses masukan sistem. Lebih jelasnya pengolahan sistem ini membantu proses pengolahan data secara keseluruhan yang ada sistem informasi, lalu mengirimkan hasil olahan data tersebut menuju output dan selanjutnya informasi yang ada dapat diterima oleh user/pengguna.

7. Keluaran Sistem (*Output*)

Keluaran sistem merupakan energy dari masukan sistem yang sudah diolah oleh pengolahan sistem yang selanjutnya hasil dari pengolahan tersebut dapat dimanfaatkan untuk masukan bagi subsistem yang lain, ataupun untuk user/pengguna.

8. Sasaran Sistem (*Objektive*)

Karakteristik dari sistem yang terakhir adalah objektive/sasaran sistem. Suatu sistem bisa dikatakan berhasil apabila mengenai sasaran yang dituju. Sasaran dari sistem dapat juga dikatakan sebuah analisis berupa siapa saja yang akan menggunakan sistem tersebut. Misalkan sistem informasi yang akan dijalankan adalah para staff pergudangan yang merekap data-data barang maka sistem informasi yang digunakan oleh mereka adalah sistem informasi inventori.

2.3 Informasi

Informasi adalah suatu hasil pengolahan berasal dari data yang sudah dimanipulasi dan dirangkum sedemikian rupa dengan bentuk dan cara yang lain dari pada bentuk aslinya sebagai tujuan untuk mendapatkan hasil akhir sebagai sarana pengambilan keputusan⁶. Pandangan beberapa ahli juga beragam akan tetapi merujuk pada hal yang sama.

Menurut Jogianto : **“Informasi adalah data yang diolah menjadi bentuk yang lebih berguna bagi yang menerimanya”**⁷

Menurut Azhar Susanto : **“Informasi adalah hasil pengolahan data yang memberikan arti dan manfaat”**⁸

Sesuai dengan pandangan para ahli diatas, dapat disimpulkan kembali bahwa sebenarnya informasi adalah sekumpulan data yang sudah diolah sedemikian rupa, sehingga dapat digunakan bagi yang membutuhkannya baik nanti digunakan sebagai sarana pengambilan keputusan ataupun juga sebagai sarana pengetahuan.

⁶ Edhy Sutanta ST, *Sistem Informasi Manajemen*, (Yogyakarta: Graha Ilmu, 2003)h.10

⁷ Jogianto, *Analisis dan Desain Sistem Informasi*, Edisi II, (Yogyakarta, 2004),h.8

⁸ Susanto Azhar, *Sistem Informasi Akuntansi*, (Bandung: Linggar Jaya, 2013),h.46

Terlepas dari pandangan para ahli diatas tentang pengertian informasi, kualitas dari informasi pun juga perlu diperhatikan. Berikut kualitas dari informasi yang umumnya digunakan sebagai acuan:

1. **Akurat**, akurat disini maksudnya adalah informasi yang di dapat harus bebas dari kesalahan, tidak menyesatkan, tidak ada perkiraan atas informasi yang di dapat. Maksud dari perkiraan diatas adalah, misal seseorang membutuhkan informasi atas jawaban dari pertanyaanya dan membutuhkan satu jawaban tepat, maka hasil yang di dapatkan juga harus berupa jawaban yang tepat bukan jawaban kurang dari nilai yang diharapkan atau lebih dari itu.
2. **Tepat Waktu**, Setiap informasi yang diinginkan, biasanya berhubungan dengan ketepatan waktu yang ada. Sebuah informasi yang diterima melebihi dari waktu yang ditentukan biasanya akan menjadi sebuah informasi yang kedaluarsa dan tidak berguna sehingga menyebabkan operasional yang ada terganggu. Sehingga diperlukanlah ketepatan waktu atas jawaban dari setiap permintaan informasi yang ada.
3. **Relevan**, relevan disini maksudnya adalah informasi yang diberikan kepada user yang sedang mencari informasi adalah benar-benar informasi yang dibutuhkan. Jadi misalkan seorang akuntan sedang mencari informasi tentang bagaimana rumus neraca, maka yang ditampilkan adalah rumus dari neraca yang dimaksud, bukan rumus persamaan pythagoras atau rumus persamaan atom kimia⁹.

2.4 Sistem Informasi

Sistem Informasi adalah suatu sistem komputer yang menyediakan informasi berdasarkan olahan data yang diterima melalui input – proses - output kepada user dimana user itu sendiri membutuhkan informasi tersebut untuk membantunya

⁹ Edhy Sutanta ST, *Sistem Informasi Manajemen*, (Yogyakarta: Graha Ilmu, 2003)h.13

dalam mengambil keputusan. Sistem informasi sendiri adalah sebuah kerangka dimana untuk mendapatkan sebuah informasi yang dibutuhkan maka perlu mengkoordinasikan antara sistem komputer dengan manusia.

Sistem Informasi sendiri memiliki fungsi yang berbeda, dalam pengembangannya, sistem informasi punya tujuan sesuai dengan kebutuhan masing-masing user yang akan menggunakan. Adapun dalam penerapannya, sistem yang sudah berjalan tidak bisa berdiri sendiri tanpa menggunakan basis data untuk sarana menampung informasi.

2.5 Perancangan Sistem

Perancangan sistem bisa dikatakan suatu kegiatan untuk melakukan perancangan, penggambaran, dan pendesainan baik untuk pembuatan sistem baru ataupun menggantikan sistem yang secara sebagian ataupun keseluruhan.

2.6 E-Learning

E-Learning adalah sebuah sistem atau konsep pendidikan yang memanfaatkan teknologi informasi dalam suatu proses belajar mengajar.

Berikut pengertian *E-Learning* dari beberapa sumber:

Menurut Michael Allen dalam bukunya *Michael Allen's Guide to E-learning*: **“E-learning adalah Pembelajaran yang disusun ialah dengan tujuan menggunakan suatu sistem elektronik atau juga komputer sehingga mampu untuk mendukung suatu proses pembelajaran”**¹⁰

Menurut Rusman : **“E-Learning merupakan segala aktivitas belajar yang menggunakan bantuan teknologi elektronik. Melalui e-learning, pemahaman siswa tentang sebuah materi tidak tergantung pada guru/instruktur tetapi dapat diperoleh dari media elektronik. Teknologi elektronik yang banyak digunakan misalnya internet, intranet, tape**

¹⁰Michael Allen, *Michael Allen's Guide to E-learning*, (Canada: John Wiley & Sons, 2013)h.27

¹²Dewey John, *Teori E-Learning menurut Cendikiawan*, Diakses: 15 May 2019 pukul 17.40, dari: <https://www.silabus.web.id/e-learning/>

video atau audio, penyiaran melalui satelit, televisi interaktif serta CD-ROM.”

Pengertian tersebut didukung oleh pendapat Elliot Masie, Cisco, dan Cornelia menyatakan pengertian e-learning adalah **“Pembelajaran dimana bahan pembelajaran disampaikan melalui media elektronik seperti internet, intranet, satelit, tv, CDROM, dan lain-lain.”**

Jadi *E-Learning* tidak hanya dapat dilakukan dengan internet, banyak contoh media elektronik yang dapat digunakan, dan internet merupakan salah satu bagian dari *E-Learning*.

Menurut Nana Syaodih Sukmadinata : **“E pada *e-learning* tidak hanya singkatan dari electronic saja akan tetapi merupakan singkatan dari experience (pengalaman), extended (perpanjangan), dan expended (perluasan).”**¹²

Dalam pengertian di atas, Nana Syaodih Sukmadinata menekankan bahwa kata *electronic* pada *e-learning* artinya memanfaatkan adanya penambahan unsur teknologi pada proses belajar sehingga lebih melibatkan perangkat keras, perangkat lunak, dan proses elektronik lain.

2.7 Sistem Informasi E-Learning

Sistem Informasi E-Learning adalah suatu penggabungan antara sistem informasi dengan kegiatan e-learning yang mana fungsi dari sistem tersebut adalah untuk menghubungkan suatu proses komunikasi antara dosen dan mahasiswa, sehingga dapat berjalan sesuai tujuannya. Sistem informasi elearning telah banyak digunakan untuk kepentingan pembelajaran bahkan terdapat banyak website yang menyediakan fasilitas untuk melakukan hal tersebut. Sistem informasi elearning juga merupakan salah satu metode unggulan dalam program Digitalent Academy yang diselenggarakan oleh KOMINFO pada tahun 2019 ini.

2.7.1 Ciri-ciri Sistem Informasi E-Learning

Adapun ciri-ciri dari Sistem Informasi E-Learning adalah sebagai berikut:

1. Memanfaatkan jasa teknologi elektronik. Sehingga mendapatkan informasi dan berkomunikasi dengan mudah dan cepat baik untuk pengajar dengan pembelajar, atau pembelajar dengan pembelajar.
2. Memanfaatkan media komputer, contohnya jaringan komputer atau media digital.
3. Memakai materi pembelajaran untuk dipelajari secara mandiri oleh peserta didik atau murid.
4. Materi pembelajaran dapat disimpan di komputer sehingga dapat diakses oleh guru dan siswa kapan saja dan di mana saja ketika diperlukan kembali.
5. Memanfaatkan komputer untuk proses pembelajaran dan juga untuk mengetahui hasil kemajuan belajar, atau administrasi pendidikan serta untuk memperoleh informasi yang banyak dari berbagai sumber informasi.

2.8 Internet

2.8.1 Pengertian Internet

Berasal dari dua kata yaitu *interconnection network* adalah jaringan komputer yang saling terhubung antara satu dengan yang lain baik melalui media kabel LAN, *Wifi*, ataupun Satelit sehingga membentuk sistem jaringan mencakup seluruh dunia (*Global Network*).

2.8.2 HyperText Transfer Protocol (HTTP)

Adalah protokol standar sampai saat ini yang harus diikuti oleh *web browser* untuk meminta dokumen atau halaman serta untuk *web server* dalam menyediakan permintaan yang datang dari browser¹¹.

2.8.3 Uniform Resource Locator (URL)

Uniform Resource Locator adalah alamat yang menentukan lokasi suatu file atau halaman yang akan dituju pada suatu *web server*¹². Adapun URL yang ada saat ini terbagi menjadi beberapa bagian seperti berikut:

1. Protokol yang akan digunakan oleh *web browser* untuk mengambil informasi pada *web server*.
2. Nama komputer/server tempat dimana halaman/informasi itu disimpan.
3. Serta nama file informasi yang sedang kita akses.

Pada kasus ini, Sistem Informasi E-Learning masih berjalan diatas Laravel Server Development. Sistem informasi ini berjalan pada alamat default <http://127.0.0.1:8000/> :

Nama Protokolnya : http://

Nama Servernya : 127.0.0.1

Nama Port : 8000

Nama Filenya : / (Laravel bekerja menggunakan Blade Engine, tanda Slash tersebut artinya kita sedang mengakses direktori public pada laravel. Dengan menggunakan Route, apabila direktori tersebut dipanggil pada URL, file welcome.blade.php akan ditampilkan)..

2.8.4 Web Browser

¹¹ Bambang Harianto, *Dasar informatika dan Komputer*(Yogyakarta: Graha Ilmu, 2008), h232 & 240

¹² Bambang Harianto, *Dasar informatika dan Komputer*(Yogyakarta: Graha Ilmu, 2008), h233 & 241

Web Browser adalah suatu sistem aplikasi yang digunakan di komputer dirancang untuk me-*request* serta menampilkan informasi-informasi dari server.

Contoh web browser yang sering kita temui pada komputer adalah:

- Mozilla Firefox
- Google Chrome
- Safari
- Internet Explorer

2.8.5 Email Gateway

Email gateway adalah sebuah sistem yang digunakan untuk mengirim ataupun menerima *Email* yang mana di dalam sistem tersebut terdapat konfigurasi yang menjembatani antara sistem *email* dengan aplikasi yang akan diintegrasikan dengan *email* tersebut.

2.9 DATABASE

2.9.1 Pengertian Database

Secara umum, *database* atau dalam bahasa Indonesia yaitu basis data adalah koleksi data yang saling berkaitan. Dengan kata lain database dapat dianggap sebagai suatu data yang disusun dan terstruktur disimpan dalam media pengingat (*hard disk*) yang tujuannya adalah agar data yang disimpan dapat diakses dengan mudah dan cepat.¹³

2.9.2 Database Management System(DBMS)

¹³ Kadir Abdul, *Belajar Database Menggunakan MySQL*, (Yogyakarta: ANDI OFFSET, 2008), h.3

Database Management System atau yang sering disingkat dengan DBMS adalah sebuah software yang digunakan untuk mengelola database, dalam hal ini mulai dari membuat database itu sendiri kemudian membuat tabel serta field-field yang dibutuhkan, melakukan proses-proses yang berlaku setelah database itu dibuat seperti *insert*, *edit*, *delete* atau melakukan *query* terhadap data yang sebelumnya sudah dibuat¹⁴.

Ada beberapa macam database seperti database *hirarkis*, jaringan dan *relasional* akan tetapi dalam penelitian ini hanya akan membahas dan menggunakan database *relasional* karena dianggap database tersebut adalah yang paling cocok untuk digunakan dalam penulisan. Sebuah database *relasional* / RDBMS tersusun atas sejumlah tabel yang memiliki hubungan dengan tabel lainnya.¹⁵

2.10 PERANGKAT LUNAK IMPLEMENTASI SISTEM

2.10.1 Web Application

Web Application / Aplikasi Web adalah suatu perangkat lunak yang dikembangkan untuk mempermudah dari sisi developer ataupun dari sisi user dimana penggunaanya cukup menggunakan web browser tanpa harus menginstal satu persatu di masing-masing pc. Selain itu aplikasi web bisa diakses dari mana saja selama aplikasi tersebut terkoneksi dengan layanan internet baik melalui *operating system* yang sama ataupun berbeda dengan kata lain web aplikasi mendukung sistem operasi *cross platform*.

2.10.2 Web Server

¹⁴ Bambang Harianto, *Dasar informatika dan Komputer*(Yogyakarta: Graha Ilmu, 2008), h196-197

¹⁵ Kadir Abdul, *Belajar Database Menggunakan MySQL*,(Yogyakarta: ANDI OFFSET, 2008),h.3

Web Server adalah server yang digunakan dan mampu untuk melayani koneksi transfer antar data di dalam protokol HTTP. *Web Server* adalah suatu hal yang wajib dimiliki dalam pembuatan aplikasi berbasis web karena bertugas menjalankan alur proses sistem yang berjalan¹⁶.

2.10.3 PHP

PHP (*Hypertext Preprocessor*) adalah bahasa pemrograman open source dibawah lisense *PHP License* dikembangkan pada tahun 1995 oleh Rasmus Lerdorf yang dapat dipergunakan oleh semua orang secara gratis tanpa harus membayar, dirancang sebagai bahasa pemrograman *script server-side* yang didesain untuk pengembangan aplikasi berbasis web. Disebut sebagai bahasa pemrograman *server-side* karena PHP diproses pada komputer utama yaitu server berbeda dengan bahasa pemrograman lain seperti javascript atau turunanya yang pemrosesannya dilakukan pada web browser masing-masing client¹⁷.

2.10.4 MySQL

MySQL(dibaca: *mai-se-kyu-el*) adalah software yang tergolong sebagai DBMS (*Database Management System*) yang bersifat *Open Source* yang sudah mendukung manajemen basis data relasional RDBMS dibawah lisence GPL(*General Public Lisence*). Dalam keteranganya *Open Source* menyatakan bahwa software ini dilengkapi dengan *source code*, artinya kode yang dipakai untuk membuat MySQL itu sendiri disertakan dalam paket tersebut, selain itu *executeble*-nya atau kodenya yang dapat dijalankan di

¹⁶ Bambang Harianto, *Dasar informatika dan Komputer*(Yogyakarta: Graha Ilmu, 2008), h234

¹⁷ Bambang Harianto, *Dasar informatika dan Komputer*(Yogyakarta: Graha Ilmu, 2008), h237

semua sistem operasi yang dapat diperoleh dengan cara mendownload langsung dari internet secara gratis¹⁸.

Sebagai software DBMS yang sudah mendukung relasional MySQL memiliki sejumlah fitur diantaranya:

- *Multiplatform*
- Andal, cepat, dan mudah digunakan.
- Jaminan Keamanan Akses
- Mendukung Banyak Bahasa Pemrograman
- Dukungan SQL

2.10.5 APACHE

Apache adalah salah satu dari sekian *web server* yang bersifat open source yang sudah digunakan oleh banyak *developer* dalam mengembangkan programnya baik yang *open source* ataupun yang berbayar. Apache saat ini menjadi *web server* paling populer di kalangan *programmer* karena kemudahan penggunaannya dan support hampir di semua sistem operasi.

2.10.6 XAMPP

Xampp adalah perangkat lunak bebas, yang mendukung banyak sistem operasi yang merupakan kompilasi dari berbagai program. Fungsinya adalah sebagai server yang berdiri sendiri (localhost), yang terdiri atas program HTTP Apache Server, DBMS MySQL, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan Perl.

Singkatan dari XAMPP itu sendiri, yaitu:

- X = Program ini dapat dijalankan di banyak sistem operasi

¹⁸ Kadir Abdul, Belajar Database Menggunakan MySQL, (Yogyakarta: ANDI OFFSET, 2008), h.2

- A = Apache artinya paket aplikasi XAMPP ini didalamnya sudah tersedia apache server.
- M = MySQL dimana aplikasi XAMPP ini sudah mendukung Database Management System yaitu MySQL.
- P = PHP dimana aplikasi ini support untuk aplikasi web yang menggunakan bahasa pemrograman berbasis PHP.
- P = Perl dimana aplikasi ini juga mensupport penggunaan bahasa perl sebagai basis dari aplikasi.



Gambar 2.1 *Tampilan XAMPP control panel*

2.10.7 Composer

Composer adalah alat manajemen *dependency* pada PHP seperti npm (Node.js) dan Bundler (Ruby). Composer memungkinkan untuk membuat library pada projek dan composer sendiri akan menginstall atau mengupdate secara otomatis tanpa anda harus menginstall manual¹⁹.

¹⁹ IDCloudHost, *Pengertian dan Manfaat Composer Bagi Developer*, Diakses: 13 Juni 2019 pukul 21.30, dari: <https://idcloudhost.com/pengertian-dan-manfaat-composer-bagi-developer/>



Gambar 2.2 Logo Composer

(sumber: <https://getcomposer.org/img/logo-composer-transparent3.png>)

2.10.8 Node.js

Node.js adalah perangkat lunak yang didesain untuk mengembangkan aplikasi berbasis web dan ditulis dalam sintaks bahasa pemrograman JavaScript. Bila selama ini kita mengenal JavaScript sebagai bahasa pemrograman yang berjalan di sisi client / browser saja, maka Node.js ada untuk melengkapi peran JavaScript sehingga bisa juga berlaku sebagai bahasa pemrograman yang berjalan di sisi server, seperti halnya PHP, Ruby, Perl, dan sebagainya. Node.js dapat berjalan di sistem operasi Windows, Mac OS X dan Linux tanpa perlu ada perubahan kode program.



Gambar 2.3 Logo Node.js

(sumber: <https://nodejs.org/static/images/logos/nodejs-new-pantone-black.png>)

2.10.9 Laravel

adalah sebuah framework PHP yang dirilis dibawah lisensi MIT, dibangun dengan konsep MVC (model view controller). Laravel adalah pengembangan website berbasis MVP yang ditulis dalam PHP yang dirancang

untuk meningkatkan kualitas perangkat lunak dengan mengurangi biaya pengembangan awal dan biaya pemeliharaan, dan untuk meningkatkan pengalaman bekerja dengan aplikasi dengan menyediakan sintaks yang ekspresif, jelas dan menghemat waktu²⁰.



Gambar 2.4 *Logo Laravel*

(sumber: <http://www.webcubicttechnologies.com/wp-content/uploads/2017/10/laravel-logo-1.png>)

2.11 TOOLS PERANCANGAN SISTEM

2.11.1 UML(*Unified Modeling Language*)

UML merupakan suatu metode untuk memvisualisasikan perancangan sistem yang berorientasi objek, UML sendiri menjadi bahasa standarisasi visualisasi agar orang dari belahan dunia yang lain juga bisa mengerti rancangan yang kita buat. Bisa dikatakan UML adalah sebagai rancangan dasar suatu program(*penggambaran rancangan*) untuk mendapatkan hasil akhir dari suatu program secara maksimal²¹.

Ada beberapa macam jenis UML yang ada:



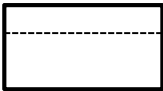
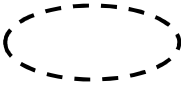
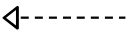
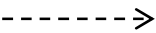
²⁰ IDCloudHost, *Pengertian dan Keunggulan Framework Laravel*, Diakses: 7 Juni 2019 pukul 21.40, dari: <https://idcloudhost.com/pengertian-dan-keunggulan-framework-laravel/>

²¹ Sholiq, *Pemodelan Sistem Informasi Berorientasi Obyek dengan UML*(Yogyakarta: Graha Ilmu,2006)h.6-7

2.11.2 Class Diagram

Tujuan utama dari diagram kelas adalah untuk menciptakan sebuah kosa kata yang digunakan untuk menuangkan pemikiran analysts agar dapat diterima oleh pengguna. Class Diagram biasanya merupakan ide-ide atau konsep yang ada dalam aplikasi²².

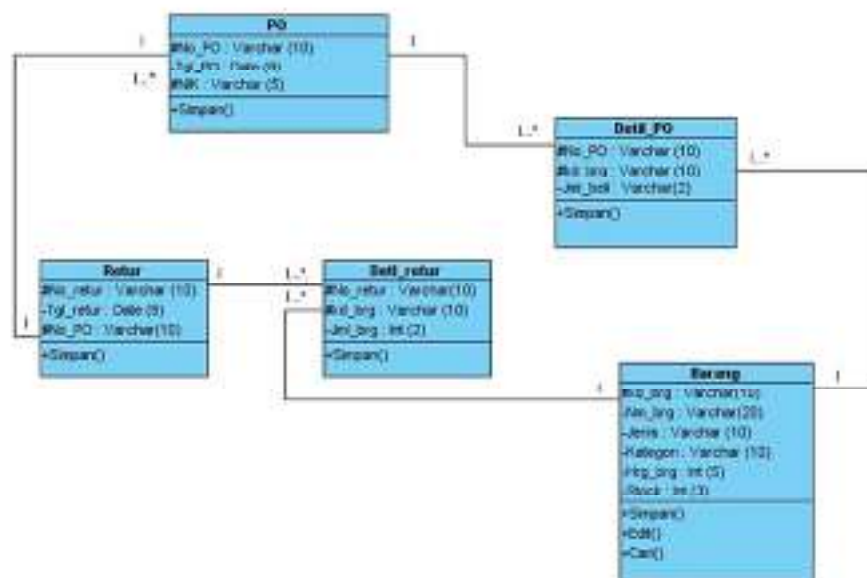
Tabel 2.2 *Simbol Class Diagram*

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

²² Sholiq, *Pemodelan Sistem Informasi Berorientasi Obyek dengan UML*(Yogyakarta: Graha Ilmu,2006)h.13

			mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri.
7	—	<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.

Contoh penggunaan *Class Diagram*:



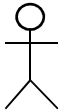
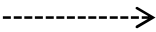
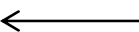
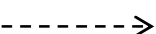
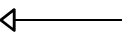

Gambar 2.5 Contoh penggunaan class diagram

2.11.3 Use Case Diagram


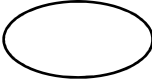


Use case diagram dapat dikatakan adalah sebuah abstraksi hubungan antara system dengan aktor yang menggunakan sistem itu sendiri. *Use case*

bekerja dengan menjalankan tipe interaksi antara user dengan sebuah sistem tentang bagaimana sistem tersebut dipakai²³.

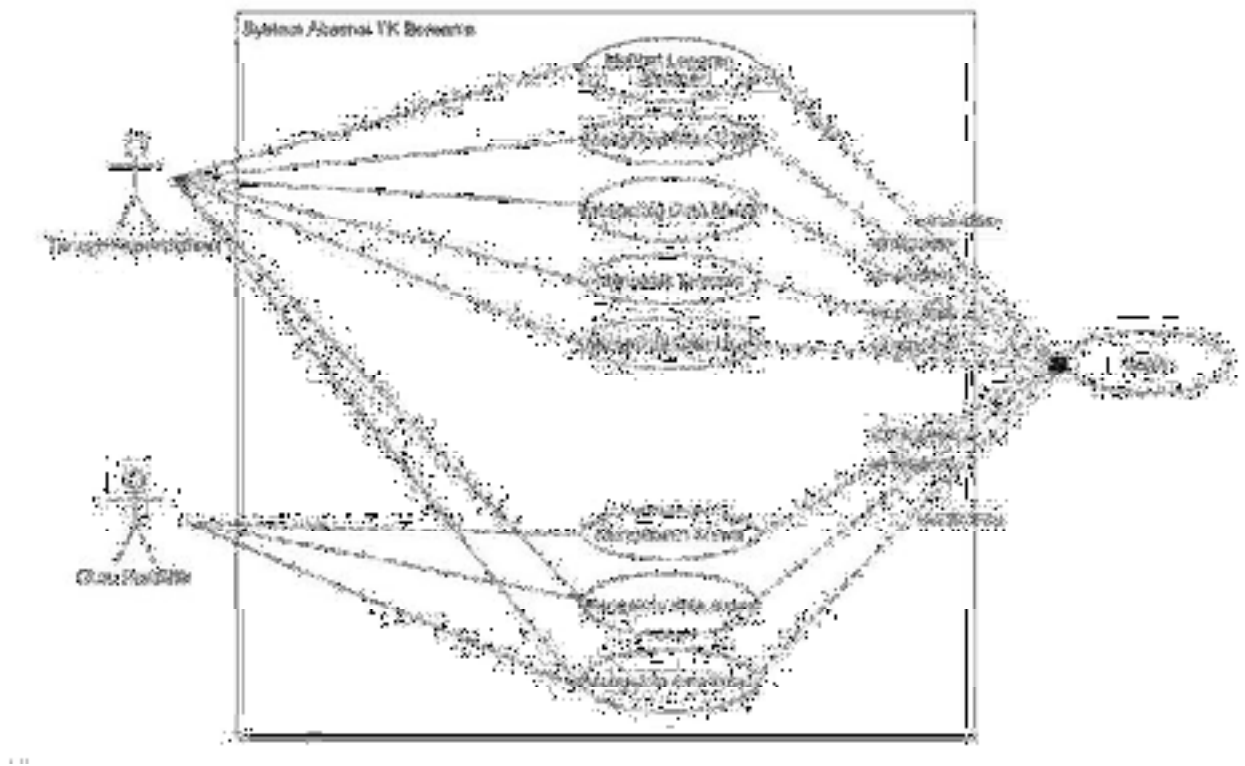
Tabel 2.3 Simbol Use Case Diagram

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna gunakan ketika berinteraksi dengan sistem.
2		<i>Dependency</i>	Hubungan dimana perubahan terjadi pada suatu element yang mandiri(<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (<i>not independent</i>)
3		<i>Generalization</i>	Hubungan dimana object anak (<i>descendent</i>) berbagi perilaku dan struktur data object yang ada di atasnya objek induk (<i>ancestor</i>).
4		<i>Include</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara <i>eksplisit</i> .
5		<i>Extend</i>	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.
6		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.

²³ Sholiq, *Pemodelan Sistem Informasi Berorientasi Obyek dengan UML*(Yogyakarta: Graha Ilmu,2006)h.7

7		<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
8		<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil terukur suatu aktor.
9		<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya(sinergi)
10		<i>Note</i>	Elemen Fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi.

Contoh penggunaan *Use Case Diagram*:



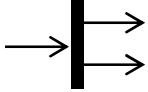
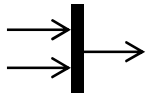



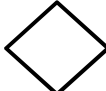
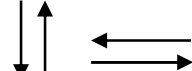
Gambar 2.6 Contoh penggunaan use case diagram

2.11.4 Activity Diagram

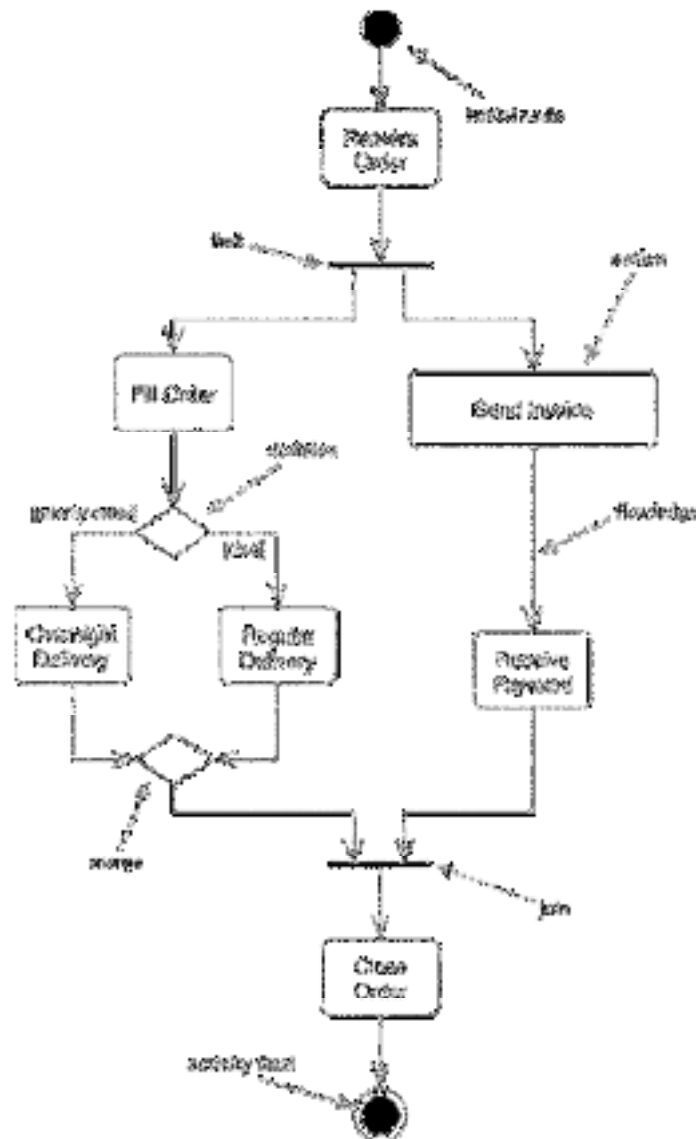
Diagram aktifitas adalah sebuah diagram yang menggambarkan bagaimana alur suatu sistem yang dirancang itu berjalan. Dalam diagram ini di sebutkan bagaimana awal sistem mulai bekerja dan berakhirnya sistem, selain itu dalam aktifitas diagram memungkinkan terjadinya alur sistem yang paralel²⁴.

Tabel 2.4 Simbol Activity Diagram

²⁴ Sholiq, *Pemodelan Sistem Informasi Berorientasi Obyek dengan UML*(Yogyakarta: Graha Ilmu, 2006)h.8-9

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Fork</i>	Percabangan
2		<i>Join</i>	Penggabungan
3		<i>Action</i>	State dari sistem yang mencerminkan eksekusi dari suatu aksi.
4		<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali.
5		<i>Activity Final Node</i>	Bagaimana objek dibentuk dan diakhiri.
6		<i>Decission</i>	Digunakan untuk menggambarkan suatu keputusan / tindakan yang harus diambil pada kondisi tertentu.
7		<i>Line Conector</i>	Digunakan untuk menghubungkan satu simbol ke simbol yang lain

Contoh penggunaan *Activity Diagram* :

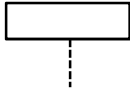
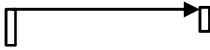
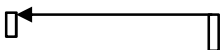


Gambar 2.7 Contoh penggunaan activity diagram

2.11.5 Sequence Diagram

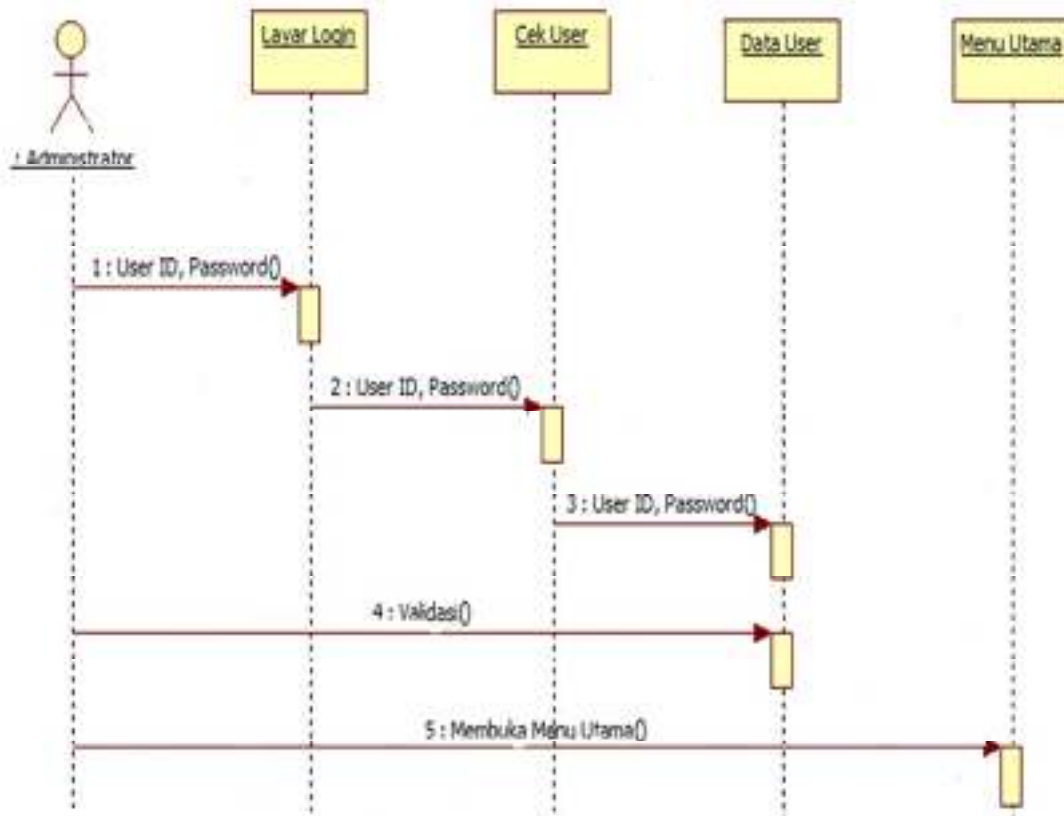
Sequence Diagram atau bisa disebut sebagai diagram urutan adalah suatu diagram yang menggambarkan skenario program berjalan dimulai dari level yang paling rendah hingga sampai ke level yang paling tinggi dalam sebuah sistem secara berurutan²⁵.

Tabel 2.7. Simbol Sequence Diagram

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

²⁵ Sholiq, *Pemodelan Sistem Informasi Berorientasi Obyek dengan UML*(Yogyakarta: Graha Ilmu,2006)h.9-11

Contoh penggunaan *Sequence Diagram*:



Gambar 2.8 Contoh penggunaan *Sequence Diagram*

BAB III

METODOLOGI PENELITIAN

3.1 Tujuan Dan Manfaat Penelitian

3.1.1 Tujuan Penelitian

Tujuan dari dilakukanya penelitian serta pembuatan Sistem Informasi E-Learning STMIK Muhammadiyah Jakarta ini adalah:

- a. Menciptakan rancangan secara terintegrasi Sistem Informasi E-Learning STMIK MJ dengan menggunakan rekayasa sistem metode SDLC (*System Development Life Cycle*) *waterfall* Model dan perangkat permodelan UML (*Unified Modelling Language*).
- b. Mengimplementasikan serta menerapkan sistem yang sebelumnya dibuat kedalam sistem aplikasi menggunakan bahasa pemrograman PHP *framework laravel* dan menggunakan *database* MySQL.

3.1.2 Manfaat Penelitian

Sedangkan manfaat yang akan didapatkan dari penelitian ini antara lain:

- a. Memberikan kontribusi terhadap ilmu pengetahuan khususnya pengetahuan sistem informasi akan pengetahuan-pengetahuan baru yang belum diketahui sebelumnya.
- b. Membantu pihak instansi yang digunakan untuk penelitian sebagai solusi masalah jangka panjang serta membantu pihak user yang berhubungan langsung dengan masalah tersebut.

3.2 Tempat Dan Waktu Penelitian

Tempat penelitian dilaksanakan pada STMIK Muhammadiyah Jakarta, pada Kampus Utama yang beralamat di Jalan Kelapa Dua Wetan no. 17, RT. 007/RW.004, Kec. Ciracas, Kota Jakarta Timur, Daerah Khusus Ibukota Jakarta 13730, dan Kampus A yang beralamat di Jl. Kh. Ahmad Dahlan No.20, RT.1/RW.9, Kayu Manis, Kec. Matraman, Kota Jakarta Timur, Daerah Khusus Ibukota Jakarta 13130. Untuk penelitian dilakukan dari bulan agustus sampai dengan september 2020.

3.2.1 Gambaran Umum Instansi Pendidikan

3.2.1.1 Sejarah

STMIK Muhammadiyah Jakarta berdiri pada tanggal 15 Jummadil Awwal 1413 H bertepatan dengan tanggal 10 November 1992 M sebagai perwujudan rapat panitia pendiri yang dibentuk dalam musyawarah Pimpinan Wilayah Muhammadiyah (PWM) DKI Jakarta tanggal 4 September 1992 yang kemudian dikukuhkan dengan Surat Keputusan (SK) PWM tanggal 10 November 1992 Nomor I.A/ SK/ 1149/ 1992.

Dalam perjalanannya, kepemimpinan STMIK Muhammadiyah Jakarta diamanahkan kepada:

- Dr. H Abuzar Asra, Msc, APU Menjabat Ketua STMIKMJ Periode 1993-1997
- Dr. Drs. Abdurrahman, SE, DPT Menjabat Ketua STMIKMJ Periode 1997-2001
- Dr. Ir. H M. Givi Efgivia, M Kom Menjabat Ketua STMIKMJ Periode 2001-2009

- Faiz Rafdhi, M. Kom Menjabat Ketua STMIKMJ Periode 2011-sekarang.

Pada tanggal 5 Juni 1995 STMIK Muhammadiyah Jakarta Jurusan Sistem Informasi (SI) dan Teknik Informatika (TI) memperoleh status Terdaftar dari Kementerian Pendidikan dan Kebudayaan Republik Indonesia melalui surat Keputusan Direktur Jendral Pendidikan Tinggi Nomor: 152/D/0/1995. Pada tanggal 25 Januari 2001 STMIKMJ Berhasil mempunyai Status Terakreditasi B oleh Badan Akreditasi Nasional Republik Indonesia melalui Surat Keputusan Nomor: 038/BAN-PT/Ak-IV/2001. Pada tanggal 10 Juli 2007 STMIKMJ berhasil mendapat status Terakreditasi dari Badan Akreditasi Nasional Perguruan Tinggi dengan Surat Keputusan Nomor: 015/BAN-PT/Ak-VII/2007 untuk Program Studi Sistem Informasi, dan pada tanggal 18 Agustus 2007 STMIKMJ berhasil mendapat status Terakreditasi dari Badan Akreditasi Nasional Perguruan Tinggi dengan Surat Keputusan Nomor: 018/BAN-PT/Ak-VIII/2007 untuk Program Studi Teknik Informatika.²⁶

3.3 Jadwal Penelitian

Aktifitas penelitian ini secara bertahap keseluruhan dilaksanakan mulai bulan september 2020 sampai dengan bulan november 2020.

²⁶ *Sejarah STMIK Muhammadiyah*, diakses: 10 juni 2019 pukul 19.20, dari: <https://www.stmikmj.ac.id/stmik-mj/sejarah-stmik-muhammadiyah-jakarta>

Tabel 3.1 *Time Schedule*

NO	KEGIATAN	SEPTEMBER				OKTOBER				NOVEMBER			
		I	II	III	IV	I	II	III	IV	I	II	III	IV
1	Pengumpulan dan Analisa Data												
2	Penulisan Laporan												
3	Asistensi BAB I												
4	Asistensi BAB II												
5	Asistensi BAB III												

3.4 Objek Penelitian

Penelitian yang dilakukan di STMIK Muhammadiyah Jakarta bertujuan untuk mengetahui bagaimana alur sistem yang berjalan. Sehingga peneliti dapat menggunakan hasil penelitiannya dalam merancang sistem informasi yang dibangun agar sesuai dengan apa yang diharapkan.

Penelitian ini juga menitikberatkan pada standar kurikulum yang diterapkan pada STMIK Muhammadiyah Jakarta. Sehingga seluruh materi pembelajaran dapat memenuhi kriteria kelayakan berdasarkan seluruh kajian yang dilakukan.

3.5 Metode Pengumpulan Data

3.5.1 Observasi Dan Wawancara

Observasi adalah proses pengamatan dan pencatatan secara sistematis mengenai gejala-gejala yang diteliti. Observasi adalah salah satu metode pengumpulan data dengan mengamati atau meninjau secara cermat dan langsung di lokasi penelitian untuk mengetahui kondisi yang terjadi atau membuktikan kebenaran dari sebuah desain penelitian demi mengetahui sebuah alur sistem. Objek observasi dalam penelitian ini, adalah kegiatan perkuliahan

di dalam kelas maupun di luar kelas, guna mengetahui secara lebih dalam bagaimana suatu sistem pembelajaran dari segi akademi dan administrasi.

Berdasarkan pengalaman peneliti yang berkuliah di STMIK Muhammadiyah Jakarta, peneliti cukup mengetahui bagaimana sistem pembelajaran dari perspektif mahasiswa, namun belum mengetahui bagaimana materi yang akan diajarkan ditentukan oleh dosen, dan sistem administrasi mahasiswa. Sistem administrasi sendiri akan berpengaruh besar kepada rancangan produk yang akan dibangun, karena menentukan bagaimana *role* akan dibuat, siapa saja yang boleh menggunakan sistem informasi ini, dan bagaimana materi atau konten yang sesuai untuk ditampilkan, apakah harus sesuai dengan KRS atau tidak, dan masih banyak lagi yang berhubungan dengan sistem administrasi STMIK Muhammadiyah yang telah dibangun untuk diimplementasikan pula pada sistem informasi yang baru.

Sedangkan wawancara merupakan percakapan antara dua orang atau lebih dan berlangsung antara narasumber dan pewawancara. Tujuan dari wawancara adalah untuk mendapatkan informasi yang tepat dari narasumber yang terpercaya. Wawancara dilakukan dengan cara penyampaian sejumlah pertanyaan dari pewawancara kepada narasumber.

Dalam penelitian ini, peneliti akan melakukan wawancara kepada dosen pembimbing yang merupakan tindak lanjut dari apa yang telah diobservasi. Pedoman wawancara yang akan dilakukan sebagai berikut:

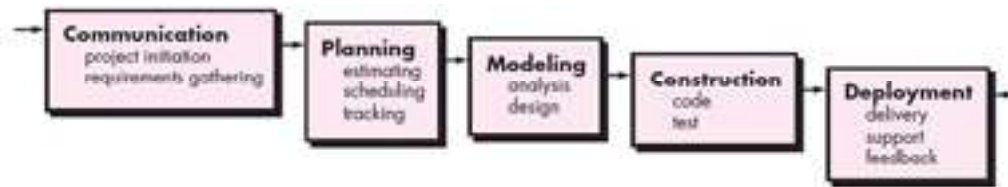
Tabel 3.2 *Form Wawancara*

PEDOMAN WAWANCARA		
Komponen Wawancara: <i>Input, Proses, Output</i> Nara Sumber : 1. Dosen Pembimbing 2. Staff Pengajar		
No	Komponen/Sub	Substansi Pertanyaan
1	Input Materi Pembelajaran	1. Bagaimana materi pembelajaran itu ditentukan? 2. Apakah materi pembelajaran pada sistem yang berjalan harus diimplementasikan juga ke dalam sistem informasi ini?
2	Input data user	1. Siapa saja yang boleh menggunakan sistem informasi ini selain mahasiswa dan dosen? 2. Perlukah mengimport seluruh database mahasiswa STMIK Muhammadiyah Jakarta untuk autentikasi login menggunakan NIM?

3.6 Metodologi Pengembangan Perangkat Lunak

3.6.1 *Waterfall Model*

Waterfall Model adalah model klasik yang bersifat sistematis, berurutan dalam membangun software. Waterfall Model terdiri dari 5 tahapan untuk pengembangan. Berikut adalah penjelasan dari tahap – tahap yang dilakukan di dalam model ini menurut Pressman:



Gambar 3.1 *Proses Penelitian Waterfall Model*

1. Communication (Project Initiation & Requirements Gathering)

Sebelum memulai pekerjaan yang bersifat teknis, sangat diperlukan adanya komunikasi dengan customer demi memahami dan mencapai tujuan yang ingin dicapai. Hasil dari komunikasi tersebut adalah inisialisasi proyek seperti menganalisis permasalahan yang dihadapi dan mengumpulkan data-data yang diperlukan, serta membantu mendefinisikan fitur dan fungsi dari aplikasi. Pengumpulan data-data tambahan bisa juga diambil dari jurnal, artikel, paper dan internet.

2. Planning (Estimating, Scheduling, Tracking)

Tahap berikutnya adalah tahapan perencanaan yang menjelaskan tentang estimasi tugas-tugas teknis yang akan dilakukan, resiko-resiko yang dapat terjadi, sumber daya yang diperlukan dalam membuat sistem, produk kerja yang ingin dihasilkan, penjadwalan kerja yang akan dilaksanakan, dan tracking proses pengerjaan sistem.

3. Modeling (Analysis & Design)

Tahapan ini adalah tahap perancangan dan pemodelan arsitektur sistem yang berfokus pada perancangan struktur data, arsitektur software, tampilan interface, dan algoritma program. Tujuannya untuk lebih memahami gambaran besar dari apa yang akan dikerjakan.

4. *Construction (Code & Test)*

Tahapan ini merupakan proses penerjemahan bentuk desain menjadi kode atau bentuk bahasa yang dapat dibaca oleh mesin. Setelah pengkodean selesai, dilakukan pengujian terhadap sistem dan juga kode yang sudah dibuat. Tujuannya untuk menemukan kesalahan yang mungkin terjadi untuk nantinya diperbaiki.

5. *Deployment (Delivery, Support, Feedback)*

Tahapan terakhir ini merupakan tahapan implementasi software ke customer, perbaikan software, evaluasi software, dan pengembangan software berdasarkan umpan balik yang diberikan agar sistem dapat tetap berjalan dan berkembang sesuai dengan fungsinya.

Dengan demikian, metode waterfall dianggap pendekatan yang lebih cocok digunakan untuk proyek pembuatan sistem baru dan juga pengembangan software dengan tingkat resiko yang kecil serta waktu pengembangan yang cukup lama. Tetapi salah satu kelemahan paling mendasar adalah menyamakan pengembangan hardware dan software dengan meniadakan perubahan saat pengembangan. Padahal, error diketahui saat software dijalankan, dan perubahan-perubahan akan sering terjadi²⁷.

3.6.2 Kelebihan dan kekurangan *Waterfall Model*

3.6.2.1 Kelebihan:

Kelebihan dari metode SDLC Model *Waterfall* adalah:

1. Prosesnya lebih terstruktur, hal ini membuat kualitas software baik dan tetap terjaga.

²⁷ Roger R. Pressman, *Rekayasa Perangkat Lunak*, (Yogyakarta: Andi, 2015)hal. 42-43

2. Dari sisi user juga lebih menguntungkan, karena dapat merencanakan dan menyiapkan kebutuhan data dan proses yang diperlukan sejak awal.
3. Penjadwalan juga menjadi lebih menentu, karena jadwal setiap proses dapat ditentukan secara pasti.
4. Model ini merupakan jenis model yang bersifat dokumen lengkap sehingga proses pemeliharaan dapat dilakukan dengan mudah.

3.6.2.2 Kekurangan:

Sedangkan untuk kekurangannya antara lain:

1. Bersifat kaku, sehingga sulit melakukan perubahan di tengah proses. Jika terdapat kekurangan proses/prosedur dari tahap sebelumnya, maka tahapan pengembangan harus dilakukan mulai dari awal lagi.
2. Membutuhkan daftar kebutuhan yang lengkap sejak awal.

BAB IV

HASIL DAN PEMBAHASAN

4.1 Analisa Sistem (*System Analysis*)

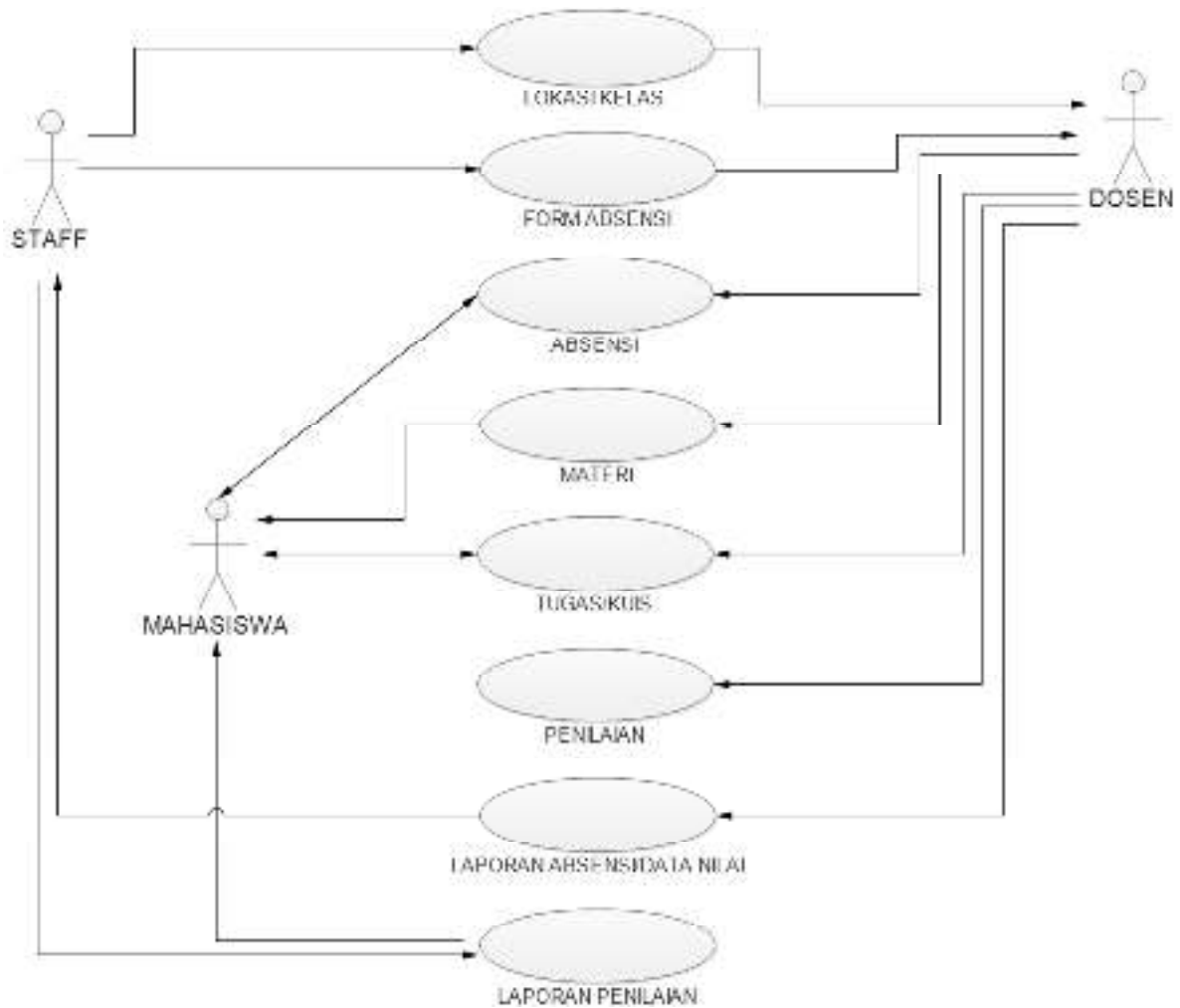
Dalam bab empat ini, dilakukan pembahasan yang mengacu pada identifikasi masalah yang ada. Permasalahan yang diangkat dalam penulisan skripsi ini adalah bagaimana membuat sistem informasi controlling sertifikasi awak pesawat yang bisa memudahkan staff controlling dalam mengontrol sertifikasi para aircrew.

4.1.1 Analisa Sistem Yang Berjalan

STMIK Muhammadiyah Jakarta menerapkan metode konvensional yaitu tatap muka dalam sistem perkuliahannya. Seluruh proses kegiatan belajar dilakukan oleh 2(dua) aktor yaitu Dosen dan Mahasiswa di dalam ruangan kelas, dengan Admin sebagai penentu lokasi kelas.

Berikut adalah sistem perkuliahan STMIK Muhammadiyah Jakarta yang disajikan dengan menggunakan *UML model* :

4.1.1.1 Use Case Diagram Sistem Perkuliahan



Gambar 4.1 Use Case Diagram Sistem Perkuliahan
STMIK Muhammadiyah Jakarta

Gambar 4.1 menjelaskan tentang peran setiap aktor terhadap sistem yang berjalan. Berikut penjelasannya:

Deskripsi Use Case (Lokasi Kelas)

<p>Aktor Utama: Staff</p> <p>Aktor Tambahan: Dosen, Mahasiswa</p> <p>Nama Use Case: Lokasi Kelas</p> <p>Tujuan Use Case: Menentukan kelas mana yang dapat digunakan untuk tiap sesi mata kuliah dan menginfokan kepada dosen dan mahasiswa agar menggunakan kelas yang ditentukan.</p> <p>Kondisi Awal: Lokasi kelas belum ditentukan. Mahasiswa belum mengetahui kelas mana yang harus dihadiri.</p> <p>Kondisi Akhir: Dosen dan mahasiswa melakukan perkuliahan pada kelas yang telah ditentukan.</p>	<p>Optimistic Flow:</p> <ol style="list-style-type: none"> 1. Staff melakukan survey terhadap kelas mana yang dapat digunakan untuk sebuah sesi mata kuliah. 2. Staff mengambil keputusan berdasarkan status pada tiap kelas. 3. Staff menginfokan kepada dosen dan mahasiswa bersangkutan untuk menggunakan kelas yang ditentukan. 4. Dosen dan mahasiswa melakukan perkuliahan pada kelas tersebut. <p>Pesimistic Flow:</p> <ol style="list-style-type: none"> 1. Tidak ada kelas kosong tersedia atau ada kendala pada ruangan kelas yang tersedia.
---	---

Deskripsi Use Case (Form Absensi)

<p>Aktor Utama: Staff</p> <p>Aktor Tambahan: Dosen</p> <p>Nama Use Case: Form Absensi</p> <p>Tujuan Use Case: Menentukan kelas mana yang dapat digunakan untuk tiap sesi mata kuliah dan menginfokan kepada dosen dan mahasiswa agar menggunakan kelas yang ditentukan.</p> <p>Kondisi Awal: Data form absensi belum terisi.</p> <p>Kondisi Akhir: Data form absensi sudah terisi.</p>	<p>Optimistic Flow:</p> <ol style="list-style-type: none"> 1. Staff memberikan form absensi kepada dosen. 2. Dosen mengisi informasi umum pada form seperti tanggal, bulan, tahun, dan mata kuliah. Dosen juga melakukan tanda tangan pada form absensi. 3. Dosen membawa form tersebut ke dalam kelas sebagai media pengisian data kehadiran mahasiswa.
--	--

Deskripsi Use Case (Absensi)

<p>Aktor Utama: Dosen, Mahasiswa</p> <p>Aktor Tambahan: -</p> <p>Nama Use Case: Lokasi Absensi</p> <p>Tujuan Use Case: Melakukan proses absensi demi mendapatkan data kehadiran mahasiswa.</p> <p>Kondisi Awal: Dosen menyebutkan nama mahasiswa satu per satu.</p> <p>Kondisi Akhir: Mahasiswa memberikan <i>feedback</i> kepada dosen menunjukkan kehadirannya.</p>	<p>Optimistic Flow:</p> <ol style="list-style-type: none"> 1. Staff memberikan form absensi kepada dosen. 2. Dosen mengisi informasi umum pada form seperti tanggal, bulan, tahun, dan mata kuliah. Dosen juga melakukan tanda tangan pada form absensi. 3. Dosen membawa form tersebut ke dalam kelas sebagai media pengisian data kehadiran mahasiswa.
---	--

Deskripsi Use Case (Materi)

<p>Aktor Utama: Dosen</p> <p>Aktor Tambahan: Mahasiswa</p> <p>Nama Use Case: Materi</p> <p>Tujuan Use Case: Melakukan penyampaian materi kuliah oleh dosen kepada mahasiswa.</p> <p>Kondisi Awal: Adanya proses pembukaan mata kuliah, dilanjutkan dengan proses penyampaian materi.</p> <p>Kondisi Akhir: Adanya proses Tanya jawab, diskusi kelompok dan penjelasan mengenai pertanyaan yang diajukan.</p>	<p>Optimistic Flow:</p> <ol style="list-style-type: none"> 1. Staff memberikan form absensi kepada dosen. 2. Dosen mengisi informasi umum pada form seperti tanggal, bulan, tahun, dan mata kuliah. Dosen juga melakukan tanda tangan pada form absensi. 3. Dosen membawa form tersebut ke dalam kelas sebagai media pengisian data kehadiran mahasiswa.
--	--

Deskripsi Use Case (Tugas/Kuis)

<p>Aktor Utama: Dosen, Mahasiswa</p> <p>Aktor Tambahan: -</p> <p>Nama Use Case: Tugas/Kuis</p> <p>Tujuan Use Case: Dosen memberikan tugas maupun kuis kepada mahasiswa, dan mahasiswa memberikan <i>feedback</i> terhadap apa yang diberikan.</p> <p>Kondisi Awal: Dosen memberikan tugas atau kuis kepada mahasiswa.</p> <p>Kondisi Akhir: Mahasiswa menyerahkan tugas atau menjawab kuis yang telah diberikan.</p>	<p>Optimistic Flow:</p> <ol style="list-style-type: none"> 1. Dosen memberikan tugas maupun kuis kepada mahasiswa berdasarkan materi yang telah disampaikan. 2. Mahasiswa mengerjakan atau menjawab tugas/kuis. 3. Mahasiswa menyerahkan kembali tugas/kuis yang telah dikerjakan sebagai bahan penilaian.
--	--

Deskripsi Use Case (Penilaian)

<p>Aktor Utama: Dosen</p> <p>Aktor Tambahan: -</p> <p>Nama Use Case: Penilaian</p> <p>Tujuan Use Case: Mengarsipkan data nilai mahasiswa berdasarkan absensi, tugas, dan beberapa aspek penilaian lainnya.</p> <p>Kondisi Awal: Dosen melakukan penilaian kepada mahasiswa.</p> <p>Kondisi Akhir: Data nilai diarsipkan.</p>	<p>Optimistic Flow:</p> <ol style="list-style-type: none"> 1. Dosen melakukan pemasukkan data nilai dalam sebuah arsip berdasarkan aspek penilaian yang telah ditentukan.
--	---

Deskripsi Use Case (Laporan absensi/data nilai)

<p>Aktor Utama: Dosen</p> <p>Aktor Tambahan: Staff</p> <p>Nama Use Case: Laporan Absensi/data nilai</p> <p>Tujuan Use Case: Melakukan proses komputerisasi terhadap data absensi dan nilai yang masih berbentuk <i>hard copy</i>.</p> <p>Kondisi Awal: Dosen memberikan data kepada staff di dalam sebuah form agar dapat didokumentasikan ke dalam komputer.</p> <p>Kondisi Akhir: Staff memasukkan data ke dalam <i>software</i> computer agar dapat diolah kembali.</p>	<p>Optimistic Flow:</p> <ol style="list-style-type: none"> 1. Dosen melakukan pemasukkan data nilai dalam sebuah arsip berdasarkan aspek penilaian yang telah ditentukan.
--	---

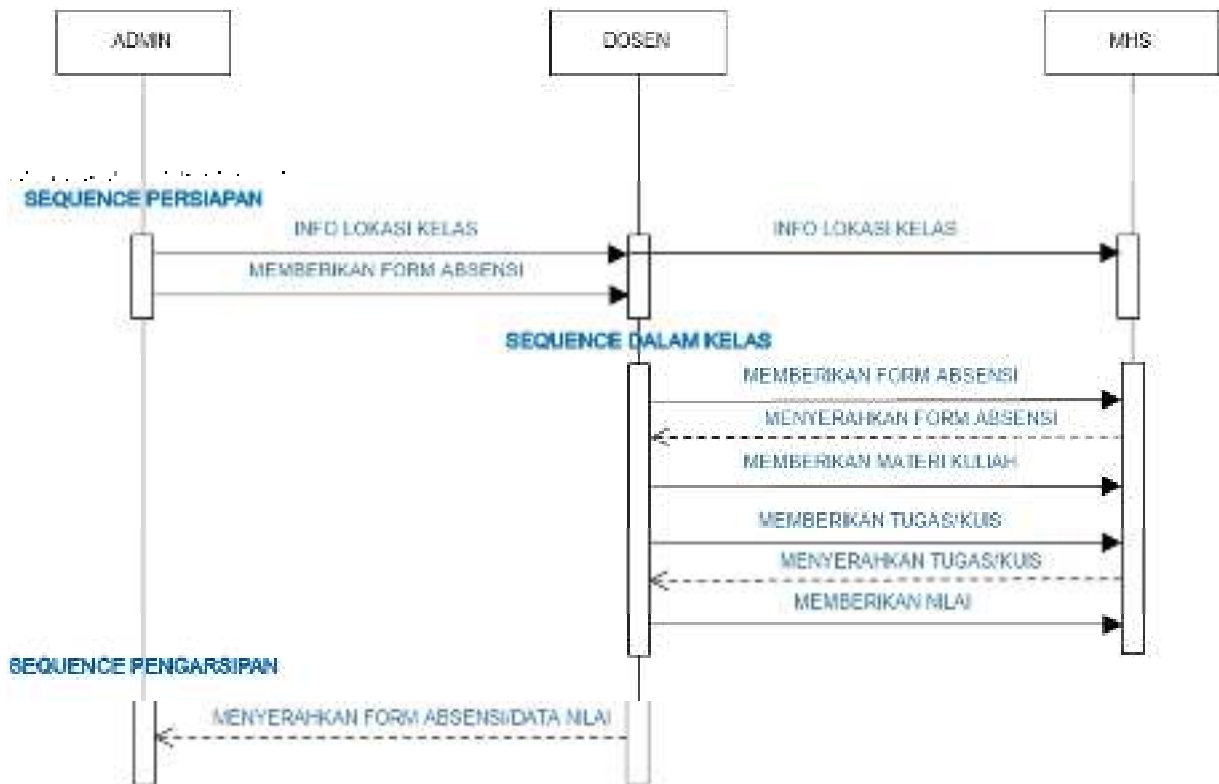
Deskripsi Use Case (Laporan penilaian)

<p>Aktor Utama: Staff</p> <p>Aktor Tambahan: Mahasiswa</p> <p>Nama Use Case: Laporan Penilaian</p> <p>Tujuan Use Case: Memberikan informasi penilaian kepada mahasiswa berdasarkan data nilai yang telah diolah sebagai bentuk laporan belajar.</p> <p>Kondisi Awal: Data yang telah diolah melalui proses komputerisasi, di <i>print out</i> dan siap untuk diserahkan kepada mahasiswa.</p> <p>Kondisi Akhir: Hasil <i>print out</i> diserahkan kepada mahasiswa sebagai laporan belajar.</p>	<p>Optimistic Flow:</p> <ol style="list-style-type: none"> 1. Staff melakukan <i>print out</i> data penilaian mahasiswa. 2. Staff menyerahkan hasil <i>print out</i> kepada mahasiswa
---	--

sistem perkuliahan STMIK Muhammadiyah Jakarta. Berikut adalah penjelasan bagaimana aktivitas itu dilakukan:

1. Pada awal persiapan sebelum dosen dan mahasiswa memasuki ruang kelas, staff bertugas untuk menentukan ruangan kelas yang akan digunakan dalam proses belajar. Staff juga memberikan form absensi kepada Dosen.
2. Setelah kelas ditentukan, Dosen dan Mahasiswa diwajibkan untuk memasuki kelas yang ditentukan.
3. Saat Dosen dan seluruh Mahasiswa siap untuk belajar, Dosen terlebih dahulu memberikan form absensi untuk ditandatangani oleh Mahasiswa. Kemudian form tersebut dikembalikan lagi kepada Dosen.
4. Kemudian Dosen memberikan materi kuliah, dan Mahasiswa bertugas untuk mendengarkan, mencatat, dan diperbolehkan bertanya tentang materi yang telah diberikan.
5. Pada umumnya Dosen melakukan sesi Tanya jawab atau memberikan tugas setelah materi diberikan, dan seluruh Mahasiswa dianggap tidak ada yang bertanya lagi. Kewajiban bagi Mahasiswa untuk menjawab pertanyaan yang diberikan maupun tugas, karena Dosen memiliki wewenang untuk memberikan nilai lewat media tersebut.
6. Saat jam perkuliahan selesai, Dosen memberikan kembali form absensi, untuk diarsipkan ke dalam computer.

4.1.1.3 Sequence Diagram Sistem Perkuliahan



Gambar 4.3 Sequence Diagram Sistem Perkuliahan
STMIK Muhammadiyah Jakarta

Gambar 4.3 menjelaskan tentang aktivitas yang dikelompokkan ke dalam sebuah *sequence*. Penjelasannya adalah sebagai berikut:

Deskripsi Sequence (Persipan)

Nama Activity.	Aktor	Optimistic Flow
Info Lokasi Kelas	Aktor Utama: Staff Aktor Tambahan: Dosen, Mahasiswa	1. Staff melakukan survey dan menginfokan Lokasi kelas yang dapat digunakan untuk proses perkuliahan kepada dosen. 2. Dosen membagikan info tersebut kepada mahasiswa agar mahasiswa dapat hadir pada kelas yang ditentukan Staff.
Memberikan Form Absensi	Aktor Utama: Staff Aktor Tambahan: Dosen	1. Staff memberikan form absensi kepada dosen agar dapat diisi.

Deskripsi Sequence (Dalam Kelas)

Nama Activity.	Aktor	Optimistic Flow
Memberikan Form Absensi Feedback: Menyerahkan Form Absensi	Aktor Utama: Dosen, Mahasiswa	<ol style="list-style-type: none"> 1. Dosen memberikan form absensi untuk ditandatangani oleh mahasiswa. 2. Mahasiswa menandatangani form absensi yang diberikan. 3. Mahasiswa menyerahkan kembali form absensi kepada dosen.
Memberikan Materi Kuliah	Aktor Utama: Dosen Aktor Tambahan: Mahasiswa	<ol style="list-style-type: none"> 1. Dosen memberikan materi kuliah kepada mahasiswa 2. <i>Feedback</i> dapat berupa pertanyaan dari mahasiswa kepada dosen.
Memberikan Tugas/Kuis Feedback: Menyerahkan Tugas/Kuis	Aktor Utama: Dosen, Mahasiswa	<ol style="list-style-type: none"> 1. Dosen memberikan tugas atau kuis kepada mahasiswa berdasarkan materi kuliah yang telah

		<p>disampaikan.</p> <p>2. Mahasiswa mengerjakan tugas atau menjawab kuis yang diberikan oleh dosen.</p> <p>3. Mahasiswa menyerahkan tugas atau kuis yang telah mereka kerjakan.</p>
Memberikan Nilai	<p>Aktor Utama: Dosen</p> <p>Aktor Tambahan: Mahasiswa</p>	<p>1. Dosen memberikan nilai dan melakukan arsip terhadap nilai tersebut.</p>

Deskripsi Sequence (Pengarsipan)

Nama Activity.	Aktor	Optimistic Flow
Menyerahkan form absensi/data nilai	Aktor Utama: Dosen Aktor Tambahan: Staff	3. Dosen menyerahkan hasil absensi atau data nilai kepada staff, untuk dimasukkan ke dalam computer dan diolah menjadi sebuah laporan.

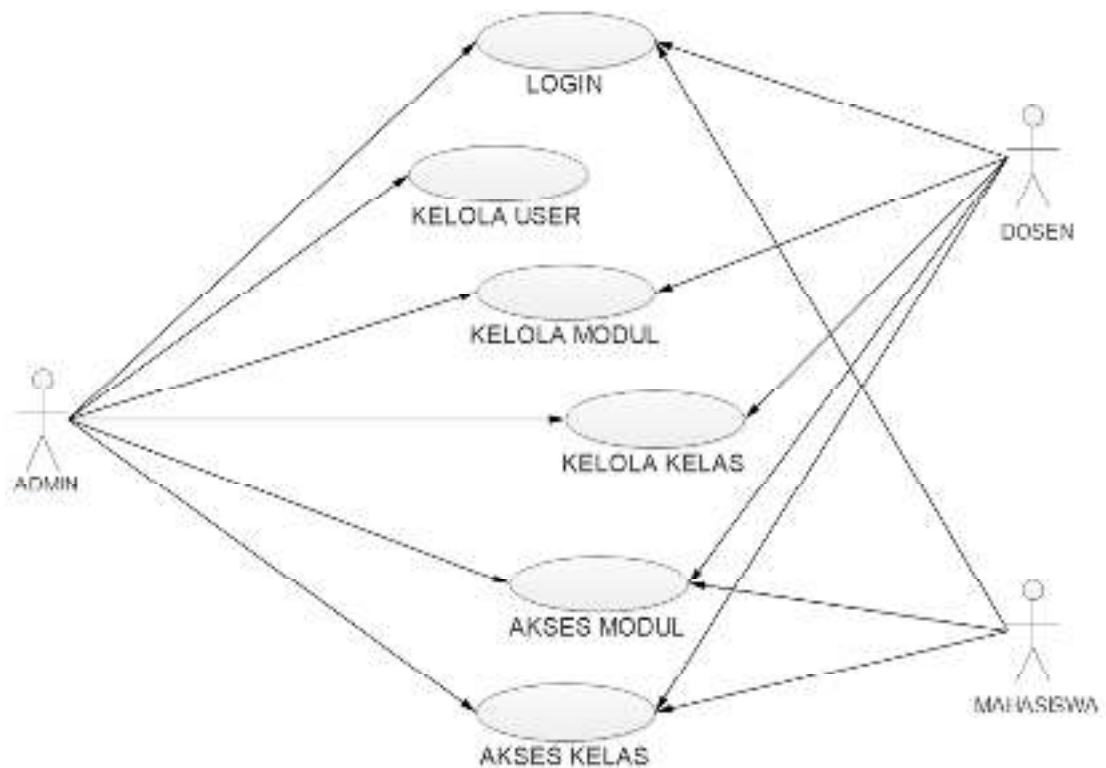
4.1.1.4 Identifikasi Masalah Sistem Yang Berjalan

Berikut adalah identifikasi kekurangan dari sistem yang berjalan pada STMIK Muhammadiyah saat ini:

1. Kegiatan belajar mengajar antara mahasiswa dan dosen masih dilakukan dengan cara konvensional, tanpa memiliki sistem alternative lain. Hal ini akan menyulitkan sebagian mahasiswa yang memiliki kewajiban bekerja.
2. Belum ada tempat penyimpanan yang mengintegrasikan seluruh bahan ajar dosen agar dapat di akses kembali oleh mahasiswa, layaknya repository seluruh modul dan materi.
3. Adanya kemungkinan mahasiswa tidak mendapatkan materi apabila dosen tidak masuk perkuliahan, hal ini sangat merugikan bagi mahasiswa yang mana wajib membayar sks untuk tiap mata kuliah.
4. Kegiatan belajar bagi mahasiswa tidak efektif dan efisien apabila materi yang dibagikan oleh dosen hanya melalui presentasi, catatan, maupun file. Karena ada kemungkinan materi tersebut tersimpan pada masing-masing device mahasiswa dalam kondisi acak.

4.1.2 Analisa Sistem Yang Diusulkan

4.1.2.1 Use Case Diagram Sistem Yang Diusulkan



Gambar 4.4 Use Case Diagram Sistem Yang Diusulkan

Gambar 4.4 menjelaskan tentang peran setiap aktor terhadap sistem yang diusulkan. Use Case di atas hanya menjelaskan tentang garis besar dari sistem yang diusulkan. Berikut penjelasannya:

Deskripsi Use Case (Login)

<p>Aktor Utama: Admin, Dosen, Mahasiswa</p> <p>Aktor Tambahan: -</p> <p>Nama Use Case: Login</p> <p>Tujuan Use Case: Sistem melakukan pengecekan credentials terhadap <i>user</i> yang masuk berdasarkan informasi pada pangkalan data.</p> <p>Kondisi Awal: Menampilkan halaman login pada website dengan 2(dua) input, yaitu email/username dan password.</p> <p>Kondisi Akhir: Success: <i>User</i> akan dialihkan pada halaman beranda user. Failure: Ada notifikasi yang menampilkan pesan error, berdasarkan kesalahan yang dialami.</p>	<p>Optimistic Flow:</p> <ol style="list-style-type: none"> 1. <i>User</i> memasukkan identitas user pada kolom tersedia. 2. <i>User</i> melakukan submit pada website. 3. <i>User</i> akan dialihkan ke halaman beranda. <p>Pesimistic Flow:</p> <ol style="list-style-type: none"> 1. <i>User</i> memasukkan <i>credentials</i> yang tidak dikenali oleh website. 2. Muncul notifikasi error berdasarkan kesalahan yang dialami.
--	--

Deskripsi Use Case (Kelola User)

<p>Aktor Utama: Admin</p> <p>Aktor Tambahan: -</p> <p>Nama Use Case: Kelola User</p> <p>Tujuan Use Case: Admin melakukan pengelolaan berupa tambah, edit, maupun delete terhadap user.</p> <p>Kondisi Awal: Admin mengakses menu <i>activities</i>. Diberikan dua jenis pilihan user yaitu, dosen dan mahasiswa.</p> <p>Kondisi Akhir: Admin akan dialihkan pada halaman user berdasarkan user yang dipilih.</p>	<p>Optimistic Flow:</p> <ol style="list-style-type: none"> 1. Admin melakukan pengelolaan data pada halaman user. 2. Pada halaman tersebut tersedia berbagai fitur seperti tambah, edit dan delete. <p>Pesimistic Flow:</p> <ol style="list-style-type: none"> 1. Terjadi kesalahan pada server sehingga laman tidak dapat diakses.
--	--

Deskripsi Use Case (Kelola Modul)

<p>Aktor Utama: Admin, Dosen</p> <p>Aktor Tambahan: -</p> <p>Nama Use Case: Kelola Modul</p> <p>Tujuan Use Case: Admin atau Dosen melakukan pengelolaan modul, berupa tambah, edit, delete dan publish.</p> <p>Kondisi Awal: Admin atau Dosen menekan menu modul. Kemudian akan dialihkan pada laman administrasi modul.</p> <p>Kondisi Akhir: Admin atau Dosen akan dialihkan pada laman administrasi modul.</p>	<p>Optimistic Flow:</p> <ol style="list-style-type: none"> 1. Website menampilkan laman administrasi modul. 2. Admin atau dosen melakukan pengelolaan modul pada laman modul. 3. Pada halaman tersebut tersedia berbagai fitur seperti tambah, edit, delete dan publish. <p>Pesimistic Flow:</p> <ol style="list-style-type: none"> 1. Terjadi kesalahan pada server sehingga laman tidak dapat diakses.
---	--

Deskripsi Use Case (Kelola Kelas)

<p>Aktor Utama: Admin, Dosen</p> <p>Aktor Tambahan: -</p> <p>Nama Use Case: Kelola User</p> <p>Tujuan Use Case: Admin atau Dosen melakukan pengelolaan kelas, berupa tambah, edit, delete dan publish.</p> <p>Kondisi Awal: Admin atau Dosen menekan menu modul. Kemudian akan dialihkan pada laman administrasi kelas.</p> <p>Kondisi Akhir: Admin atau Dosen akan dialihkan pada laman administrasi kelas.</p>	<p>Optimistic Flow:</p> <ol style="list-style-type: none"> 1. Website menampilkan laman administrasi kelas. 2. Admin atau dosen melakukan pengelolaan kelas pada laman kelas. 3. Pada halaman tersebut tersedia berbagai fitur seperti tambah, edit, delete dan publish. <p>Pesimistic Flow:</p> <ol style="list-style-type: none"> 1. Terjadi kesalahan pada server sehingga laman tidak dapat diakses.
--	--

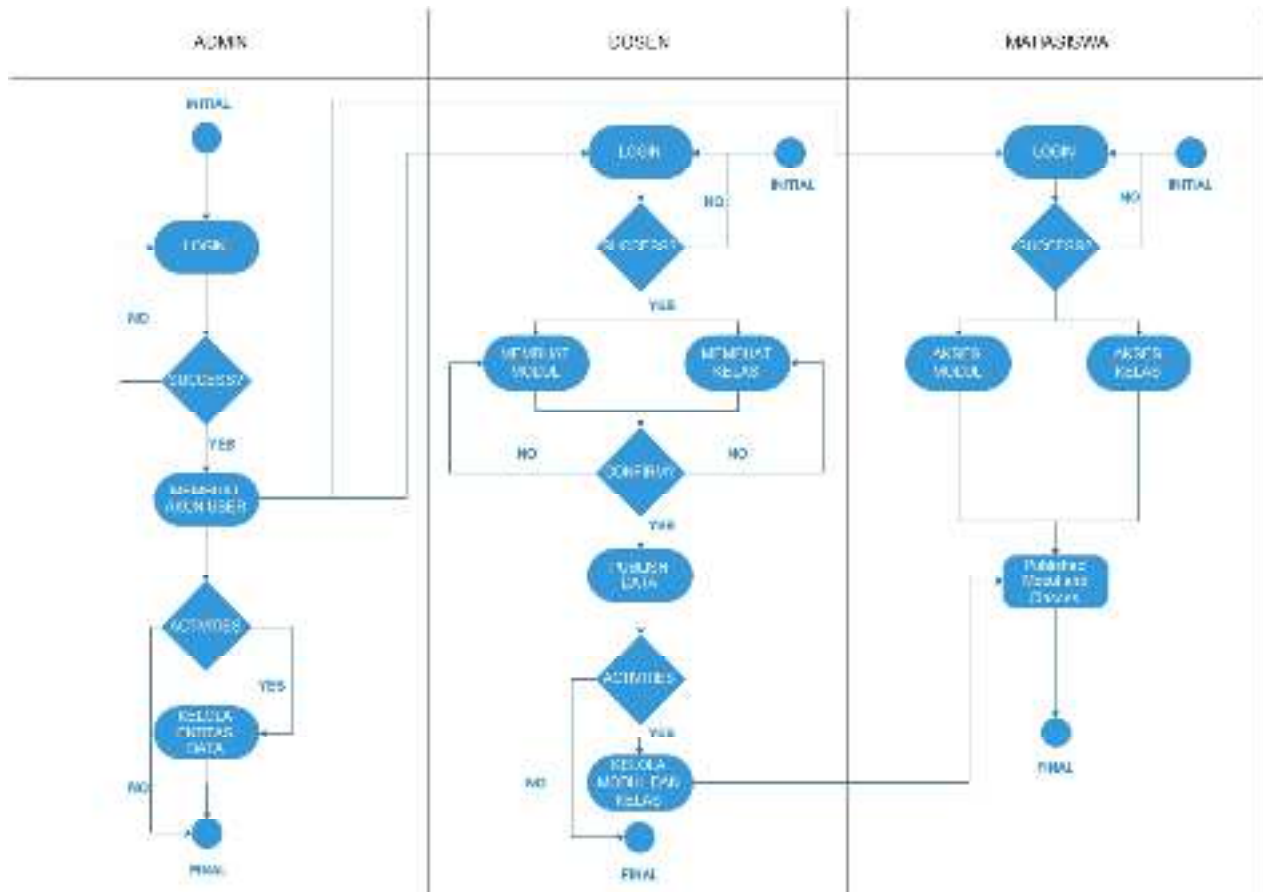
Deskripsi Use Case (Akses Modul)

<p>Aktor Utama: Admin, Dosen, Mahasiswa</p> <p>Aktor Tambahan: -</p> <p>Nama Use Case: Akses Modul</p> <p>Tujuan Use Case: Admin, Dosen, maupun mahasiswa melakukan interaksi terhadap website pada laman modul. Interaksi yang dilakukan adalah membaca modul-modul yang telah dipublish sebelumnya.</p> <p>Kondisi Awal: Pada laman beranda, admin, dosen maupun mahasiswa dapat menekan link yang mengarahkan mereka pada detail lengkap modul yang diinginkan.</p> <p>Kondisi Akhir: Akan muncul laman detail modul yang diklik sebelumnya. <i>User</i> dapat membaca atau <i>me-review</i> modul yang telah dipublish</p>	<p>Optimistic Flow:</p> <ol style="list-style-type: none"> 1. Website menampilkan laman detail modul. 2. Seluruh aktor dapat membaca seluruh isi modul. <p>Pesimistic Flow:</p> <ol style="list-style-type: none"> 1. Terjadi kesalahan pada server sehingga laman tidak dapat diakses.
--	--

Deskripsi Use Case (Akses Kelas)

<p>Aktor Utama: Admin, Dosen, Mahasiswa</p> <p>Aktor Tambahan: -</p> <p>Nama Use Case: Akses Kelas</p> <p>Tujuan Use Case: Admin, Dosen, maupun mahasiswa melakukan interaksi terhadap website pada laman kelas. Interaksi yang dilakukan dapat berupa absen, <i>chatting</i>, <i>posting</i>, dan <i>upload</i> berkas.</p> <p>Kondisi Awal: Aktor dapat menekan link kelas yang akan mengarahkan mereka pada daftar kelas yang mereka ikuti. Dengan menekan link pada daftar kelas, maka aktor/user akan dialihkan pada detail kelas yang telah diklik sebelumnya.</p> <p>Kondisi Akhir: Akan muncul laman detail modul yang diklik sebelumnya. <i>User</i> dapat membaca atau <i>me-review</i> modul yang telah dipublish</p>	<p>Optimistic Flow:</p> <ol style="list-style-type: none"> 1. Website menampilkan laman detail kelas. 2. Seluruh aktor dapat melakukan absen, <i>posting</i>, <i>chatting</i>, dan <i>upload</i> berkas pada laman detail kelas. <p>Pesimistic Flow:</p> <ol style="list-style-type: none"> 1. Terjadi kesalahan pada server sehingga laman tidak dapat diakses.
--	---

4.1.2.2 Activity Diagram Sistem Yang Diusulkan



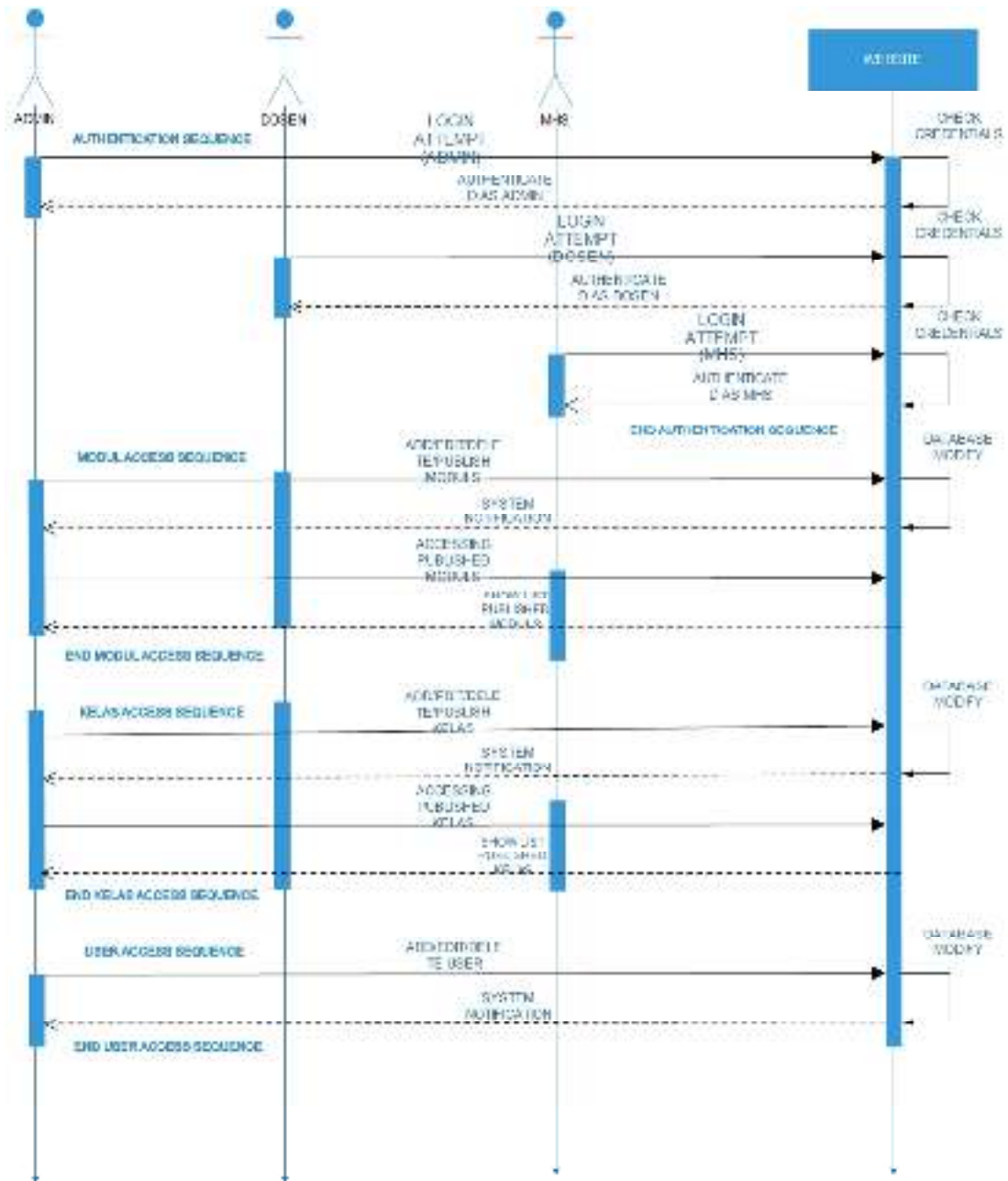
Gambar 4.5 Activity Diagram Sistem Yang Diusulkan

Gambar 4.5 menjelaskan alur sistem yang diusulkan. Alur tersebut dapat dijabarkan sebagai berikut:

1. Admin melakukan autentikasi melalui proses login. Data admin sendiri telah di *seeding* oleh Back-end Developer sebagai entitas awal.
2. Setelah berhasil login, admin dapat membuat user baru dengan 2(dua) *role* pilihan, yaitu Dosen dan Mahasiswa.

3. Admin diberikan pilihan untuk menjalankan activities atau tidak. Arti activities disini adalah menu administrative yang tersedia pada aplikasi, meliputi *CRUD*.
4. Setelah User berhasil dibuat, Dosen maupun Mahasiswadapat menggunakan informasi user yang dibutuhkan untuk login, yaitu email/*username* dan *password*.
5. Ada perbedaan *user interface* pada website tergantung *role* apa yang terautentikasi. Dosen memiliki *user interface* yang memungkinkan untuk membuat modul dan kelas, sedangkan mahasiswa tidak bisa membuat keduanya.
6. Dosen juga diberikan pilihan untuk mengelola modul dan kelas yang telah mereka buat. Admin juga diberikan otoritas ini.
7. Pada sistem ini, Admin memiliki wewenang tertinggi diikuti Dosen dan Mahasiswa.

4.1.2.3 Sequence Diagram Sistem Yang Diusulkan



Gambar 4.6 Sequence Diagram Sistem Yang Diusulkan

Gambar 4.6 menjelaskan pokok-pokok alur sistem yang dikelompokkan ke dalam beberapa *sequence*. Alur tersebut dapat dijabarkan sebagai berikut:

Deskripsi Sequence (Authentication)

Nama Activity.	Aktor	Optimistic Flow
Login Attempt (ADMIN)	<p>Aktor Utama: Admin</p> <p>Aktor Tambahan: -</p>	<ol style="list-style-type: none"> 1. Admin memasukkan username/email dan password untuk melakukan autentikasi. 2. Setelah melakukan submit, <i>back-end website</i> akan melakukan pengecekan <i>credentials</i> terhadap input user. 3. Apabila sukses, website akan mengautentikasi user dengan role admin ke dalam website.
Login Attempt	Aktor Utama:	1. Dosen

(DOSEN)	<p>Dosen</p> <p>Aktor Tambahan:</p> <p>-</p>	<p>memasukkan username/email dan password untuk melakukan autentikasi.</p> <p>2. Setelah melakukan submit, <i>back-end website</i> akan melakukan pengecekan <i>credentials</i> terhadap input user.</p> <p>3. Apabila sukses, website akan mengautentikasi user dengan role dosen ke dalam website.</p>
Login Attempt (MHS)	<p>Aktor Utama:</p> <p>Mahasiswa</p> <p>Aktor Tambahan:</p> <p>-</p>	<p>1. Mahasiswa memasukkan username/email dan password untuk melakukan autentikasi.</p> <p>2. Setelah melakukan submit, <i>back-end website</i> akan</p>

		<p>melakukan pengecekan <i>credentials</i> terhadap input user.</p> <p>3. Apabila sukses, website akan mengautentikasi user dengan role mahasiswa ke dalam website.</p>
--	--	---

Deskripsi Sequence (Modul Access)

Nama Activity.	Aktor	Optimistic Flow
Add/ Edit/ Delete/ Publish Modul	Aktor Utama: Admin, Dosen Aktor Tambahan: -	1. Admin dan Dosen mengakses menu modul. 2. Pada menu tersebut diberikan beberapa fitur yang meliputi <i>CRUD</i> dan publish untuk modul. 3. Saat admin maupun dosen hendak menggunakan fitur tersebut, sistem secara otomatis memberikan notifikasi terhadap status proses pada aplikasi. 4. Apabila sukses, user akan diberikan pesan yang menunjukkan berhasilnya perubahan data. Data pada database

		juga telah dimanipulasi pada proses ini.
Accessing Published Modul	Aktor Utama: Admin, Dosen, Mahasiswa Aktor Tambahan: -	<ol style="list-style-type: none"> 1. Seluruh role user dapat mengakses modul pada beranda. 2. Apabila link modul pada beranda diklik, user akan <i>redirect</i> pada laman detail modul. 3. Modul-modul yang ada pada beranda, adalah modul-modul yang telah <i>publish</i> oleh admin maupun dosen.

Deskripsi Sequence (Kelas Access)

Nama Activity.	Aktor	Optimistic Flow
Add/ Edit/ Delete/ Publish Kelas	Aktor Utama: Admin, Dosen Aktor Tambahan: -	<ol style="list-style-type: none"> 1. Admin dan Dosen mengakses menu kelas. 2. Pada menu tersebut diberikan beberapa fitur yang meliputi <i>CRUD</i> dan publish untuk kelas. 3. Saat admin maupun dosen hendak menggunakan fitur tersebut, sistem secara otomatis memberikan notifikasi terhadap status proses pada aplikasi. 4. Apabila sukses, user akan diberikan pesan yang menunjukkan berhasilnya perubahan data. Data pada database

		juga telah dimanipulasi pada proses ini.
Accessing Published Kelas	Aktor Utama: Admin, Dosen, Mahasiswa Aktor Tambahan: -	<ol style="list-style-type: none"> 1. Seluruh role user dapat mengakses kelas yang telah mereka ikuti. 2. Pendaftaran kelas dapat dilakukan dengan mengakses menu gabung kelas. Menu ini tidak ada pada role admin, karena admin secara sistem dapat langsung mengikuti seluruh kelas yang telah dibuat oleh user. 3. Pada menu kelas, user dapat mengakses “detail kelas” dengan mengklik link pada kelas yang diinginkan. 4. Pada detail kelas

		terdapat berbagai fitur seperti absen, <i>comment</i> , <i>upload</i> , dan <i>posting</i> .
--	--	--

Deskripsi Sequence (User Access)

Nama Activity.	Aktor	Optimistic Flow
Add/ Edit/ Delete User	<p>Aktor Utama: Admin</p> <p>Aktor Tambahan: -</p>	<ol style="list-style-type: none"> 1. Admin mengakses menu dosen/mahasiswa pada <i>dropdown</i> “activities”. 2. Setelah memilih salah satu dari 2(dua) link tersebut, Admin akan dialihkan pada laman user tersebut. 3. Pada laman ini admin dapat melihat list data seluruh user yang terdaftar. 4. Pada halaman ini juga, admin dapat mengakses fitur CRUD yang tersedia. 5. Setelah melakukan submit pada salah satu fitur, sistem akan memberikan notifikasi terhadap

		<p>proses aplikasi.</p> <p>6. Apabila sukses, sistem akan menampilkan pesan “berhasil”.</p> <p>7. Apabila gagal, sistem akan menampilkan pesan <i>error</i> berdasarkan kesalahan yang dialami.</p>
--	--	---

4.2 Analisa Kebutuhan

4.2.1 Kebutuhan Hardware

Dalam penerapan rancangan sistem yang telah dijelaskan sebelumnya, dibutuhkan beberapa perangkat keras untuk melakukan *testing*. Adapun perangkat keras yang digunakan adalah:

- Perangkat keras yang akan digunakan dalam pembuatan sistem:

Pada proses perancangan sistem, dalam mayoritas waktunya, penulis menggunakan Laptop Acer e15-975g dengan spesifikasi sebagai berikut :

Tabel 4.1 Kebutuhan Hardware Dalam Perancangan

No.	Nama Hardware	Spesifikasi
1	Processor	Intel @i5-7200
2	RAM	4GB
3	HDD	1TB
4	VGA	NVIDIA GeForce 940MX

- Minimum Spesifikasi Perangkat keras yang digunakan dalam implementasi sistem:

Tabel 4.2 Minimum Spesifikasi Kebutuhan Hardware

No.	Nama Hardware	Min. Spesifikasi
1	Processor	Intel @Pentium 4
2	RAM	2GB
3	HDD	500GB
4	Monitor	-

4	Perangkat keras tambahan	-
---	--------------------------	---

4.2.2 Kebutuhan Software

- Perangkat lunak yang digunakan untuk pembuatan sistem:

Tabel 4.3 Perangkat Lunak Pembuatan Sistem

No.	Nama Software	Keterangan
1	Windows 10 64 bit	Sebagai Sistem Operasi
2	Visual Code Studio	Sebagai Text Editor
3	XAMPP	Sebagai Local Server
4	Google Chrome, Mozilla Firefox, Opera	Sebagai Web Browser
5	GitBash	Sebagai Terminal
6	GitLab	Sebagai Repository

- Perangkat lunak yang digunakan untuk pembuatan *user interface*:

Tabel 4.4 Perangkat Lunak Pembuatan Sistem

No.	Nama Software	Keterangan
1	Adobe Photoshop CS6	Sebagai Aplikasi Desain grafis dengan format pixel
2	Undraw.co	Website yang menyediakan <i>free open-source</i> desain dengan format .svg
3	Adobe Illustrator CS6	Sebagai aplikasi desain grafis dengan format vector

- Perangkat lunak yang digunakan untuk implementasi:

Tabel 4.5 Perangkat Lunak Implementasi Sistem

No.	Nama Software	Keterangan
1	Windows 10 64 bit	Sebagai Sistem Operasi
2	Google Chrome	Sebagai Web Browser
3	XAMPP	Sebagai Local Server

4.2.3 Kebutuhan Brainware

Kebutuhan Brainware adalah kebutuhan dimana aplikasi memerlukan operator untuk dapat berjalan, dan melakukan seluruh proses yang diinginkan. Kebutuhan Brainware memegang peranan penting dan dibutuhkan sumber daya manusia yang memenuhi kriteria karena sebuah kesalahan fatal dapat mengakibatkan seluruh sistem *down*. Sumber daya manusia yang dibutuhkan dalam pengoperasian *web* ini antara lain:

1. Sistem Analis

Analisis Sistem adalah penguraian dari suatu sistem informasi yang utuh ke dalam bagian-bagian komponennya. Maka Sistem analis berarti orang yang bertugas merancang, *men-design*, dan manajemen seluruh kebutuhan perangkat lunak, untuk dibuat sebagai sistem informasi secara utuh.

Kegagalan dalam memahami kebutuhan sistem adalah kesalahan fatal dalam *project* pengembangan *software*. Maka dari itu, sistem analis haruslah merupakan orang yang paham betul mengenai seluruh bagian vital dalam perancangan sistem. Dan pada umumnya, seorang sistem analis harus mengetahui tentang *data flow*, *schema*, *structure database*, bahkan sedikit tentang *coding*. Untuk aplikasi ini, sistem analisis diperlukan saat adanya perubahan fitur, relasi data dan lain sebagainya.

2. Web Developer

Bertugas melakukan penulisan kode-kode pemrograman dan melakukan perancangan program sesuai dengan arahan sistem analis. Karena aplikasi ini merupakan *web-based* maka diperlukan 2(dua) jenis kemampuan yang harus dikuasai, antara lain:

a. Front-end Developer

Tugas pokoknya melakukan perancangan desain *user-interface*. Tidak hanya itu, Front-end Developer juga harus memiliki koordinasi yang baik dengan Back-end Developer, karena akan bertanggung jawab menangani data binding dan request form.

b. Back-end Developer

Bertugas untuk menerima request dari front-end, melakukan strukturisasi database sesuai arahan sistem analis, membuat *controller* ataupun *endpoint* sesuai dengan kebutuhan sistem dan melakukan *seeding* atau bahasa mudahnya, melakukan input data awal ke database untuk user admin.

3. Admin

Pada sistem *E-Learning* STMIK Muhammadiyah Jakarta ini, tugas admin adalah sebagai berikut:

a. Membuat User

Tugas dari seorang admin adalah membuatkan user bagi pengguna aplikasi agar dapat digunakan untuk autentikasi ke

dalam aplikasi. Ada 2(dua) jenis User yang harus dibuatkan oleh Admin, yaitu:

1. Dosen

Dosen adalah user yang memiliki kewenangan untuk membuat, mengedit, dan menghapus modul dan kelas.

2. Mahasiswa

Mahasiswa adalah user yang hanya memiliki kewenangan untuk mengikuti modul dan kelas.

- b. Mengelola Data

Tugas lainnya dari admin adalah mengelola entitas data. Admin diberikan wewenang untuk melakukan *CRUD*, yaitu *Create, Read, Update* dan *Delete*.

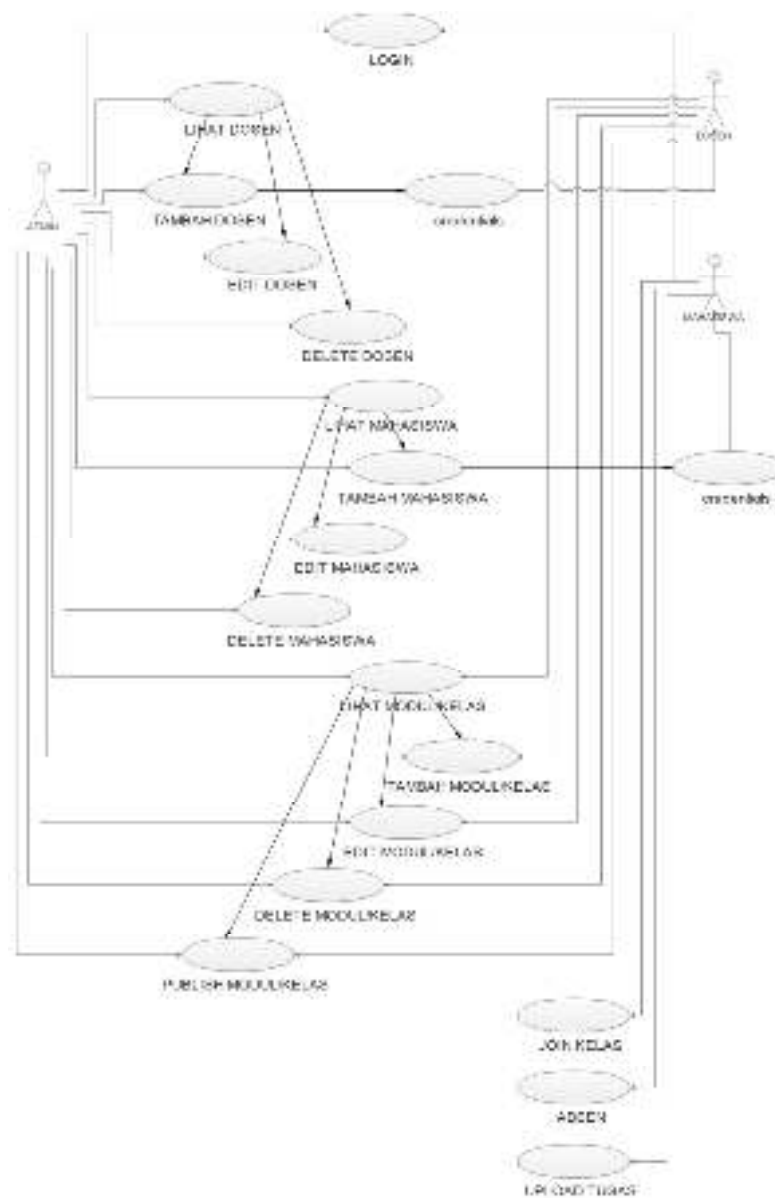
4. User

User atau pengguna, adalah orang yang melakukan interaksi di dalam aplikasi. User dibagi menjadi 3(tiga) *role*, yaitu; Admin, Dosen dan Mahasiswa.

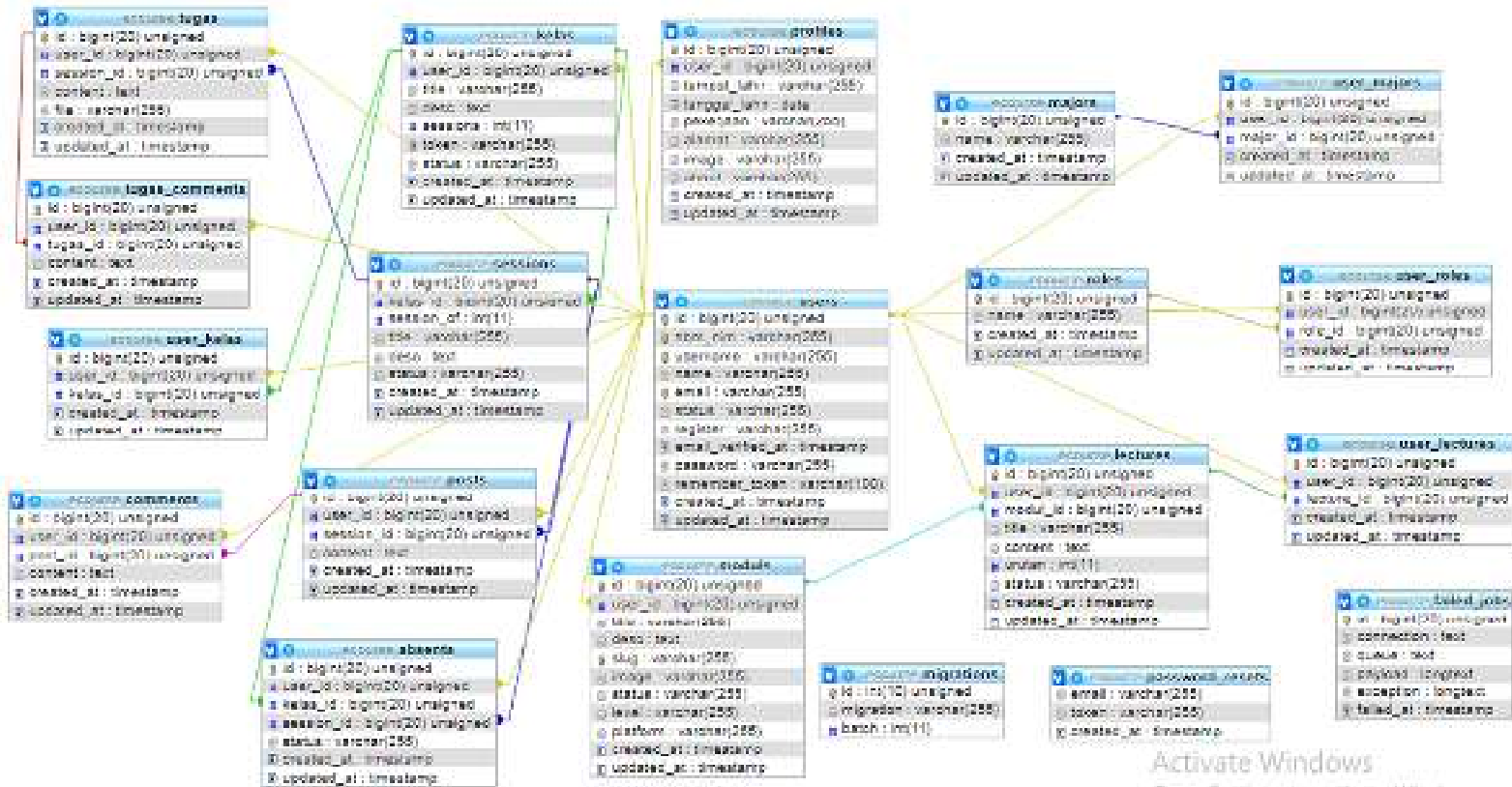
4.3 Desain Sistem (*System Design*)

Untuk mendesain sistem disini digunakan UML diagram yang meliputi *use case diagram*, *class diagram*, *sequence diagram* dan *activity diagram*.

4.3.1 Use Case Diagram Sistem



4.3.2 Diagram Class Sistem



DAFTAR PUSTAKA

- Abdul, Kadir. 2008. *Belajar Database Menggunakan MySQL*. Yogyakarta: ANDI OFFSET.
- Allen, Michael. 2013. *Michael Allen's Guide to E-learning*. Canada: John Wiley & Sons.
- Azhar, Susanto. 2013. *Sistem Informasi Akuntansi*. Bandung: Linggar Jaya.
- Harianto, Bambang. 2008. *Dasar informatika dan Komputer*. Yogyakarta: Graha Ilmu.
- Jogianto. 2004. *Analisis dan Desain Sistem Informasi*. Edisi II. Yogyakarta.
- Mulyadi. 2010. *Sistem Akuntansi*. Jakarta: Salemba Empat.
- Pressman, Roger R. 2015. *Rekayasa Perangkat Lunak*. Yogyakarta: Andi.
- Sholih. 2006. *Pemodelan Sistem Informasi Berorientasi Obyek dengan UML*. Yogyakarta: Graha Ilmu
- Sutanta, Edhy. 2003. *Sistem Informasi Manajemen*. Yogyakarta: Graha Ilmu.
- Sutarman. 2009. *Pengantar Teknologi Informasi*. Jakarta: Bumi Aksara.
- IDCloudHost. 2016. *Pengertian dan Keunggulan Framework Laravel*,
<https://idcloudhost.com/pengertian-dan-keunggulan-framework-laravel/> ,
 Diakses 7 Juni 2019 pukul 21.40.
- IDCloudHost. 2016. *Pengertian dan Manfaat Composer Bagi Developer*,
<https://idcloudhost.com/pengertian-dan-manfaat-composer-bagi-developer/> ,
 Diakses 13 Juni 2019 pukul 21.30.
- John, Dewey. 2016. *Teori E-Learning menurut Cendikiawan*,
<https://www.silabus.web.id/e-learning/> , Diakses 15 May 2019 pukul 17.40.

STMIK Muhammadiyah Jakarta. *Sejarah STMIK Muhammadiyah*,
<https://www.stmikmj.ac.id/stmik-mj/sejarah-stmik-muhammadiyah-jakarta> ,
diakses 10 juni 2019 pukul 19.20.