

**RANCANG BANGUN SISTEM INFORMASI MONITORING
KEGIATAN BELAJAR MENGAJAR MENGGUNAKAN *WEB
SERVICE* DI RUMPUN MATEMATIKA FMIPA UNJ**

Proposal Skripsi

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



**Oleh:
Aldi Rahmansyah
3145161324**

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

2021

LEMBAR PERSETUJUAN

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

Nama : Aldi Rahmansyah

No. Registrasi : 3145161324

Program Studi : Ilmu Komputer

Judul : Rancang Bangun Sistem Informasi Monitoring Kegiatan
Belajar Mengajar menggunakan *Web Service* Di Rumpun
Matematika FMIPA UNJ

Menyatakan bahwa proposal ini telah siap diajukan untuk seminar pra skripsi.

Menyetujui,

Dosen Pembimbing I

Dosen Pembimbing II

Med Irzal, M.Kom.

NIP. 19770615 200312 1 001

Ir. Fariani Hermin Indiyah, M.T.

NIP. 19600211 198703 2 001

Mengetahui,

Koordinator Program Studi Ilmu Komputer

Ir. Fariani Hermin Indiyah, M.T.

NIP. 19600211 198703 2 001

KATA PENGANTAR

Puji dan syukur penulis panjatkan kepada Allah SWT atas segala rahmat, berkah dan hidayah-Nya sehingga penulis dapat menyelesaikan proposal skripsi yang berjudul "Rancang Bangun Sistem Informasi Monitoring Kegiatan Belajar Mengajar Menggunakan *Web Service* Di Rumpun Matematika FMIPA UNJ" dengan baik. Proposal skripsi ini dibuat dengan tujuan untuk memenuhi salah satu syarat memperoleh gelar Sarjana Komputer.

Dalam proses pembuatan proposal skripsi ini, penulis mendapat banyak bantuan dan dorongan. Oleh karena itu, penulis ingin menyampaikan rasa terima kasih kepada Bapak Med Irzal, M.Kom selaku Dosen Pembimbing I dan Ibu Ir. Fariani Hermin Indiyah, M.T selaku Dosen Pembimbing II yang telah membimbing, mengarahkan, mengoreksi, serta memberikan berbagai saran dan masukan yang sangat bermanfaat dalam penyusunan proposal skripsi ini.

Dalam penyusunan proposal skripsi ini, penulis menyadari bahwa proposal skripsi yang telah dibuat ini masih sangat jauh dari kata sempurna dengan segala keterbatasan ilmu dan pengetahuan penulis sendiri. Baik dari segi penulisan, materi, dan bahasa. Oleh karena itu, penulis sangat membutuhkan kritik dan saran untuk dijadikan sebagai pembelajaran dan untuk membantu penulis menjadi lebih baik lagi kedepannya.

Akhir kata, penulis berharap agar proposal yang telah dibuat ini dapat memberikan manfaat kepada semua pihak, khususnya penulis sendiri. Penulis juga berharap ditulisnya proposal skripsi ini dapat menjadi motivasi dan memberikan semangat bagi rekan-rekan yang sedang dan akan menyusun proposal skripsi. Semoga Allah SWT senantiasa membalas kebaikan semua pihak yang telah membantu penulis dalam menyusun proposal skripsi ini.

Jakarta, 2 Juni 2021

Penulis

DAFTAR ISI

KATA PENGANTAR	iii
DAFTAR ISI	v
DAFTAR GAMBAR	vii
DAFTAR TABEL	vii
I PENDAHULUAN	1
1.1 Latar Belakang Masalah	1
1.2 Identifikasi Masalah	4
1.3 Pembatasan Masalah	4
1.4 Rumusan Masalah	5
1.5 Tujuan Penelitian	5
1.6 Manfaat Penelitian	5
II KAJIAN PUSTAKA	7
2.1 Monitoring dan Evaluasi	7
2.2 Penjaminan Mutu	9
2.3 Rencana Pembelajaran Semester	11
2.4 Sistem Informasi	12
2.4.1 Definisi Sistem	12
2.4.2 Definisi Informasi	13
2.4.3 Definisi Sistem Informasi	15
2.5 Software Development Life Cycle (SDLC)	16
2.6 <i>Unified Modeling Language</i> (UML)	19

2.7	<i>Entity Relationship Diagram</i>	22
2.8	<i>Database</i>	24
2.9	<i>Web Service</i>	26
2.10	<i>Framework</i>	27
2.11	<i>MVC</i>	28
III IMPLEMENTASI PROGRAM		30
3.1	Analisis Kebutuhan	30
3.2	Desain Sistem	31
3.2.1	<i>Use Case Diagram</i>	32
3.2.2	<i>Activity Diagram</i>	34
3.2.3	<i>Class Diagram</i>	39
3.2.4	<i>Entity Relationship Diagram</i>	40
3.2.5	Rancangan Tampilan Antarmuka Program	41
DAFTAR PUSTAKA		49
LAMPIRAN		50

DAFTAR GAMBAR

Gambar 2.1	Form 05	10
Gambar 2.2	Form 06	11
Gambar 2.3	Contoh Sederhana ERD	24
Gambar 2.4	Pola Desain MVC	29
Gambar 3.1	Desain <i>Use Case Diagram</i>	33
Gambar 3.2	<i>Activity</i> Mengunggah RPS dan Menambah Form 05	35
Gambar 3.3	<i>Activity</i> Membuat Tugas dan Mengunggah Nilai	36
Gambar 3.4	<i>Activity</i> Mengisi Presensi dan Memvalidasi Form 05	37
Gambar 3.5	<i>Activity</i> Memilih Penanggung Jawab Pengganti	38
Gambar 3.6	<i>Activity</i> Mengganti <i>role user</i>	38
Gambar 3.7	<i>Activity</i> Memvalidasi Presensi	39
Gambar 3.8	Desain Class Diagram	40
Gambar 3.9	Desain Entity Relationship Diagram	41
Gambar 3.10	Desain Halaman <i>Login</i>	42
Gambar 3.11	Desain Halaman <i>Dashboard</i> Pada mahasiswa	42
Gambar 3.12	Desain Halaman Rekap Presensi	43
Gambar 3.13	Desain Halaman <i>Monitoring</i>	44
Gambar 3.14	Desain Halaman Daftar Kelas	44
Gambar 3.15	Desain Halaman Form 05 Pada Dosen	45
Gambar 3.16	Desain Halaman Form 05 Pada Mahasiswa	45
Gambar 3.17	Desain Halaman Presensi	46
Gambar 3.18	Desain Halaman Tugas Pada Dosen	46
Gambar 3.19	Desain Halaman Tugas Pada Mahasiswa	47
Gambar 3.20	Desain Halaman Nilai Pada Dosen	47

DAFTAR TABEL

Tabel 2.1	Simbol-Simbol Pada <i>Use Case Diagram</i>	20
Tabel 2.2	<i>Multiplicity</i> atau <i>Cardinality</i> pada <i>Class Diagram</i>	21
Tabel 2.3	Simbol-Simbol Pada <i>Class Diagram</i>	21
Tabel 2.4	Simbol-Simbol Pada <i>Activity Diagram</i>	22

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Tri Dharma Perguruan Tinggi adalah tiga pilar dasar pemikiran yang harus ada pada semua aspek di dalam sebuah perguruan tinggi mulai dari mahasiswa, dosen dan berbagai civitas akademika yang terlibat. Tri Dharma Perguruan Tinggi terdiri dari 3 poin, yaitu Pendidikan dan Pengajaran, Penelitian dan Pengembangan, dan Pengabdian Kepada Masyarakat.

Kegiatan belajar mengajar adalah suatu kegiatan yang merupakan bagian dari salah satu isi Tri Dharma Perguruan Tinggi yang merupakan pendidikan dan pengajaran. Kegiatan belajar mengajar dijalankan oleh dua pihak, yaitu dosen dan mahasiswa. Keduanya merupakan faktor yang sangat berpengaruh dalam berhasilnya suatu kegiatan belajar mengajar. Dosen dan mahasiswa merupakan komponen penting dalam suatu sistem pembelajaran di perguruan tinggi dimana peran, tugas, dan tanggung jawab seorang dosen terutama dalam sebuah proses pembelajaran merupakan hal yang sangat penting dalam mewujudkan tujuan dari satu pilar Tri Dharma Perguruan Tinggi. Dalam melaksanakan kegiatan belajar mengajar perlu dilakukan pemantauan dan evaluasi untuk memastikan pelaksanaan kegiatan belajar mengajar mencapai tujuan yang diinginkan dan sesuai dengan standar yang ditetapkan oleh perguruan tinggi.

Monitoring merupakan kegiatan pemantauan yang dilakukan untuk mengetahui proses kegiatan Belajar Mengajar yang dilakukan oleh dosen. Sedangkan evaluasi merupakan hasil akhir dari monitoring yang dilakukan selama proses Kegiatan Belajar Mengajar yang telah dilaksanakan selama satu semester. Monitoring dan evaluasi kegiatan belajar mengajar adalah tugas dari Tim Penjamin Mutu. Kegiatan ini dilaku-

kan dengan tujuan untuk memastikan Kegiatan Belajar Mengajar yang disampaikan oleh dosen menerapkan aturan dan standar yang telah diterapkan oleh perguruan tinggi serta sesuai dengan Rencana Pembelajaran Semester (RPS).

Seiring berkembangnya teknologi, sistem informasi dapat mempermudah berbagai kegiatan untuk menghasilkan informasi sebagai penunjang pengambilan keputusan dan mempermudah penyelesaian suatu masalah dan meningkatkan kinerja berbagai aktivitas. Penggunaan sistem informasi juga dapat dimanfaatkan pada berbagai macam aktivitas akademik pada sebuah perguruan tinggi seperti aktivitas kegiatan belajar mengajar beserta monitoring dan evaluasinya.

Pada Fakultas Matematika dan Ilmu Pengetahuan Alam (FMIPA) UNJ, kegiatan belajar mengajar dipantau melalui formulir 05 dan 06 oleh Tim Penjamin Mutu tingkat Program Studi yang selanjutnya diserahkan kepada Gugus Penjamin Mutu FMIPA UNJ. Dalam menjalankan kegiatan belajar mengajar, mahasiswa akan mengambil formulir 05 dan 06 dari ruang program studi setiap perkuliahan akan dimulai dan akan dikembalikan ke ruang program studi ketika perkuliahan selesai. Formulir 05 dan 06 akan dievaluasi pada awal, pertengahan dan akhir semester oleh Tim Penjamin Mutu program studi. Evaluasi pada awal semester dilakukan untuk memantau jalannya perkuliahan pertama dan melihat apakah perkuliahan berjalan sesuai dengan jadwal akademik. Evaluasi kedua dilakukan setelah pertemuan ke-8 yang merupakan pertemuan Ujian Tengah Semester (UTS). Evaluasi tersebut dilakukan untuk memastikan semua pertemuan sebelumnya sudah lengkap sebelum melakukan UTS. Evaluasi terakhir dilakukan setelah pertemuan ke-16 dan dilakukan untuk melihat jumlah seluruh pertemuan mencapai target minimal, yaitu 80% pertemuan. Namun karena sistem yang masih manual dan formulir yang menggunakan kertas, proses evaluasi memakan waktu yang cukup lama karena pihak Tim Penjamin Mutu harus mengecek setiap formulir 05 dan 06.

Formulir 05 dan 06 merupakan formulir yang berisi tentang bagaimana kegiatan belajar mengajar dilaksanakan oleh seorang dosen dan mahasiswa. Dimana formulir 05 berisi tentang materi yang disampaikan oleh dosen dan formulir 06 berisi presensi mahasiswa dan nilai-nilai yang diberikan oleh dosen kepada mahasiswa. Nilai-nilai tersebut terdiri dari nilai-nilai tugas, Ujian Tengah Semester (UTS), dan Ujian Akhir Semester (UAS). Setelah adanya SIAKAD (Sistem Informasi Akademik) UNJ, dosen-dosen tidak lagi mengisi kolom nilai pada formulir 06 melainkan langsung memasukkan nilai akhir di SIAKAD pada saat akhir semester. Tidak diisinya kolom nilai pada formulir 06 menyebabkan transparansi nilai yang berkurang untuk pihak mahasiswa.

Hasil penelitian yang telah dilakukan oleh Andini et al. (2017) dalam jurnal yang berjudul "Perancangan dan Implementasi Sistem Absensi *Online* Berbasis Android di Lingkungan Universitas Negeri Jakarta" menunjukkan bahwa sistem presensi online dapat diterapkan menjadi salah satu cara agar proses presensi mahasiswa dapat berlangsung secara cepat dan membuat data presensi menjadi semakin terstruktur. Pada penelitian lain oleh Kultsum (2021) yang melakukan penelitian berjudul "Rancang Bangun Sistem Presensi Akademik Berbasis Web Dengan *Framework* Laravel di Lingkungan Program Studi Ilmu Komputer Universitas Negeri Jakarta" menunjukkan bahwa penggunaan aplikasi web memiliki beberapa kelebihan seperti kemudahan akses, kemudahan perawatan, dan kebutuhan perangkat keras yang lebih rendah dan tetap dapat diterapkan sebagai opsi untuk mengurangi masalah yang terjadi pada sistem manual.

Pada penelitian ini, penulis mengembangkan dari penelitian-penelitian sebelumnya yang hanya berupa sistem presensi dengan tambahan fitur monitoring untuk mempermudah tugas Tim Penjamin Mutu, kelengkapan form 06 berupa pengisian nilai untuk memberikan transparansi nilai kepada mahasiswa dan membuat data

nilai menjadi lebih terstruktur. Penulis memilih membuat sistem berbasis *website* agar dapat lebih mudah diakses tanpa membedakan *device* yang digunakan. Dalam pembuatan sistem informasi ini, penulis menggunakan metode Spiral sebagai metode pengembangannya karena penerapannya yang cukup mudah dan merupakan metode yang fleksibel jika terjadi perubahan pada sistem.

Dengan adanya sebuah sistem informasi monitoring yang dapat membantu proses monitoring kegiatan belajar mengajar akan sangat memudahkan Tim Penjamin Mutu untuk melakukan monitor dan evaluasi kegiatan belajar mengajar tanpa memakan waktu yang cukup lama, mempermudah pengisian form 05 dan 06 pada perkuliahan dengan tidak menggunakan form manual berbentuk kertas dan memberikan mahasiswa transparansi nilai yang diberikan oleh dosen.

1.2 Identifikasi Masalah

Berdasarkan latar belakang di atas, masalah yang akan diidentifikasi adalah sebagai berikut:

1. Pengisian form 05 dan 06 dalam proses pembelajaran masih manual, sehingga monitoring yang dilakukan kurang efisien karena masih melihat kertas form satu persatu.
2. Kurangnya transparansi nilai yang diberikan dosen kepada mahasiswa.

1.3 Pembatasan Masalah

Pada perancangan sistem ini, penulis membatasi masalah sebagai berikut:

1. Sistem ini hanya dibuat untuk digunakan pada Rumpun Matematika FMIPA UNJ.
2. Implementasi sistem di jaringan lokal.

1.4 Rumusan Masalah

Berdasarkan uraian pada latar belakang yang diutarakan di atas, maka perumusan masalah pada penelitian ini adalah “Bagaimana merancang suatu sistem informasi monitoring kegiatan belajar mengajar berbasis *website* di Rumpun Matematika FMIPA UNJ?”.

1.5 Tujuan Penelitian

Tujuan pada penelitian ini yaitu:

1. Memudahkan Tim Penjamin Mutu program studi melakukan monitoring kegiatan belajar mengajar.
2. Mempermudahkan evaluasi kinerja dosen pada akhir perkuliahan secara objektif.

1.6 Manfaat Penelitian

Hasil penelitian ini diharapkan dapat memberikan manfaat bagi berbagai pihak, di antaranya:

1. Bagi Mahasiswa

Sebagai suatu media untuk memudahkan memantau nilai yang didapat pada setiap mata kuliah secara transparan.

2. Bagi Dosen

Mampu mempermudah dosen untuk melakukan pengisian form 05 dan 06 dan memantau kehadiran mahasiswanya.

3. Bagi Tim Penjamin Mutu

Mempermudah monitoring kegiatan belajar mengajar untuk melakukan evaluasi kinerja dosen.

4. Bagi Program Studi

Mempermudah evaluasi kehadiran dosen dan mahasiswa dalam rangka menyusun borang akreditasi program studi.

BAB II

KAJIAN PUSTAKA

2.1 Monitoring dan Evaluasi

Monitoring adalah sebuah kegiatan mengamati secara seksama suatu keadaan atau kondisi, termasuk perilaku atau kegiatan tertentu, dengan tujuan agar semua data atau informasi masukan yang didapat dari hasil pengamatan tersebut dapat dijadikan landasan dalam pengambilan keputusan tindakan yang perlu dilakukan selanjutnya.

Tindakan tersebut diperlukan seandainya dari hasil pengamatan didapatkan adanya hal atau kondisi yang tidak sesuai dengan rencana atau aturan yang dibuat sebelumnya. Monitoring dilaksanakan dengan maksud agar kegiatan yang dilakukan dapat mencapai tujuan yang diinginkan secara efektif dan efisien.

Proses dasar dalam monitoring meliputi tiga tahap, yaitu menetapkan standar pelaksanaan, pengukuran pelaksanaan, dan menentukan perbedaan antara pelaksanaan dengan aturan dan rencana awal. Monitoring memiliki empat fungsi, yaitu: (Dunn, 2014)

- Ketaatan (compliance). Monitoring menentukan apakah tindakan setiap bagian anggota yang terlibat sesuai dengan norma, nilai, dan standar yang ditetapkan dalam menjalankan tugasnya.
- Pemeriksaan (auditing). Monitoring menetapkan apakah fasilitas dan layanan yang diperuntukkan bagi suatu pihak benar-benar tersampaikan dengan baik.
- Laporan (accounting). Monitoring menghasilkan informasi dan data untuk membantu melihat hasil perubahan sosial sebagai akibat implementasi keputusan sesudah beberapa periode waktu.

- Penjelasan (explanation). Monitoring menghasilkan informasi yang dapat membantu menjelaskan bagaimana efek diterapkannya sebuah kebijakan dan menjelaskan alasan kecocokan perencanaan dan pelaksanaannya.

Monitoring dan evaluasi memiliki kaitan yang sangat erat, karena dalam evaluasi dibutuhkan hasil monitoring untuk melihat pengaruh dari program yang berjalan.

Evaluasi merupakan proses yang sistematis dan berkelanjutan untuk mengumpulkan, mendeskripsikan, menginterpretasikan, dan menyajikan informasi untuk dapat digunakan sebagai dasar membuat keputusan, menyusun kebijakan maupun menyusun program selanjutnya (Widoyoko, 2009). Evaluasi menggunakan informasi yang dimiliki setelah dilakukannya monitoring dalam pengambilan keputusan untuk menilai suatu objek, keadaan, peristiwa atau kegiatan tertentu. Evaluasi adalah tindakan penilaian atas program yang telah dilakukan dan membandingkan apakah jalannya program tersebut sesuai dengan rencana dan tujuan yang dibuat. Hasil dari evaluasi berupa suatu kebijakan yang akan diterapkan untuk menjalankan program program selanjutnya dengan harapan jalannya program tersebut akan lebih baik lagi. Berikut adalah tujuan-tujuan dilakukannya sebuah evaluasi, yaitu: (Wirawan, 2011)

1. Menentukan kesesuaian objek evaluasi dengan rencana awal yang dibuat. Menilai apakah objek evaluasi telah dilaksanakan sesuai rencana.
2. Mengukur kesesuaian pelaksanaan dari objek evaluasi dengan standar yang telah dibuat.
3. Mengidentifikasi dan menentukan kekurangan dari objek evaluasi.
4. Pengembangan pengguna dari objek yang dievaluasi.
5. Mengambil keputusan mengenai objek yang dievaluasi.
6. Akuntabilitas

7. Memberikan saran kepada user.
8. Mengembangkan teori evaluasi dan riset evaluasi.

2.2 Penjaminan Mutu

Dalam Undang-undang No.12 Tahun 2012 Pasal 7 Ayat (3) huruf c mengatur tentang pendidikan tinggi mengatur tentang penjaminan mutu suatu pendidikan tinggi yaitu salah satu tugas dan wewenang menteri atas penyelenggaraan pendidikan tinggi meliputi peningkatan penjaminan mutu, relevansi, keterjangkauan, pemerataan yang berkeadilan dan akses pendidikan tinggi yang berkelanjutan. Pada undang-undang yang sama penjaminan mutu dibagi menjadi lima bagian yaitu bagian pertama sistem penjaminan mutu, bagian kedua standar pendidikan tinggi, bagian ketiga akreditasi, bagian keempat pangkalan data pendidikan tinggi, dan bagian kelima lembaga layanan pendidikan tinggi (UU No 12 Tahun 2012).

Sistem penjaminan mutu pendidikan tinggi adalah sebuah aktivitas yang dilakukan secara terus menerus dengan tujuan meningkatkan kualitas pendidikan menjadi terencana dan berkelanjutan. Mutu pendidikan itu sendiri adalah tingkat kesesuaian kenyataan pelaksanaan pendidikan di lapangan dengan aturan dan standar pendidikan yang ditetapkan oleh perguruan tinggi itu sendiri.

Penjaminan mutu di UNJ dilakukan oleh Lembaga Penjamin Mutu UNJ atau biasa disebut LPjM. LPjM telah berdiri sejak tanggal 26 maret 2006 berdasarkan Surat Keputusan Rektor Universitas Negeri Jakarta nomor 293/SP/2006 (FMIPA, 2016). Lembaga ini menjalankan sistem penjaminan mutu pada tingkat universitas, sedangkan pada tingkat fakultas dijalankan oleh Gugus Penjamin Mutu atau biasa disingkat GPjM dan untuk tingkat program studi oleh Tim Penjamin Mutu atau TPjM.

Penjaminan mutu di UNJ dilakukan dengan cara memonitoring form 05 dan form 06 secara berkala. Form 05 merupakan form agenda yang berisi tanggal, materi

yang diajarkan, jumlah mahasiswa yang hadir dan tanda tangan dosen dan penanggung jawab mata kuliah. Sedangkan form 06 merupakan form absensi mahasiswa untuk setiap pertemuannya, beserta nilai tugas, UTS, UAS, dan nilai akhir mahasiswa.

LOGO

DAFTAR HADIR PERKULIAHAN DOSEN

SEMESTER :

1. Dosen/Asisten :
 2. Mata Kuliah :
 Kode/SKS :
 3. Ruang :
 4. Jam :

Form 05

PERTEMUAN KE	HARI/TANGGAL	POKOK BAHASAN	JML MHS YANG HADIR	TANDA TANGAN	
				KETUA MHS	DOSEN
1					
2					
3					
4					
5					
6					
7					
8					

Catatan :

- 1.
- 2.
- 3.
-
-
-

Tanda tangan

Gambar 2.1: Form 05

Evaluasi form 05 dan 06 dilakukan tiga kali setiap semester. evaluasi pertama dilakukan pada minggu pertama untuk melihat mulainya pelaksanaan kuliah pertemuan pertama. Evaluasi kedua dilakukan sebelum Ujian Tengah Semester untuk melihat apakah jumlah pertemuan yang dilakukan sesuai atau cukup untuk melaksanakan Uji-

an Tengah Semester. Evaluasi ketiga dilakukan pada minggu terakhir untuk melihat apakah jumlah pertemuan mencapai syarat minimal 80%. Evaluasi ini dilakukan oleh Tim Penjamin Mutu program studi dan nantinya akan dilaporkan ke Gugus Penjamin Mutu.

Rencana Pembelajaran Semester (RPS) adalah sebuah perangkat pembelajaran yang digunakan sebagai standar proses pembelajaran yang dilakukan. Menurut Permendikbud Nomor 3 Tahun 2020 tentang standar nasional pendidikan tinggi, RPS dikembangkan oleh dosen secara mandiri atau bersama dalam kelompok keahlian suatu bidang ilmu pengetahuan atau teknologi dalam sebuah program studi. Sebuah RPS paling sedikit memuat: (Kemendikbud, 2020)

1. Nama program studi, nama dan kode mata kuliah, semester, sks, dan nama dosen pengampu.
2. Capaian pembelajaran lulusan yang dibebankan pada mata kuliah.
3. Kemampuan akhir yang direncanakan pada tiap tahap pembelajaran untuk memenuhi capaian.

4. Bahan kajian yang terkait dengan kemampuan yang akan dicapai.
5. Metode pembelajaran.
6. Waktu yang disediakan untuk mencapai kemampuan pada tiap tahap pembelajaran.
7. Pengalaman belajar mahasiswa yang diwujudkan dalam deskripsi tugas yang harus dikerjakan oleh mahasiswa dalam satu semester.
8. Kriteria, indikator, dan bobot penilaian.
9. Daftar referensi yang digunakan.

2.4 Sistem Informasi

2.4.1 Definisi Sistem

Sistem adalah kumpulan atau grup dari bagian atau komponen apapun, baik fisik ataupun non fisik yang saling berhubungan satu sama lain dan bekerja sama untuk mencapai satu tujuan tertentu (Susanto, 2017). Sebuah sistem mempunyai karakteristik atau sifat-sifat tertentu, yaitu mempunyai komponen-komponen, batas sistem, lingkungan luar sistem, penghubung, masukan, keluaran, pengolah, dan sasaran atau tujuan (bin Ladjamudin, 2005).

Menurut Romney & Steinbart (2009), sistem adalah suatu rangkaian yang terdiri dari beberapa komponen yang saling berhubungan dan saling berinteraksi antara satu sama lain untuk mencapai tujuan. Sistem biasanya terbagi menjadi subsistem yang lebih kecil yang mendukung sistem yang lebih besar. Dari pengertian-pengertian diatas, dapat disimpulkan bahwa sistem adalah kumpulan dari komponen-komponen yang saling berhubungan dan berinteraksi satu sama lain dalam sebuah rangkaian untuk mencapai tujuan.

2.4.2 Definisi Informasi

Informasi adalah data yang telah diolah ke dalam bentuk yang lebih berguna dan lebih berarti bagi penggunaannya. Informasi merupakan salah satu sumber daya paling penting yang dimiliki oleh suatu organisasi. Menurut Susanto (2017) definisi informasi adalah hasil pengolahan data yang memberikan arti dan manfaat. Alat pengolahan informasi dapat berupa komponen komputer, komponen non komputer, atau kombinasi keduanya (bin Ladjamudin, 2005).

Tidak semua data dapat digunakan untuk mendapatkan informasi yang berkualitas, data tersebut harus memiliki karakteristik tertentu. Untuk mendapat informasi yang berkualitas dibutuhkan data-data yang memiliki karakteristik sebagai berikut (Kroenke & Boyle, 2009):

1. Akurat

Data yang benar, lengkap, dan akurat dan dengan didasarkan pengolahan yang sesuai akan menghasilkan informasi yang berkualitas. Dalam bisnis untuk mengambil keputusan sangatlah membutuhkan data yang akurat. Penggunaan data yang tidak akurat sebagai dasar pengambilan keputusan dapat memberikan dampak yang tidak diinginkan.

2. Tepat Waktu

Ketepatan waktu pada suatu informasi memiliki pengaruh yang sangat besar. Maksud dari ketepatan waktu tersebut adalah data yang digunakan dapat tersedia setiap waktu untuk digunakan sesuai kebutuhannya. Mengolah dan menghasilkan data secara cepat dan tepat akan menghasilkan informasi yang berkualitas dan dapat dimanfaatkan dengan tepat.

Tepat waktu dimaksudkan dengan ketersediaan data setiap waktu diperlukan untuk dapat digunakan untuk kebutuhan tertentu. Informasi yang berkualitas

didapatkan dari data yang dapat diolah dan dihasilkan dengan cepat dan tepat agar pemanfaatannya dapat digunakan dengan tepat. Contohnya, ketika membuat sebuah laporan bulanan pada sebuah instansi, data yang diproses haruslah data yang didapatkan dalam satu bulan dan laporan tersebut harus selesai dengan secepatnya karena akan menjadi pertimbangan manajemen perusahaan dalam membuat keputusan selanjutnya.

3. Relevan

Informasi harus didapatkan dari data yang relevan baik dalam konteks maupun subjek. Relevansi data berdasarkan konteks dimaksudkan pada penggunaan data harus sesuai dalam bidangnya. Contohnya, untuk karyawan dengan sistem *payroll*, data jam kerja merupakan data yang relevan dengan pekerjaan karyawan *payroll* tersebut. Namun, daftar data jam kerja tersebut menjadi tidak relevan secara kontekstual jika digunakan untuk karyawan yang tidak menggunakan sistem payroll.

Relevansi data berdasarkan subjek dimaksudkan pada data yang diatur berdasarkan subjeknya. Contohnya, ketika sebuah instansi membutuhkan data penawaran kredit oleh beberapa bank, maka data suku bunga merupakan data yang relevan berdasarkan subjek bank tersebut.

4. Cukup

Data yang cukup juga sangat berpengaruh dengan kualitas informasi yang dihasilkan. Kecukupan data dimaksudkan dengan ketersediaan data yang memenuhi keperluan dan tidak berlebihan dalam mengolah data tersebut menjadi informasi.

5. Sebanding dengan biaya

Memperoleh data tidaklah gratis. Ada biaya dalam melakukan pengolahan sebuah data termasuk memelihara sistem pengolah data, membayar gaji karyawan yang mengolah data, dan lain sebagainya. Atas dasar hal tersebut, data harus digunakan dengan bijak sehingga informasi yang dihasilkan dapat mengimbangi biaya yang dikeluarkan dalam mengolah data tersebut menjadi informasi.

2.4.3 Definisi Sistem Informasi

Sistem informasi adalah sebuah sistem dalam suatu keorganisasian yang mempertemukan kebutuhan pengelolaan, mendukung operasi, bersifat manajerial, dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang dibutuhkan (Hutahaean, 2015). Sistem informasi merupakan gabungan dari serangkaian komponen yang dapat berupa manusia, prosedur, data, dan teknologi yang digunakan untuk menghasilkan informasi yang bernilai sebagai dasar pengambilan keputusan.

Sistem informasi memiliki komponen-komponen penting yang berinteraksi satu sama lain dan bekerja sama untuk melakukan suatu proses. Berikut adalah komponen-komponen dalam sistem informasi (Jogiyanto, 2005):

1. Input, komponen input menjelaskan bagaimana sistem memperoleh atau menyediakan data untuk diproses lebih lanjut.
2. Process, komponen process menjelaskan bagaimana suatu data diproses sehingga dapat menghasilkan informasi yang lebih bernilai.
3. Output, komponen output adalah komponen hasil dari proses yang telah dilakukan sebelumnya.
4. Penyimpanan, komponen penyimpanan merupakan komponen kegiatan yang dilakukan untuk menyimpan dan memelihara data.

5. Pengendalian, komponen pengendalian merupakan komponen yang menjamin sistem informasi dapat berjalan sesuai yang diharapkan dan merupakan komponen paling penting dalam suatu sistem informasi.

Berdasarkan teori-teori dan pendapat diatas, dapat disimpulkan bahwa sistem informasi merupakan sebuah sistem dalam suatu organisasi yang bersifat manajerial dengan berbagai komponen yang berinteraksi dan bekerja sama antara satu dengan yang lain yang digunakan untuk menghasilkan informasi yang berkualitas dan dapat menjadi dasar dari suatu pengambilan keputusan.

2.5 Software Development Life Cycle (SDLC)

Proses pengembangan sebuah perangkat lunak dan sistem informasi selalu dipengaruhi oleh beberapa metodologi pengembangan. Metodologi pengembangan perangkat lunak mengacu pada kerangka kerja yang digunakan untuk merencanakan, mengelola, dan mengendalikan proses pengembangan sebuah sistem informasi. Metodologi pengembangan perangkat lunak disebut sebagai siklus hidup pengembangan perangkat lunak atau biasa dikenal sebagai *Software Development Life Cycle*. SDLC adalah proses yang menjelaskan metode dan strategi mendesain dan memelihara proyek perangkat lunak untuk memastikan tujuan, sasaran, fungsional, dan kebutuhan pengguna dapat tercapai (Arora & Arora, 2016).

Proses pada metodologi SDLC umumnya dibagi menjadi beberapa fase, yaitu (Dora & Dubey, 2013):

1. *Requirement Analysis*

Requirement Analysis adalah fase pertama dalam SDLC. Tujuan dari *requirement analysis* adalah mengetahui dan mendokumentasikan kebutuhan sebenarnya dari pengguna. Fokus dari *requirement analysis* adalah mengetahui apa

yang dibutuhkan dari sistem yang akan dibuat. Fase ini adalah fase paling penting dalam SDLC. Hasil dari *requirement analysis* adalah deskripsi lengkap dan komprehensif dari kegiatan yang dilakukan perangkat lunak yang akan dikembangkan.

2. *Design*

Fase desain adalah fase paling kreatif pada SDLC. Pada fase desain, spesifikasi kebutuhan yang ada diubah menjadi bentuk terstruktur atau terencana. Fase ini merupakan proses perencanaan dan *problem solving* untuk mencari solusi dengan perangkat lunak. Hasil dari fase ini adalah dokumen desain dari perangkat lunak yang akan dikembangkan.

3. *Coding*

Pada fase *coding*, dokumen desain dari perangkat lunak diubah menjadi *code* dengan menggunakan bahasa pemrograman tertentu. Fase ini adalah fase *login* dari SDLC. Hasil dari fase ini adalah kode program.

4. *Testing*

Sebelum setelah fase *coding*, *testing* dilakukan untuk mengetahui hasil dari aplikasi yang telah dikembangkan. *Testing* dibuat untuk mengetahui hasil asli dan hasil yang diharapkan. Fase ini adalah fase paling penting dan berpengaruh. *Testing* yang efektif akan menghasilkan produk perangkat lunak dengan kualitas tinggi, akurat, dan hasil yang dapat diandalkan.

5. *Maintenance*

Fase akhir dari SDLC, dimana perangkat lunak yang telah dikembangkan didistribusikan ke pengguna akhir untuk pemeliharaan dan menggunakan untuk pengoperasian seharusnya.

SDLC memiliki berbagai macam model seperti Waterfall, V model, Spiral, Iterative, Agile, Extreme Programming, Scrum, dan lain sebagainya. Model yang akan dipilih dalam penelitian ini adalah Spiral. Model Spiral adalah model SDLC yang menggabungkan proses model Iterative dengan elemen-elemen dari model Waterfall. Model Spiral berfokus pada analisis risiko dan meminimalkan risiko dari proyek. Hal tersebut dilakukan dengan cara membagi sebuah proyek menjadi beberapa bagian kecil sehingga mempermudah jika terdapat perubahan sekaligus melakukan evaluasi risiko dan mempertimbangkan kelanjutan proyek dalam prosesnya. Kemudahan dalam menerapkan model Spiral secara umum ditambah dengan kemudahan dalam melakukan perubahan sistem ketika ada fitur yang ingin ditambahkan ini adalah alasan penulis memilih model Spiral. Fase-fase pada model Spiral dapat dideskripsikan sebagai berikut (Alshamrani & Bahattab, 2015):

1. Perencanaan

Fase ini meliputi pemahaman dari kebutuhan sistem dengan melakukan komunikasi yang berkesinambungan antara pengguna dan sistem analis.

2. Analisis risiko

Dalam fase ini dilakukan sebuah proses untuk mengidentifikasikan risiko yang dapat muncul dan solusi-solusi yang dapat dilakukan. Pada akhir fase ini menghasilkan sebuah prototipe.

3. Pengembangan

Pada fase ini program atau perangkat lunak dihasilkan sekaligus dengan dilakukannya pengujian.

4. Evaluasi

Fase ini memungkinkan pengguna untuk mengevaluasi hasil dari iterasi proyek sebelum dilanjutkan ke spiral selanjutnya.

2.6 *Unified Modeling Language (UML)*

Menurut Windu dan Grace (2013) dalam jurnal Suendri (2018) *Unified Modeling Language* atau UML adalah bahasa spesifikasi standar yang biasa dipergunakan untuk mendokumentasikan, menspesifikasikan dan membangun perangkat lunak. UML merupakan sebuah metodologi dalam mengembangkan sistem berorientasi objek dan juga merupakan alat untuk mendukung pengembangan sistem.

Saat ini UML telah menjadi standar bahasa visual yang digunakan untuk memodelkan sistem yang berorientasi objek. Fungsi dari UML adalah untuk melakukan pemodelan sehingga penggunaan dari UML sendiri tidak terbatas pada metodologi tertentu, meskipun pada kenyataannya UML paling banyak digunakan untuk metodologi berorientasi objek.

Terdapat beberapa jenis diagram UML dan pada penelitian ini akan dibahas beberapa diantaranya, yaitu:


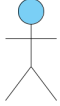

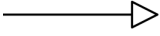
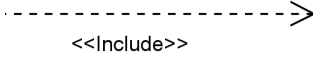
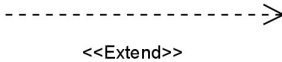
1. *Use Case Diagram*

Use case diagram adalah diagram UML yang menggambarkan hal-hal apa saja yang dapat dilakukan oleh aktor atau pengguna dalam sistem yang dibuat. Aktor yang dimaksud di dalam *Use Case Diagram* dapat berupa user, sistem internal, atau bahkan sistem eksternal yang dihubungkan dengan sistem yang dibuat. Simbol-simbol yang ada pada use case diagram dapat dilihat pada tabel 2.1.

2. *Class Diagram*

Class diagram adalah sebuah diagram yang digunakan untuk menggambarkan

Tabel 2.1: Simbol-Simbol Pada *Use Case Diagram*

Simbol	Deskripsi
	<i>Use case</i> adalah langkah atau tindakan yang dapat dilakukan di dalam sistem.
	Aktor merupakan sebuah abstraksi dari pengguna atau peran yang lain yang terlibat dan berinteraksi dengan sistem.
	Asosiasi adalah hubungan antara aktor dan <i>use case</i> yang menggambarkan hubungan interaksi langsung antara keduanya.
	Generalisasi adalah hubungan antara aktor yang menggambarkan kesamaan perilaku objek induk dengan objek anak.
	<i>Include</i> adalah hubungan antara <i>use case</i> yang menggambarkan kebutuhan suatu <i>use case</i> untuk dilanjutkan dengan <i>use case</i> lain.
	<i>Extend</i> adalah hubungan antara <i>use case</i> yang menggambarkan kemungkinan suatu <i>use case</i> untuk dilanjutkan dengan <i>use case</i> lain.

kelas, atribut-atribut kelas, dan hubungan antar setiap kelas. Menurut Whitten dalam jurnal Suendri (2018) kelas adalah suatu set objek yang memiliki atribut dan perilaku yang sama. Hubungan antar kelas dalam *class diagram* memiliki keterangan yang biasa disebut dengan *Multiplicity* atau *Cardinality*. Ada beberapa jenis *Multiplicity* sebagaimana digambarkan pada tabel 2.2. Simbol-simbol yang ada pada *class diagram* dapat dilihat pada tabel 2.3.

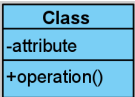

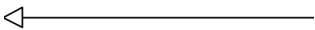
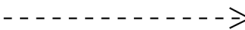

3. Activity Diagram

Activity diagram adalah diagram yang digunakan untuk menggambarkan proses jalannya suatu aktivitas atau kegiatan yang dapat dilakukan dalam sebuah

Tabel 2.2: *Multiplicity* atau *Cardinality* pada *Class Diagram*

<i>Multiplicity</i>	Deskripsi Hubungan
1	Satu dan hanya satu.
0..*	Boleh tidak ada atau ada.
0..1	Boleh tidak ada atau hanya satu.
1..*	Harus ada dan boleh lebih dari satu.
n..m	Batasan jarak antara n dan m. Minimal n dan maksimal m.




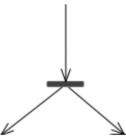
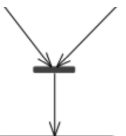

Tabel 2.3: Simbol-Simbol Pada *Class Diagram*

Simbol	Deskripsi
	Kelas merupakan elemen dan interaksi utama pada aplikasi.
	Asosiasi adalah hubungan atau relasi umum antara kelas yang disertai dengan <i>multiplicity</i> .
	Generalisasi adalah hubungan antara kelas yang menjelaskan bahwa suatu kelas merupakan <i>superclass</i> dari kelas yang lain.
	Generalisasi adalah hubungan antara kelas yang menggambarkan kesamaan perilaku <i>superclass</i> dan <i>subclass</i> .
	Agregasi adalah hubungan atau relasi antar kelas yang bermakna semua objek kelas merupakan bagian dari kelas lain.

sistem. *Activity diagram* biasanya menggambarkan bagaimana masing-masing kegiatan dimulai, keputusan yang mungkin terjadi di dalam kegiatan tersebut, hingga berakhirnya kegiatan yang dilakukan. *Activity diagram* dapat menggambarkan lebih dari satu proses aksi dalam waktu bersamaan. Menurut Haviuddin

dalam jurnal Suendri (2018) *Activity diagram* adalah aktivitas-aktivitas, objek, *state*, transisi *state*, dan *event*. Dengan kata lain kegiatan diagram alur kerja menggambarkan perilaku sistem untuk aktivitas. Simbol-simbol yang digunakan pada activity diagram dapat dilihat pada tabel 2.4.

Tabel 2.4: Simbol-Simbol Pada *Activity Diagram*

Simbol	Deskripsi
	<i>Start Point</i> adalah poin dimulainya <i>activity</i> .
	<i>End Point</i> adalah poin berakhirnya <i>activity</i> .
	<i>Activity</i> menggambarkan suatu kegiatan atau proses bisnis.
	<i>Fork</i> adalah pembagian sebuah kegiatan menjadi lebih dari satu.
	<i>Join</i> adalah penggabungan kegiatan menjadi satu.
	<i>Decision Node</i> adalah pembagian sebuah kegiatan menjadi lebih dari satu berdasarkan keputusan yang diambil.

2.7 Entity Relationship Diagram

Entity Relationship Diagram (ERD) adalah sebuah tipe diagram struktural yang digunakan untuk mendesain *database*. ERD memiliki banyak simbol dan penghubung yang digunakan untuk menggambarkan dua informasi penting, yaitu entitas utama dalam lingkup sistem dan hubungan antara masing-masing entitas yang ada.

Menurut Simarmata(2010:67) dalam jurnal Fridayanthie & Mahdiati (2016) ERD adalah alat pemodelan data utama dan akan membantu mengorganisasi data dalam suatu proyek ke dalam entitas-entitas dan menentukan hubungan antar entitas. Sebuah ERD terdiri dari tiga komponen utama yaitu:

1. *Entity*

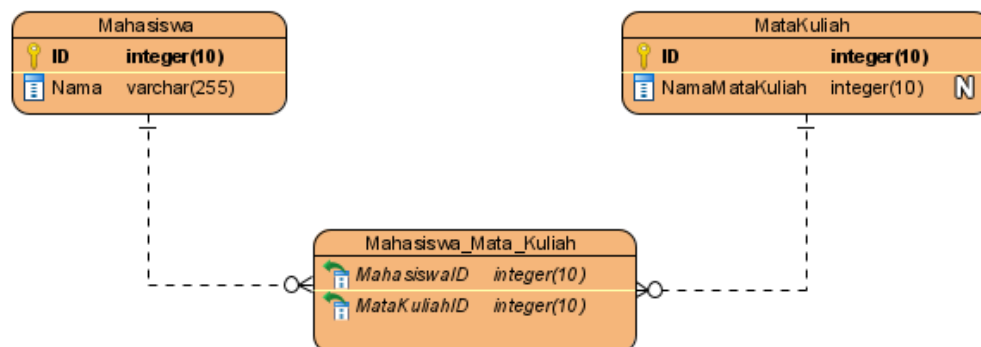
Entity atau entitas adalah sebuah benda atau konsep yang dapat didefinisikan di dalam sistem seperti Mahasiswa, Profil, Nilai, dan lain-lain. Notasi entitas biasanya dilambangkan dengan persegi panjang dengan nama dari entitas tersebut di atasnya dan berisi atribut dari entitas tersebut

2. *Attribute*

Attribute adalah properti atau karakteristik dari suatu entitas. Sebuah atribut memiliki nama yang menjelaskan karakteristiknya dan tipe yang menjelaskan jenis dari karakteristik tersebut seperti *varchar* atau *int*. Sebagai contoh dalam entitas Mahasiswa terdapat atribut nama, nomor registrasi, alamat, dan lain-lain.

3. *Relationship*

Relationship dalam ERD adalah hubungan antara dua entitas yang menggambarkan bahwa kedua entitas tersebut memiliki suatu kaitan tertentu. Sebagai contoh seorang mahasiswa dapat mendaftarkan kelas yang ingin diikuti. *Relationship* didefinisikan dengan menggunakan *Foreign Key* yang mengacu pada sebuah *Primary Key*.



Gambar 2.3: Contoh Sederhana ERD

2.8 Database

Database merupakan sebuah kumpulan data-data yang terstruktur sehingga menjadi suatu informasi yang dapat dimanfaatkan. *Database* terbentuk dari sekelompok data-data yang memiliki jenis atau sifat yang sama. Contohnya *database* karyawan dapat berisi data-data berupa nama-nama, divisi-divisi, dan jabatan-jabatan karyawan.

Database adalah komponen yang paling penting dalam pembangunan sebuah sistem informasi, karena *database* merupakan tempat untuk menyimpan dan mengelola seluruh data yang dimiliki. Data diatur sedemikian rupa sehingga tidak ada data yang terduplikasi agar data dapat diolah dengan tepat dan mudah untuk menghasilkan informasi.

Menurut Susanto dalam jurnal Sudrajat & Sudrajat (2014) *database* atau basis data adalah suatu kumpulan data terhubung (*interrelated data*) yang disimpan secara bersama-sama dalam suatu media, tanpa mengatap satu sama lain atau tidak perlu kerangkaan data, data disimpan dengan cara-cara tertentu sehingga mudah untuk digunakan atau ditampilkan kembali. *Database* dapat didefinisikan sebagai kumpulan data yang terintegrasi dan diatur sedemikian rupa sehingga data tersebut dapat dima-

nipulasi, diambil, dan dicari secara cepat (Raharjo, 2011).

Untuk mengelola sebuah *database* diperlukan sebuah perangkat lunak khusus yang disebut *Database Management System* atau biasa disebut DBMS. DBMS adalah sebuah program yang digunakan untuk membuat, mendefinisikan, dan memelihara *database*. DBMS yang akan digunakan dan dibahas pada penelitian ini adalah MySQL yang juga merupakan salah satu DBMS yang paling banyak digunakan. MySQL merupakan sebuah DBMS *Open Source* sehingga *source code* dari MySQL dapat diakses secara umum dan dapat diunduh secara gratis. Menurut MySQL *manual*, MySQL merupakan perangkat lunak *database* berbasis SQL (*Structured Query Language*) yang dapat menangani sistem manajemen *database* dan sistem manajemen *database* relasional.

Sebagai sebuah DBMS, MySQL memiliki beberapa fitur seperti:

1. *Multiplatform*

MySQL tersedia pada berbagai platform umum seperti Windows, Linux, Unix, dan lain-lain.

2. Dukungan SQL

SQL sudah menjadi standar pada penggunaan *database* relasional. Memiliki pengetahuan SQL akan sangat memudahkan penggunaan MySQL.

3. Jaminan keamanan akses

MySQL memberikan jaminan keamanan *database* dengan beberapa kriteria pengaksesan.

4. Cepat, handal, dan mudah digunakan

MySQL termasuk sebagai *database server* yang handal, dapat menangani *database* dengan ukuran besar dengan cepat, memiliki banyak fungsi pengaksesan,

dan juga mudah digunakan.

2.9 Web Service

Service dalam dunia komputer adalah sebuah fungsionalitas dari program perangkat lunak yang dibuat untuk melayani kebutuhan dari program lain. Sebuah *service* bertujuan untuk memungkinkan beberapa program lainnya untuk menggunakan *service* tersebut dengan berbagai macam tujuan. Menurut Wahli dkk dalam jurnal Fauziah (2013) *web service* merupakan suatu aplikasi modular yang bersifat mandiri dan dapat dipublikasikan, dialokasikan dan dipanggil melalui sebuah jaringan. Tujuan dari dibuatnya *web service* adalah untuk membuat sebuah fungsionalitas yang dapat diakses melalui *Internet Protocol* (IP) yang tidak bergantung pada platform dan bahasa pemrograman yang digunakan. *Web service* dapat menjalankan fungsionalitas bisnis mulai dari yang sederhana seperti *request-reply* hingga interaksi bisnis proses sepenuhnya. Untuk menjalankan proses yang rumit, sebuah *web service* dapat bekerja sama atau mengandalkan *web service* lain.

Web service dapat dibangun menggunakan beberapa bentuk arsitektur, salah satunya yang akan dibahas adalah *Representational State Transfer*(REST). *Web service* yang dibuat mengikuti arsitektur REST disebut RESTful *Web Service*. Pada RESTful *web service*, *server* menyediakan data atau sumber daya dan *client* dapat mengakses data tersebut untuk digunakan. Setiap data atau *resource* pada REST *server* diidentifikasi oleh *global id* yang merupakan URIs (*Universal Resource Identifiers*). Pada umumnya data tersebut diberikan dalam format teks seperti JSON atau XML.

Metode-metode HTTP yang biasa digunakan dalam RESTful *Web Service* adalah sebagai berikut:

1. *Get*

Get adalah metode untuk membaca data.

2. *Post*

Post adalah metode untuk membuat data baru.

3. *Delete*

Delete adalah metode untuk menghapus data.

4. *Put* *Put* adalah metode untuk memperbarui data yang sudah ada atau membuat data baru jika data belum ada.

2.10 *Framework*

Framework atau kerangka kerja adalah desain keseluruhan atau sebagian dari sistem yang berupa sebuah kumpulan kelas abstrak dan fungsi-fungsinya yang digunakan untuk mempermudah pengembangan sebuah perangkat lunak. Dengan menggunakan sebuah *framework*, seorang *developer* tidak perlu lagi mengimplementasikan fungsi-fungsi umum yang biasa digunakan karena sudah disediakan oleh *framework* yang digunakan sehingga *developer* hanya perlu menyesuaikan dengan kebutuhan dari program yang akan dibuat. *Framework* dalam pembuatan *website* umumnya terdapat dua jenis, yaitu *framework* untuk membuat tampilan pengguna atau *frontend* dan *framework* untuk membuat sisi *server* atau *backend*. Beberapa *framework* juga dapat digunakan untuk membangun keseluruhan dari sebuah *website* mulai dari *frontend* hingga *backend*. Beberapa *framework* yang dapat digunakan untuk membuat *frontend* adalah AngularJS, ReactJS, dan VueJS. Sedangkan untuk *backend* dapat menggunakan CI, Laravel, dan ExpressJS. Pada penelitian ini, digunakan dua *framework* untuk membuat *frontend* dan *backend*, yaitu VueJS dan Laravel.

2.11 MVC

Konsep MVC adalah sebuah pola desain perangkat lunak yang biasa digunakan untuk membangun user interface yang memisahkan logika pada program yang dibuat menjadi tiga elemen yang saling berhubungan. Tiga elemen tersebut adalah:

1. *Model*

Model merupakan komponen yang digunakan untuk mendefinisikan data-data utama yang akan digunakan dan berhubungan langsung dengan *database*. *Model* berisi struktur dari data yang akan digunakan beserta fungsi-fungsi yang bertujuan untuk mengolah data tersebut.

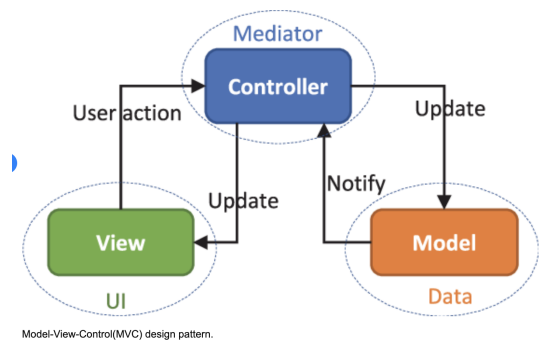
2. *View*

View adalah komponen yang mengatur tampilan *website* yang dilihat oleh pengguna. Pada komponen inilah bagan, tabel, dan berbagai bentuk tampilan diproses. Umumnya komponen *view* berupa *file* HTML. Pada Laravel komponen *view* bisa menggunakan *templating engine* Blade atau bisa menggunakan *framework* Vue.JS.

3. *Controller*

Controller adalah komponen yang berisi segala logika dan alur kerja dari sistem. *Controller* mengubah input yang diterima dari pengguna dan menentukan perintah selanjutnya yang akan dilaksanakan oleh aplikasi.

Selain dari memisahkan aplikasi menjadi tiga, konsep MVC juga mendefinisikan interaksi antara ketiganya. Ketika pengguna menggunakan aplikasi website, pengguna tersebut sedang berinteraksi dengan komponen *view*. Komponen *view* lalu menerima input dari pengguna dan meneruskan input tersebut ke *controller*. Contro-



Gambar 2.4: Pola Desain MVC

lalu mengambil atau mengubah data pada model. Setelah mendapat data yang dibutuhkan Controller lalu mengembalikan data tersebut dan memperbarui view.

BAB III

IMPLEMENTASI PROGRAM

Pada penelitian ini, penulis mengembangkan perangkat lunak dengan menggunakan proses model Spiral dimana pengembangan dilakukan dalam beberapa *cycle* atau iterasi. Tahapan-tahapan yang dilakukan pada penelitian ini adalah sebagai berikut:

3.1 Analisis Kebutuhan

Pada tahapan ini, penulis mengumpulkan informasi tentang kebutuhan dari perangkat lunak yang akan dikembangkan dengan cara mewawancarai beberapa pihak seperti Koordinator Program Studi, Tim Penjamin Mutu Prodi, dan Gugus Penjamin Mutu FMIPA. Penulis juga mengumpulkan informasi dari pengumpulan data pada penelitian serupa yang sebelumnya pernah dilakukan yaitu penelitian yang dilakukan oleh Andini et al. (2017) yang berjudul "Perancangan dan Implementasi Sistem Absensi Online Berbasis Android di Lingkungan Universitas Negeri Jakarta" dan Kulsum (2021) yang berjudul "Rancang Bangun Sistem Presensi Akademik Berbasis Web Dengan *Framework* Laravel di Lingkungan Program Studi Ilmu Komputer Universitas Negeri Jakarta". Berikut beberapa hasil kebutuhan yang penulis dapatkan:

1. Sistem memiliki lima jenis pengguna yaitu mahasiswa, penanggung jawab mata kuliah, dosen, tim penjamin mutu, dan admin.
2. Mahasiswa dapat melihat kelas-kelas yang diambil, mengisi presensi ketika sudah ada pertemuan atau form 05 yang diisi oleh penanggung jawab atau dosen, dan melihat nilai tugas-tugas yang telah diisi oleh dosen.

3. Penanggung jawab mata kuliah adalah bagian dari mahasiswa dengan tugas tambahan yaitu mengisi atau memvalidasi form 05. Penanggung jawab mata kuliah dipilih oleh dosen pada pertemuan pertama.
4. Dosen dapat melihat kelas-kelas yang diampu, mengisi form 05 untuk pertemuan pada kelas tersebut, dan dapat menutup waktu presensi untuk mahasiswa. Dosen juga dapat menambahkan tugas dan mengisi nilai tugas setiap mahasiswa dengan mengunduh format excel yang telah tersedia dan mengunggah *file excel* tersebut ketika ada pembaruan. Ketika penanggung jawab mata kuliah tidak bisa hadir, dosen juga dapat menunjuk penanggung jawab mata kuliah sementara untuk pertemuan tersebut.
5. Tim penjamin mutu adalah bagian dari dosen dengan tambahan yaitu memiliki menu monitoring yang berisi informasi semua kelas-kelas pada program studi termasuk yang tidak diampu oleh dosen tersebut. Tim penjamin mutu hanya dapat melihat form 05 dan form 06 tanpa bisa merubah data di dalamnya.
6. Admin hanya memiliki satu tugas yaitu untuk mengatur atau mengubah jenis user dosen atau tim penjamin mutu jika ada perubahan.
7. Satu kelas bisa diampu lebih dari satu dosen.

3.2 Desain Sistem

Dalam tahapan ini, penulis membuat desain atau *blueprint* dengan menggunakan UML(*Unified Modelling Language*) dari hasil analisis kebutuhan pada tahap sebelumnya. Diagram-diagram yang dibuat oleh penulis adalah *Use Case Diagram*, *Activity Diagram*, dan *Class Diagram*. Selain dari diagram-diagram UML tersebut, penulis juga membuat desain *Entity Relationship Diagram* dan *mockup* dari tampil-

an antar muka program. Penulis membuat semua diagram-diagram berikut dengan menggunakan Visual Paradigm.

3.2.1 *Use Case Diagram*

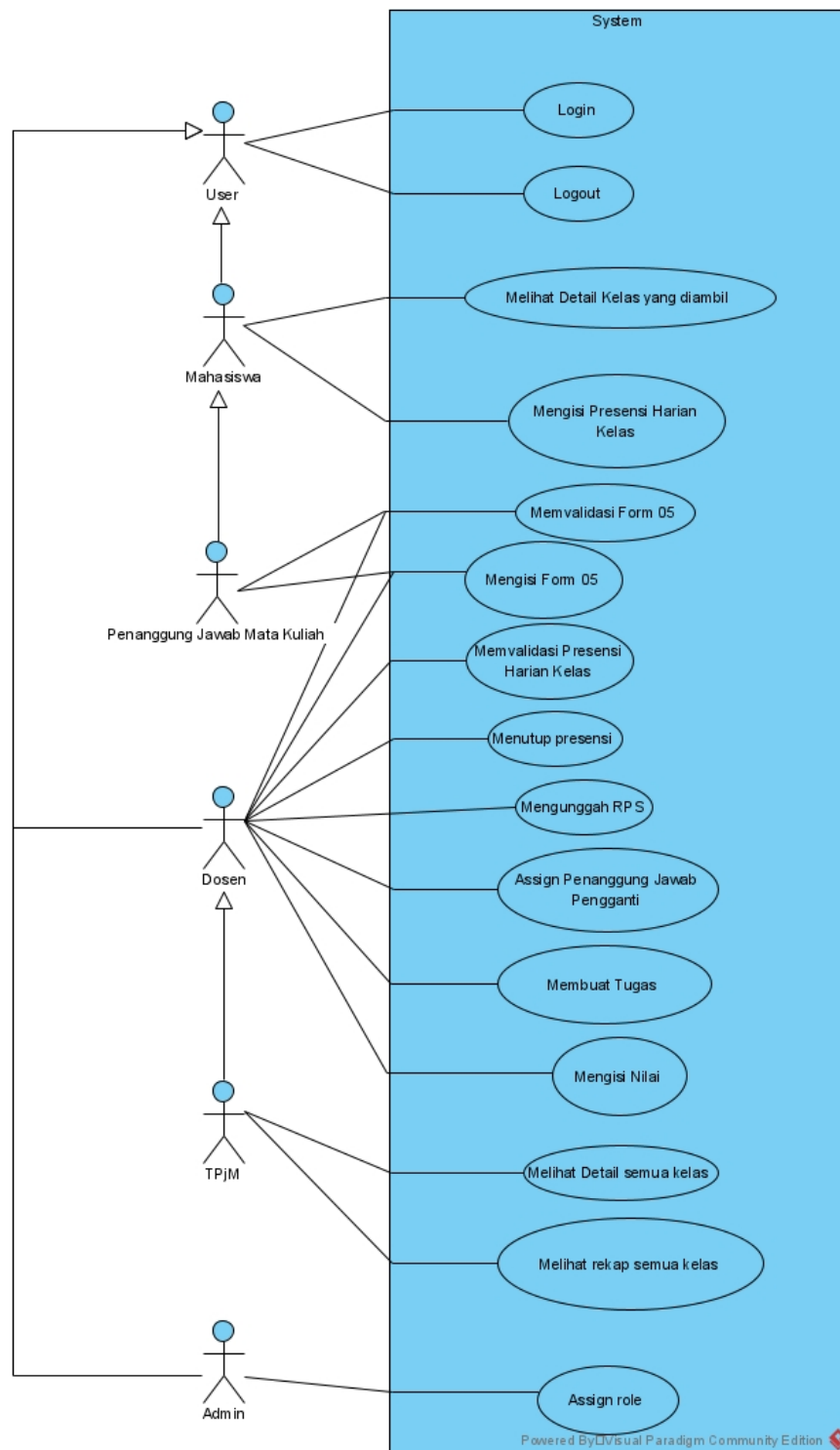
Use Case Diagram menggambarkan interaksi apa saja yang dapat dilakukan oleh setiap jenis *user* yang ada pada sistem yang dikembangkan oleh penulis. Pada sistem yang dibuat terdapat 5 aktor yaitu Mahasiswa, Penanggung Jawab Mata Kuliah, Dosen, TPjM, dan admin. Kelima aktor tersebut dapat digeneralisasi menjadi aktor *User* dimana aktor *User* harus melakukan *login* sebelum mengakses sistem. Desain *Use Case Diagram* untuk sistem yang akan dibuat ada pada Gambar 3.1.

Aktor Mahasiswa dapat melihat detail kelas-kelas yang diambil seperti presensi dan nilai setiap mahasiswa pada kelas tersebut dan mengisi presensi pertemuan yang sedang berlangsung.

Aktor Penanggung Jawab Mata Kuliah dapat digeneralisasi menjadi aktor Mahasiswa sehingga mendapat wewenang yang sama seperti aktor Mahasiswa. Aktor penanggung jawab mata kuliah juga dapat mengisi form 05 atau memvalidasi form 05 yang telah dibuat oleh dosen dan memvalidasi presensi form 06 mahasiswa lain.

Aktor Dosen memiliki wewenang yang hampir sama dengan wewenang khusus penanggung jawab mata kuliah seperti mengisi form 05, memvalidasi form 05, dan memvalidasi presensi harian kelas. Aktor dosen juga memiliki wewenang tambahan seperti membuat tugas, mengisi nilai, dan memilih penanggung jawab pengganti jika penanggung jawab tidak dapat melakukan tugasnya.

Aktor TPjM dapat digeneralisasi menjadi aktor dosen sehingga memiliki wewenang aktor dosen dengan tambahan dapat melihat rangkuman performa semua kelas dan melihat detail kelas-kelas lain di program studi yang tidak diampu olehnya.



Gambar 3.1: Desain *Use Case Diagram*

Aktor admin adalah aktor yang dapat mengganti *role user-user* lainnya jika

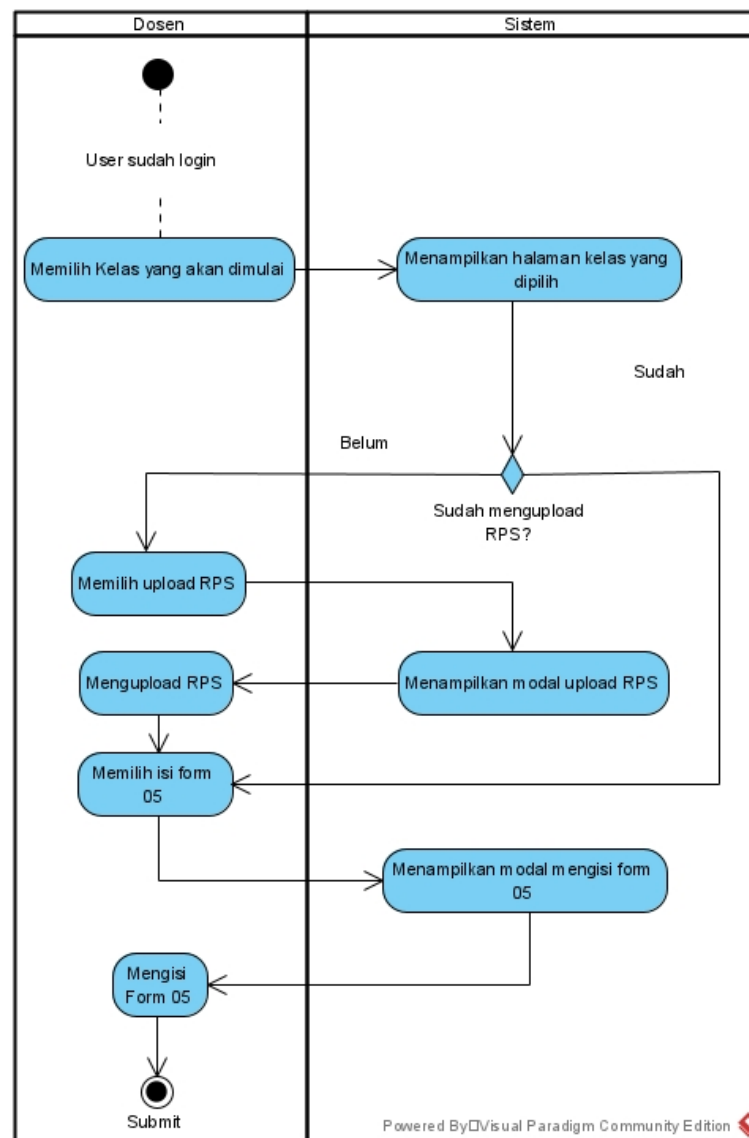
terjadi perubahan. Aktor admin juga bertanggung jawab untuk melakukan *import* data pada awal semester.

3.2.2 *Activity Diagram*

Activity Diagram menggambarkan bagaimana alur menjalankan aktivitas yang ada pada sistem. Pada desain sistem ini, dibuat beberapa diagram yang menggambarkan aktivitas-aktivitas utama yang dapat dilakukan pada sistem ini, yaitu mengisi presensi, memvalidasi form, membuat tugas, mengunggah nilai, mengunggah rps, dan memvalidasi presensi.

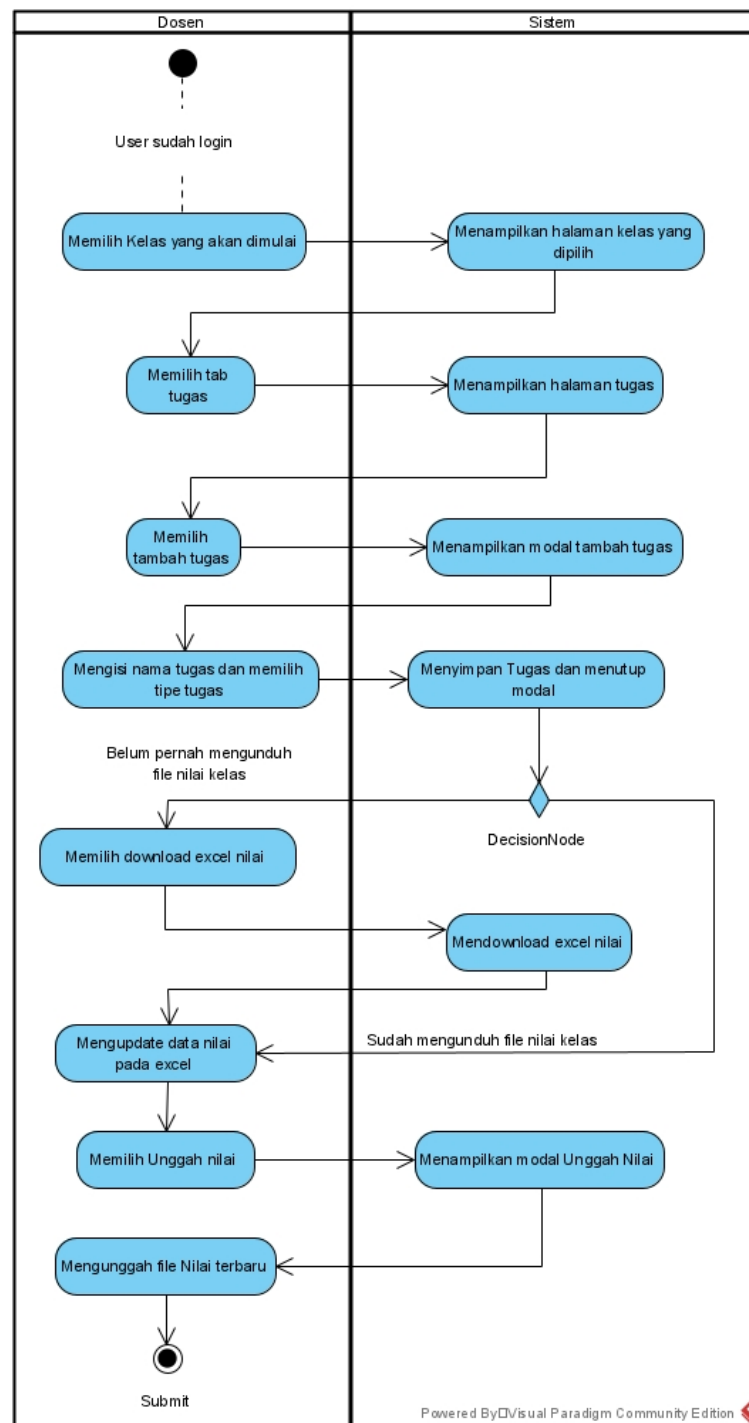
Pada gambar 3.2 menggambarkan aktivitas mengunggah RPS dan menambah form 05 yang dapat dilakukan oleh dosen. Dosen diharuskan untuk mengunggah RPS pada pertemuan awal perkuliahan untuk dapat diakses oleh mahasiswa yang mengambil mata kuliah tersebut dan Tim Penjamin Mutu sebagai bahan evaluasi kesesuaian jalannya mata kuliah dengan RPS. Jika pada mata kuliah tersebut sudah memiliki RPS, dosen bisa langsung menambahkan form 05 dan memulai kelas. Form 05 dibutuhkan untuk membuka kesempatan mahasiswa mengisi presensi pertemuan tersebut.

Pada gambar 3.3 menggambarkan aktivitas membuat tugas dan mengunggah nilai yang dapat dilakukan oleh dosen. Dosen dapat membuat tugas dengan cara menekan tombol tambah tugas dan mengisi nama tugas, tanggal, dan tipe tugas. Tipe tugas merupakan pilihan antara tugas, UTS, dan UAS. Tanggal tugas merupakan tanggal tugas tersebut diberikan dan dapat diubah jika tugas tersebut sudah diberikan kepada mahasiswa sebelumnya dan baru dimasukkan ke sistem. Dosen dapat mengunggah nilai dengan cara mengunduh *template* nilai yang merupakan *file* yang sudah berisi tabel nama-nama mahasiswa dan kolom-kolom kosong untuk nilai-nilai tugas, UTS, dan UAS. Dosen dapat mengisi file excel tersebut lalu diunggah kembali ke sistem untuk memperbarui nilai pada sistem.



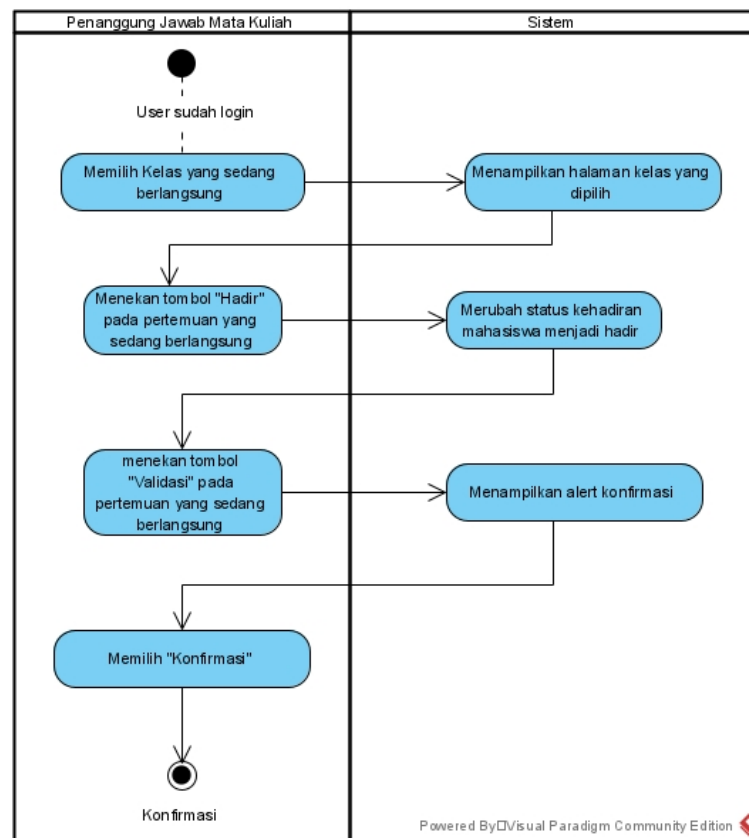
Gambar 3.2: Activity Mengunggah RPS dan Menambah Form 05

Pada gambar 3.4 menggambarkan aktivitas mengisi presensi yang dapat dilakukan oleh mahasiswa dan memvalidasi form 05 yang dapat dilakukan oleh penanggung jawab mata kuliah. Mahasiswa dapat mengisi presensi dengan cara membuka halaman kelas yang sedang berlangsung dan menekan tombol Hadir. Presensi mahasiswa tersebut akan menunggu validasi oleh dosen untuk dianggap valid dan terekam oleh sistem. Penanggung Jawab Mata Kuliah diharuskan melakukan validasi form 05



Gambar 3.3: Activity Membuat Tugas dan Mengunggah Nilai

yang telah dibuat oleh dosen dengan cara menekan tombol validasi pada tabel form 05 tersebut.

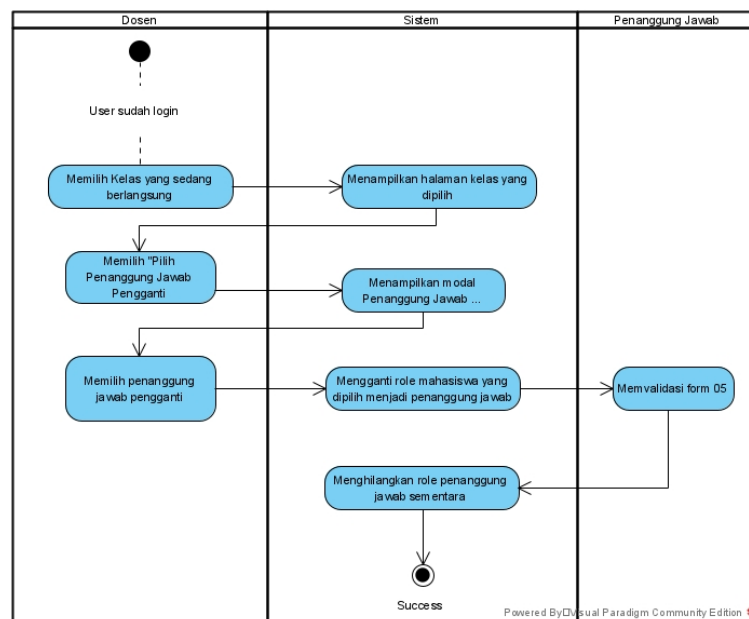


Gambar 3.4: Activity Mengisi Presensi dan Memvalidasi Form 05

Pada gambar 3.5 menggambarkan aktivitas memilih penanggung jawab pengganti yang dapat dilakukan oleh dosen jika penanggung jawab mata kuliah tidak hadir pada pertemuan yang sedang berlangsung untuk memvalidasi form 05 yang telah dibuat. Penanggung jawab pengganti bersifat sementara dan hanya berlaku pada pertemuan tersebut.

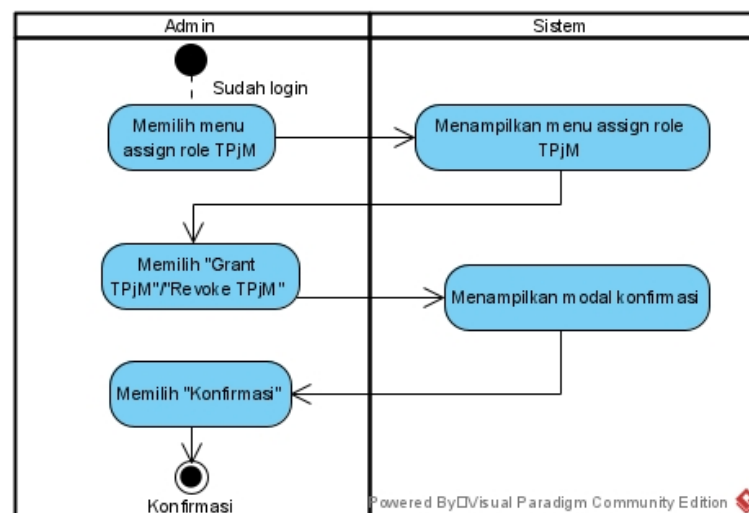
Pada gambar 3.6 menggambarkan aktivitas mengganti *role user* yang dapat dilakukan oleh admin. Aktivitas ini utamanya digunakan untuk mengganti *role* TPjM yang dapat berganti setiap tahunnya.

Pada gambar 3.7 menggambarkan aktivitas memvalidasi presensi yang dapat dilakukan oleh dosen. Dosen harus memvalidasi presensi mahasiswa agar presensi tersebut valid dan terekam oleh sistem. Dosen dapat memilih untuk memvalidasi

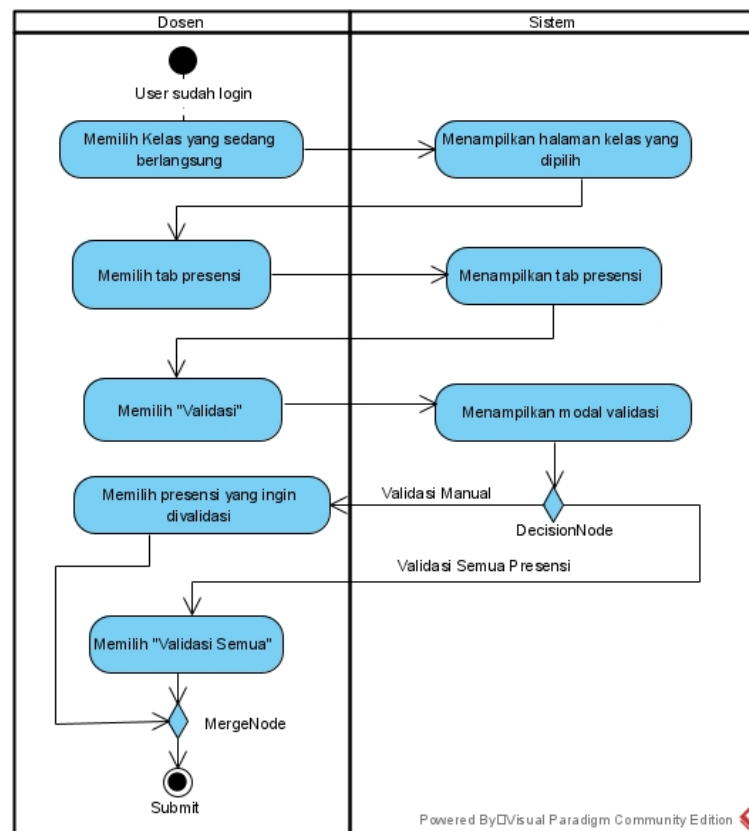


Gambar 3.5: Activity Memilih Penanggung Jawab Pengganti

semua presensi yang sudah diisi oleh mahasiswa atau memilih presensi yang tidak valid dengan cara membatalkan tanda centang *checkbox* pada *list* presensi yang akan divalidasi.



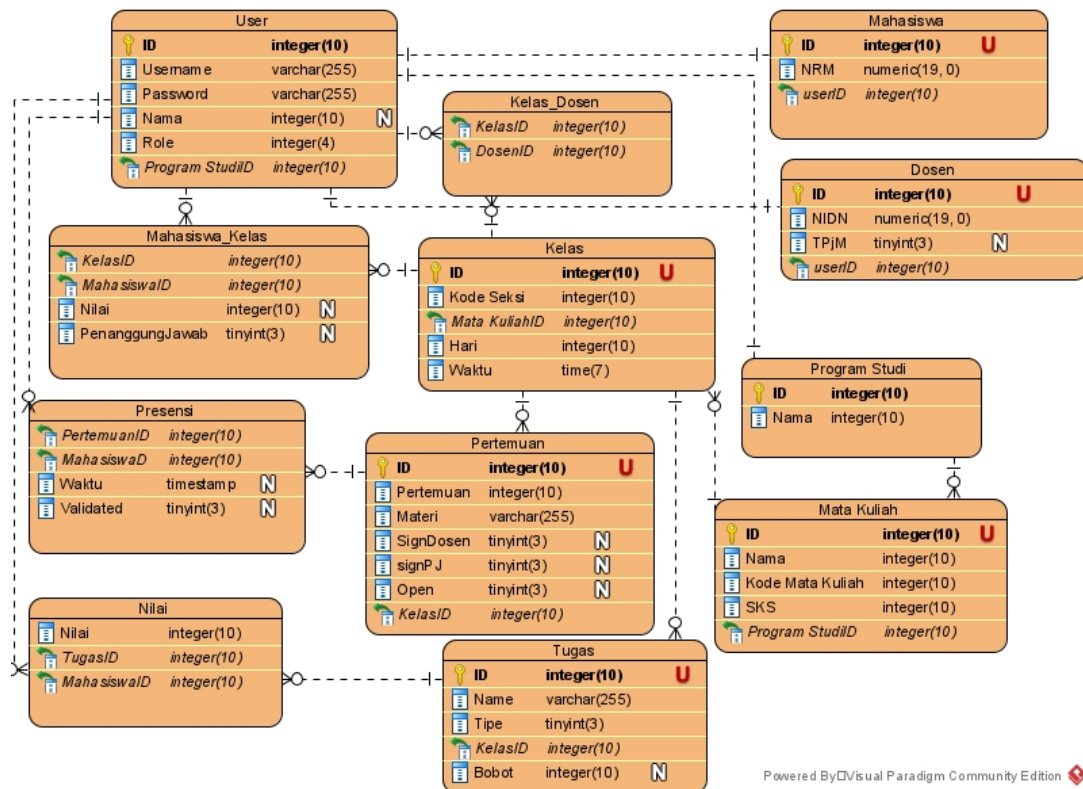
Gambar 3.6: Activity Mengganti role user



Gambar 3.7: Activity Memvalidasi Presensi

3.2.3 Class Diagram

Desain *Class Diagram* pada sistem yang dibuat digambarkan dengan mengikuti konsep MVC dimana *class* pada desain dibagi menjadi tiga jenis yaitu *model*, *view*, *controller*. Pada penggunaannya *class model* dan *controller* dibuat pada bagian *backend* dengan menggunakan Laravel sedangkan *class view* dibuat pada bagian *frontend* dengan menggunakan *Vue*. Pada diagram berikut ketiga jenis *class* dibedakan dengan menggunakan warna yang berbeda yaitu biru untuk *model*, hijau untuk *view*, dan merah untuk *controller*. Desain *class diagram* pada sistem dapat dilihat pada gambar 3.8



Gambar 3.9: Desain Entity Relationship Diagram

3.2.5 Rancangan Tampilan Antarmuka Program

Pada rancangan tampilan antar muka program, penulis membuat desain awal dengan menggunakan *tools online*. Pada halaman *login* yang dapat dilihat pada gambar 3.10, user mengisi email dan *password* yang terdaftar pada *web service* SIAKAD.

Setelah *user* berhasil melakukan *login*, *user* akan dibawa ke halaman *dashboard* dengan menu yang sedikit berbeda bagi setiap jenis *user*. Pada *user* mahasiswa yang dapat dilihat pada gambar 3.11, menu yang akan ditampilkan adalah *Home*, Rekap Presensi, dan daftar kelas yang diambil oleh mahasiswa tersebut. Pada *user* dosen, menu yang akan ditampilkan adalah *Home* dan daftar kelas yang diampu oleh dosen tersebut. Pada *user* TPjM menu tambahan akan ditampilkan berupa *Monitoring*.

Pada menu Rekap Presensi yang dapat dilihat pada gambar 3.12, mahasiswa

Sistem Informasi Kegiatan Belajar Mengajar

SISTEM INFORMASI MONITORING KBM

Silahkan Login

Email

Password

Login

Gambar 3.10: Desain Halaman *Login*

Sistem Informasi Monitoring KBM - Dashboard

SIM KBM

Home

Rekap Presensi

Daftar Kelas

Sistem Informasi Monitoring Kegiatan Belajar Mengajar

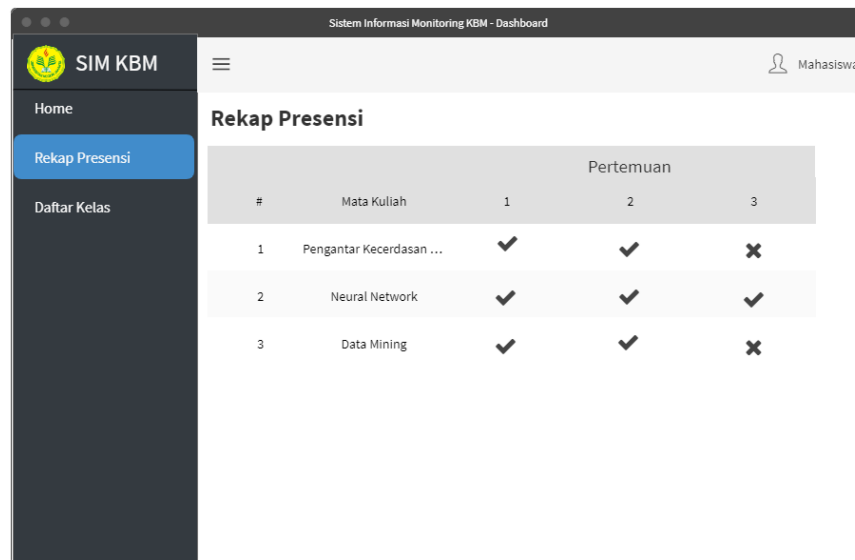
Selamat Datang, Mahasiswa

Mahasiswa

Gambar 3.11: Desain Halaman *Dashboard* Pada mahasiswa

dapat melihat rekap dari presensi setiap mata kuliah yang diambil yang tergabung menjadi satu. Pada menu *monitoring* yang dapat dilihat pada gambar 3.13, TPjM

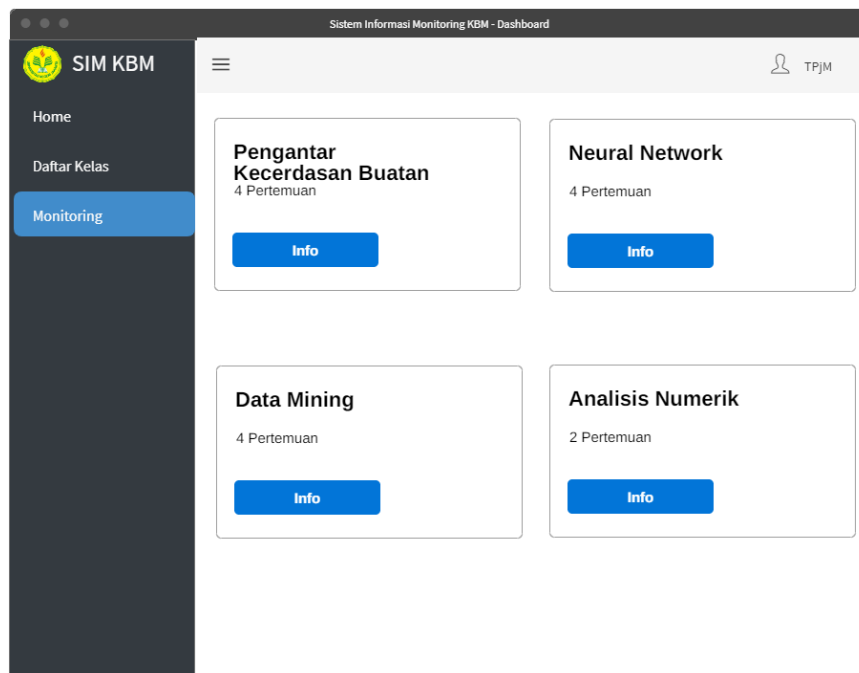
dapat melihat daftar semua kelas yang ada pada program studi dan jumlah pertemuan yang telah berjalan. TPjM juga dapat membuka halaman kelas tersebut untuk melihat informasi di dalamnya hanya untuk melakukan monitoring tanpa bisa berinteraksi kecuali jika kelas tersebut juga diampu oleh TPjM tersebut.



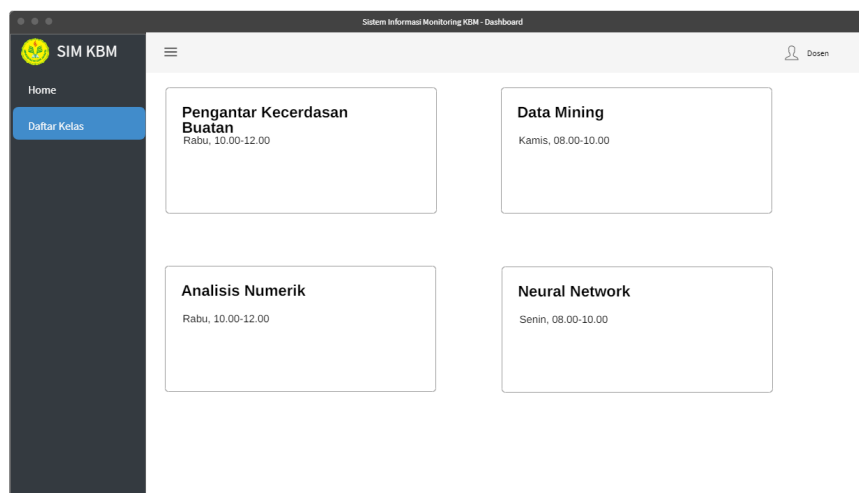
		Pertemuan		
#	Mata Kuliah	1	2	3
1	Pengantar Kecerdasan ...	✓	✓	✗
2	Neural Network	✓	✓	✓
3	Data Mining	✓	✓	✗

Gambar 3.12: Desain Halaman Rekap Presensi

Menu Daftar Kelas merupakan menu yang digunakan untuk membuka halaman kelas yang diambil. Halaman Daftar Kelas dapat dilihat pada gambar 3.14. Menu kelas terdapat empat bagian *tab* di dalamnya, yaitu Pertemuan, Presensi, Tugas, dan Nilai. Pada *tab* pertemuan yang dapat dilihat pada gambar 3.15 dosen dapat menambahkan pertemuan, memvalidasi form yang dibuat oleh penanggung jawab, memvalidasi presensi mahasiswa, dan mengisi pokok bahasan yang dibahas pada pertemuan tersebut. Setelah pertemuan sudah dibuat dan mahasiswa sudah mengisi presensi dosen juga bisa memvalidasi presensi mahasiswa tersebut. Pada *tab* ini mahasiswa dapat memilih Hadir untuk mengisi presensi pada pertemuan tersebut yang bisa dilihat pada gambar 3.16. Penanggung jawab mata kuliah memiliki fungsi yang sama dengan mahasiswa dan dosen.



Gambar 3.13: Desain Halaman *Monitoring*



Gambar 3.14: Desain Halaman Daftar Kelas

Pada *tab* presensi yang dapat dilihat pada gambar 3.17, sistem menampilkan tabel presensi dengan format form 06 dimana setiap presensi yang terekam valid akan ditandai dengan tanda centang, dan presensi kosong akan ditandai dengan tanda silang.

Pada *tab* tugas yang dapat dilihat pada gambar 3.18, dosen dapat menambah

Pertemuan	Hari/Tanggal	Materi	Jumlah Mahasiswa Hadir	Validasi	Action
1	Selasa 20/01/2021	Introduction	20	✓	
2	Selasa 28/01/2021	Quiz	20	✓	
3	Selasa 02/02/2021	Materi 2	20	Validasi Presensi	Validasi Edit Tutup Presensi

Gambar 3.15: Desain Halaman Form 05 Pada Dosen

Pertemuan	Hari/Tanggal	Pokok Bahasan	Jumlah Mahasiswa Hadir	Validasi PJ	Action
1	Selasa 20/01/2021	Introduction	20	✓	
2	Selasa 28/01/2021	Quiz	20	✓	
3	Selasa 02/02/2021	Materi 2	20	Hadir	

Gambar 3.16: Desain Halaman Form 05 Pada Mahasiswa

tugas yang diberikan pada mahasiswa atau mengubah tugas yang sudah dibuat sebelumnya. Pada *tab* ini mahasiswa tidak dapat berinteraksi dan hanya dapat melihat tabel tugas tersebut dengan tambahan informasi yaitu nilai pada tugas yang didapatkan tersebut. Tampilan halaman tugas untuk mahasiswa dapat dilihat pada gambar 3.19

Pada *tab* nilai yang dapat dilihat pada gambar 3.20, dosen dapat melihat tabel nilai setiap mahasiswa dan setiap nilai dari nilai tugas, UTS, dan UAS. Pada *tab* ini, dosen dapat mengunduh dan mengunggah nilai untuk memperbarui nilai pada sistem.

Sistem Informasi Monitoring KBM - Dashboard

Kelas Data Mining
Kamis, 08.00-10.00

Peremuan Presensi Tugas Nilai

NRM	Nama	1	2	3	4	5
3145161324	Aldi Rahmansyah	✓	✓			
3145161324	Aldi Rahmansyah	✓	✓			
3145161324	Aldi Rahmansyah	✗	✓			

Gambar 3.17: Desain Halaman Presensi

Sistem Informasi Monitoring KBM - Dashboard

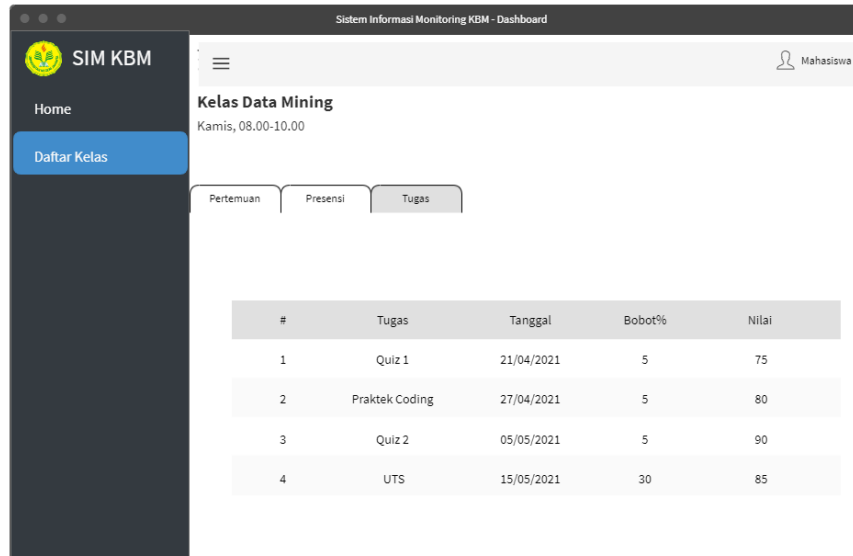
Kelas Data Mining
Kamis, 08.00-10.00

Peremuan Presensi Tugas Nilai

Tambah Tugas

#	Tugas	Tanggal	Bobot%	Action
1	Quiz 1	21/04/2021	5	<button>Edit</button>
2	Praktek Coding	27/04/2021	5	<button>Edit</button>
3	Quiz 2	05/05/2021	5	<button>Edit</button>
4	UTS	15/05/2021	30	<button>Edit</button>

Gambar 3.18: Desain Halaman Tugas Pada Dosen



Sistem Informasi Monitoring KBM - Dashboard

SIM KBM

Home


Daftar Kelas

Kelas Data Mining
Kamis, 08.00-10.00

Pertemuan Presensi Tugas

#	Tugas	Tanggal	Bobot%	Nilai
1	Quiz 1	21/04/2021	5	75
2	Praktek Coding	27/04/2021	5	80
3	Quiz 2	05/05/2021	5	90
4	UTS	15/05/2021	30	85


Gambar 3.19: Desain Halaman Tugas Pada Mahasiswa

SIM KBM


Home

Daftar Kelas

Sistem Informasi Monitoring KBM - Dashboard



Dosen



Kelas Data Mining

Kamis, 08.00-10.00

Pertemuan

Presensi

Tugas

Nilai

Download Excel

Upload Nilai

Ubah Bobot

NRM	Nama	Tugas					Nilai Akhir
		1	2	3	UTS	UAS	
3145161324	Aldi Rahmansyah	100	86	90	95		
3145161324	Aldi Rahmansyah	90	70	80	90		
3145161324	Aldi Rahmansyah	80	80	80	85		
3145161324	Aldi Rahmansyah	85	90	70	85		

Gambar 3.20: Desain Halaman Nilai Pada Dosen

DAFTAR PUSTAKA

- Alshamrani, A. & Bahattab, A. (2015). A Comparison Between Three SDLC Models Waterfall Model, Spiral Model, and Incremental/Iterative Model. *IJCSI International Journal of Computer Science Issues*, 12(1):106–111.
- Andini, A. F., Irzal, M., & Arafiah, R. (2017). Perancangan Dan Implementasi Sistem Absensi Online Berbasis Andriod Di Lingkungan Universitas Negeri Jakarta. *Jurnal Ilmu Komputer dan Aplikasi*, 1(1):1–10.
- Arora, R. & Arora, N. (2016). Analysis of SDLC Models. *International Journal of Current Engineering and Technology*, 6(1):2277–4106.
- bin Ladjamudin, A.-B. (2005). *Analisis dan Desain Sistem Informasi*. Graha Ilmu, Yogyakarta.
- Dora, S. K. & Dubey, P. (2013). Software Development Life Cycle (SDLC) Analytical Comparison And Survey On Traditional And Agile Methodology. *National Monthly Refereed Journal of Research in Science & Technology*.
- Dunn, W. N. (2014). *Public Policy Analysis*. Pearson Education, Harlow.
- Fauziah, Y. (2013). Aplikasi Iklan Baris Online menggunakan Arsitektur REST Web Service. *Telematika*, 9(2).
- FMIPA (2016). *Buku Panduan Akademik 2016/2017*.
- Fridayanthie, E. W. & Mahdiati, T. (2016). Rancang Bangun Sistem Informasi permintaan ATK Berbasis Intranet (Studi Kasus: Kejaksaan Negeri Rangkasbitung). *Jurnal Khatulistiwa Informatika*, IV(2):2.
- Hutahaean, J. (2015). *Konsep sistem informasi*. Deepublish.

- Jogiyanto, H. (2005). *Analisi dan Desain: Sistem Informasi: Pendekatan Terstruktur Teori dan Praktik Aplikasi Bisnis*. Andi, Yogyakarta.
- Kemendikbud (2020). Permendikbud No 3 Tahun 2020.
- Kroenke, D. M. & Boyle, R. J. (2009). *Using MIS*. Pearson Prentice Hall, New Jersey.
- Kultsum, U. (2021). *Rancang Bangun Sistem Presensi Akademik Berbasis Web Dengan Framework Laravel Di Lingkungan Program Studi Ilmu Komputer Universitas Negeri Jakarta*. Skripsi, Universitas Negeri Jakarta.
- Raharjo, B. (2011). *Belajar Otodidak Membuat Database Menggunakan Mysql*. Informatika, Bandung.
- Romney, M. B. & Steinbart, P. J. (2009). *Accounting Information Systems*. Pearson Education, New Jersey.
- Sudrajat, M. & Sudrajat, J. (2014). Pengembangan Sistem Informasi Kredit Usaha Rakyat Di PT. Bpr Nusamba Tanjungsari. *Jurnal Computech & Bisnis*, 8(2):80–91.
- Suendri (2018). Implementasi Diagram UML (Unified Modelling Language) Pada Perancangan Sistem Informasi Remunerasi Dosen Dengan Database Oracle (Studi Kasus: UIN Sumatera Utara Medan). *Jurnal Ilmu Komputer dan Informatika*, 3(1):1–9.
- Susanto, A. (2017). *Sistem Informasi Akuntansi: Pemahaman Konsep Secara Terpadu*. Lingga Jaya, Bandung.
- Widoyoko, E. (2009). Evaluasi Program Pembelajaran. *umpwr.ac.id*.
- Wirawan (2011). *Evaluasi : teori, model, standar, aplikasi, dan profesi*. Rajawali Press, Jakarta.

LAMPIRAN A

Analisis Kebutuhan (*User Requirement*)

1. Transkrip Wawancara Peneliti Dengan Koordinator Program Studi Ilmu Komputer FMIPA UNJ

Nama : Ir. Fariani Hermin Indiyah, MT
Jabatan : Koordinator Program Studi

Peneliti :	Bagaimana alur atau prosedur dalam evaluasi form 05 dan 06?
Narasumber :	Form 05 dan 06 selalu dievaluasi oleh TPjM (Tim Penjamin Mutu) program studi di awal, pertengahan dan di akhir semester. Karena di form 05 itu berisi tentang materi yang diajarkan di setiap pertemuan dan harus sesuai dengan RPS dan diharuskan minimal 80% dari total kehadiran yang diharuskan. Misal pertemuan 100% banyaknya 16 kali, maka dosen wajib mengisi form 05 min 80%nya. Evaluasi awal semester untuk mengecek kehadiran dosen di awal perkuliahan, di tengah untuk mengecek apakah pertemuan untuk UTS sudah dipenuhi dan evaluasi di akhir untuk melihat kehadiran dosen apakah memenuhi target minimal 80% itu. TPjM lalu melaporkan semua evaluasi dari program studi ke GPjM (Gugus Penjamin Mutu) dan ke koorprodi.
Peneliti :	Bagaimana jika nilai harus diinput ke dalam sistem mulai dari tugas, uts, uas dan transparan ke mahasiswa?
Narasumber :	Boleh, itu juga merupakan monitoring dari GPjM, seharusnya malah soal dan jawaban UTS dan UAS dikumpulkan di prodi.
Peneliti :	Lalu jika di setiap pertemuan kelas melakukan absen dari website, apakah Ibu bisa dan akan mengakses dari laptop atau smartphone?
Narasumber :	Bisa, yang praktis melalui smartphone.
Peneliti :	Untuk TPjM dan GPjM itu, anggotanya berubah per berapa waktu ya Bu?
Narasumber :	Surat Keterangan diperpanjang setiap tahun selama dosen yang bersangkutan masih bersedia.
Peneliti :	Bagaimana untuk alur form 06 atau penilaian yang dilakukan oleh dosen saat ini?
Narasumber :	Selama ini penilaian yang dilakukan oleh dosen dengan membuat tabel di excel yang terdiri dari nama, NIM, nilai tugas, UTS, UAS dengan bobot masing-masing lalu nilai angka dan nilai huruf yang kemudian diprint dan ditempel di form 06.
Peneliti :	Kalau untuk laporan dari TPjM itu isinya apa ya Bu?
Narasumber :	Nama dosen, mata kuliah yang diampu, tanggal pertemuan 1, hari, dan jam

2. Transkrip Wawancara Peneliti Dengan Ketua Gugus Penjamin Mutu FMIPA

Nama : Dra. Tritiyatma H, M.Si
Jabatan : Ketua Gugus Penjamin Mutu FMIPA

Peneliti	:	Untuk evaluasi form 05 dan 06, poin-poin apa saja yang menjadi pertimbangannya?
Narasumber	:	Dari form 05, diharapkan mendapatkan informasi bahwa dosen-dosen sudah menjalankan perkuliahan sesuai dengan RPS yang sudah dibuat. Dari form 06, diharapkan mendapatkan informasi tentang partisipasi mahasiswa dalam perkuliahan, mengerjakan tugas, mengikuti kuis, UTS dan UAS. Form 05 dan 06 akan selalu dipantau setiap 5 minggu. Jadi ada monev awal, tengah, dan akhir. Sehingga bila ada dosen yang belum memenuhi tugasnya, bisa segera diingatkan.
Peneliti	:	Kalau Ibu sendiri biasa mengabsen mahasiswa bagaimana Bu? Apakah Ibu absen satu-satu atau melalui penanggung jawab saja?
Narasumber	:	Kalau saat tatap muka dulu, ibu panggil satu-satu, sekalian menghafalkan wajahnya, kalau sekarang daring jadi bisa terlihat.
Peneliti	:	Bagaimana jika dosen memasukan nilai tugas, kuis, UTS, UAS ke dalam sistem dan nilai-nilai itu transparan ke mahasiswa?
Narasumber	:	Tidak masalah. Tapi sepertinya mau diakomodasi oleh siakad. Jadi nanti nilai UTS juga masuk siakad.
Peneliti	:	Setelah TPjM menyerahkan hasil evaluasinya ke GPjM, lalu apa tugas GPjM selanjutnya?
Narasumber	:	Tugas GPjM antara lain membantu fakultas menyiapkan borang akreditasi, menyiapkan SOP, dan memantau apakah prodi-prodi sudah menjalankan fungsinya sesuai standar yang ditetapkan.
Peneliti	:	Berarti jika ada dosen yang belum memenuhi tugas seperti sebelumnya disebutkan, masih ditangani oleh TPjM saja?
Narasumber	:	Iya, TPjM melapor ke koorprodi

3. Transkrip Wawancara Peneliti Dengan Tim Penjamin Mutu Program Studi Ilmu Komputer

Nama : Ria Arafiyah, M.Si
Jabatan : Tim Penjamin Mutu Program Studi Ilmu Komputer

Peneliti	:	Dalam menjalankan form 05 dan 06 di kelas, apakah Ibu membiarkan mahasiswa mengisi presensi sendiri atau Ibu yang memanggil satu persatu mahasiswa?
Narasumber	:	Cara mengabsen kondisional, bisa dipanggil satu-satu untuk mengenal dan memastikan kehadiran, atau random, atau menghitung siswa yang hadir, dll.
Peneliti	:	Apakah Ibu atau penanggung jawab kelas yang mengisi form 05?

Narasumber :	Penanggung jawab kelas yang mengisi dengan sepengetahuan dosen.
Peneliti :	Jika dalam penggunaan sistem, apakah Ibu dapat mengisi form 05 online di awal pertemuan atau membolehkan penanggung jawab yang mengisi form 05?
Narasumber :	Gambaran materi yang akan dipelajari kan ada pada RPS atau kontrak kuliah yang sudah dibahas oleh dosen diawal pertemuan. Isi form 05 adalah yang terjadi pada pertemuan tersebut. Saya kira bisa penanggung jawab mata kuliah yang mengisi dengan validasi dosen yang bersangkutan.
Peneliti :	Dalam sistem, ketika mahasiswa mengisi presensi apakah perlu ada validasi oleh penanggung jawab dan dosen, atau hanya perlu divalidasi oleh dosen?
Narasumber :	Bisa salah satunya, yang penting ada validatornya.
Peneliti :	Bagaimana menurut Ibu jika penilaian tugas, UTS, dan UAS dimasukan ke dalam sistem dan dapat dilihat oleh setiap mahasiswa?
Narasumber :	Baiknya hanya mahasiswa yang bersangkutan saja untuk beberapa pertimbangan.
Peneliti :	Pertimbangannya bagaimana ya Bu?
Narasumber :	Salah satunya ya kerahasiaan, privasi.
Peneliti :	Apakah lebih baik dibuat privasi semua, atau ketentuan dosen masing-masing untuk privasi atau tidaknya ya Bu?
Narasumber :	Bisa juga dibuat tampilan nilai terendah, tertinggi, sebaran nilai(plot) untuk melihat posisi mahasiswa yang bersangkutan tapi tidak ada nama mahasiswa yang lain. Jadi bisa memotivasi tanpa melanggar privasi.

4. Transkrip Wawancara Peneliti Dengan Tim Penjamin Mutu Program Studi Ilmu Komputer

Nama : Qorry Meidianingsih, M.Si
 Jabatan : Tim Penjamin Mutu Program Studi Pendidikan Matematika

Peneliti :	Bagaimana Ibu melakukan presensi perkuliahan biasanya, Apakah Ibu memanggil mahasiswa satu persatu atau membiarkan penanggung jawab mata kuliah yang mengisi?
Narasumber :	Biasanya saya membiarkan penanggung jawab mata kuliah untuk mengisi sesuai kehadiran di platform kuliahnya.
Peneliti :	Apakah Ibu juga membiarkan penanggung jawab kelas yang mengisi form 05?
Narasumber :	Iya betul, tapi nanti setiap akan dikumpulkan atau diminta tpjm selalu saya cek dulu. Jadi setiap perkuliahan saya selalu koordinasi mengenai materi dan berapa jumlah mahasiswa yang hadirnya.

Peneliti :	Jika dalam penggunaan sistem, apakah Ibu dapat mengisi form 05 online di awal pertemuan atau membolehkan penanggung jawab yang mengisi form 05 dengan validasi oleh Ibu?
Narasumber :	Pengalaman saya kendala yang biasa muncul ketika pengumpulan form 05 dan 06 itu ketika saya diminta memeriksa semua kelengkapan form 05 dan 06 itu sedikit ribet ketika sistemnya online seperti sekarang, harusnya justru segala sesuatu ketika digital lebih simpel kan tetapi malah jadi ribet. Kalau sistemnya bisa dibuat penanggung jawab yang mengisi lalu divalidasi oleh dosen mungkin itu akan lebih cepat karena dosen juga diminta untuk mengisi form monitoring perkuliahan, itu saja tidak semua dosen melaksanakan jadi untuk beberapa dosen senior berhadapan dengan form yang banyak mungkin lebih kesulitan ditambah unsur kesibukan gitu kan. Mungkin lebih baik kalau mahasiswa yang mengisi detailnya dengan kolom khusus misalnya dosen tersebut memverifikasi dan ada tombol signnya. Sehingga nanti terlihat misalnya mata kuliah apa yang sudah mengisi, sejauh mana, misalnya kan sekarang normalnya sudah pertemuan 12 jadi nanti saya bisa cek perkuliahan mana yang sudah sampai pertemuan 12. Saya bisa ceknya lebih mudah satu persatu jadi tidak menumpuk di akhir. Kalau numpuk di akhir saya harus cek dulu apakah ada paraf dosen yang belum diisi, paraf koorprodi. Jadi kalau ada sistemnya bagus, jadi saya tinggal mengecek saja apakah sudah sesuai pertemuannya dan apakah sudah terverifikasi oleh dosen. Kalau dosen yang mengisi semuanya rasanya agak lebih berat apalagi dosen senior biasanya lebih kesulitan, karena form monitoring yang kita bikin saja banyak yang tidak mengisi.
Peneliti :	Bagaimana pendapat Ibu jika penilaian tugas, UTS, dan UAS juga dimasukkan ke sistem dan dapat dilihat oleh mahasiswanya?
Narasumber :	TPjM juga menyimpan dokumen tersebut sebagai bukti akademik juga, dan setau saya mahasiswa diperkenankan bertanya kepada dosen pengampu jika ingin mengetahuinya. Untuk ahkrinya dipublikasikan atau dimasukkan ke sistem mungkin perlu kesepakatan dahulu dengan dosen lain, kalau saya tidak mempermasalahkan hal tersebut. Kalau saya mungkin prefer rekap nilai tersebut dapat dilihat hanya oleh mahasiswa yang bersangkutan. Artinya, misal kamu bisa lihat nilai, tugas, UTS, dan UASnya berapa. Tapi hanya milik kamu saja, bukan yang lain. Seperti melihat IPK ya jadinya.
Peneliti :	Untuk monitoring kira-kira poin apa saja yang ditampilkan pada rekap setiap perkuliahannya ya Bu?
Narasumber :	Form monitoring ini ada 2, ada untuk mahasiswa dan ada untuk dosen. Sebetulnya sama-sama tujuannya untuk melihat apakah perkuliahan sudah berjalan sebagaimana seharusnya, dilihat sebagaimana seharusnya tuh apakah jumlah pertemuan sudah

	<p>sesuai dengan seharusnya, kedua bisa dilihat lebih detail itu apakah materi yang dipelajari sudah sesuai dengan RPS atau belum, lalu di form juga nanti kita meminta upload RPS terbaru untuk arsip kita. Terus yang menjadi hal penting juga dari form monitoring itu terkait dengan metode pelajaran yang dilakukan karena UNJ fokus dengan case based dan proejct based learning itu kira-kira 50% dari seluruh mata kuliah jadi kita harus memantau mata kuliah mana yang pakai metode pembelajaran tersebut, saya juga ingin tau nih kita tim sebetulnya sebagian besar itu dosen menggunakan media pembelajarannya apa dari segi platform juga itu paling banyak menggunakan apa dan medianya apa. Apakah video, powerpoint, textbook atau podcasat itu juga ada rekapannya. Lalu kita juga ada meminta penilaian dari mahasiswa terhadap kinerja dosen dari materi yang diajarkan itu tingkat pemahamannya atau tingkat keefektifannya itu berapa persen, nanti itu bisa dilihat ada dosen yang lebih rendah dari dosen lain, ada yang lebih tinggi, ada yang konsisten dimanapun dia mengajar, itu juga penting untuk melihat penilaian berdasarkan mahasiswa. Karena bisa aja pertanyaan yang sama jawabannya berbeda dari dosen dan mahasiswa. Itu sih karena ada target akademik yang ditentukan oleh tim GPJM kita diminta untuk melihat mata kuliah yang melakukan metode pembelajarannya case based atau project based itu. Jadi itu adalah rekap yang dibutuhkan, biasanya ada monitoring awal, tengah, akhir biasanya disitu dilihat apakah sudah dilakukan perkuliahan sesuai pertemuan, media dan metode apa yang digunakan</p>
Peneliti	: Apakah ada fitur yang Ibu harapkan ada dalam sistem presensi atau monitoringnya sebagai dosen atau TPjM? Atau apakah ada hal yang harus diperhatikan atau menjadi concern dalam pembuatan sistem monitoring ini?
Narasumber	: Saya berharap sistemnya nanti dapat mengakomodir Penanggung Jawab mata kuliah untuk mengisinya dan dosen dapat memverifikasi secara mandiri agar prosesnya bisa lebih efisien. TPjM prodi diharapkan dapat diberi akses untuk melihatnya juga sehingga dapat memonitoring proses perkuliahan. Outputnya juga sebisa mungkin sama dengan output form 05 dan 06 yang diminta ya, kalau perlu verifikasiin lagi dengan koorprodi agar sekalian bisa diperoleh laporannya tanpa harus tanda tangan online lagi oleh koorprodi.