

**PENGEMBANGAN FITUR PENGELOLAAN
RENCANA PEMBELAJARAN SEMESTER (RPS)
PADA SISTEM PENJAMIN MUTU JURUSAN ILMU KOMPUTER**

Skripsi

Oleh

**ANNISA CITRA PRATIWI
NPM 2217051008**



**PROGRAM SARJANA ILMU KOMPUTER
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG
2026**

**PENGEMBANGAN FITUR PENGELOLAAN
RENCANA PEMBELAJARAN SEMESTER (RPS)
PADA SISTEM PENJAMIN MUTU JURUSAN ILMU KOMPUTER**

Oleh
ANNISA CITRA PRATIWI

Skripsi

**Sebagai Salah Satu Syarat untuk Mencapai Gelar
SARJANA ILMU KOMPUTER**

Pada

**Jurusan Ilmu Komputer
Fakultas Matematika dan Ilmu Pengetahuan Alam
Universitas Lampung**



**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG
2026**

ABSTRAK

PENGEMBANGAN FITUR PENGELOLAAN RENCANA PEMBELAJARAN SEMESTER (RPS) PADA SISTEM PENJAMIN MUTU JURUSAN ILMU KOMPUTER

Oleh

ANNISA CITRA PRATIWI

Pengelolaan Rencana Pembelajaran Semester (RPS) merupakan bagian penting dalam mendukung penjaminan mutu akademik di perguruan tinggi. Di Jurusan Ilmu Komputer Universitas Lampung, proses pengelolaan RPS masih dilakukan secara manual sehingga menimbulkan kendala dalam standarisasi format, pelacakan validasi, dan aksesibilitas dokumen. Penelitian ini bertujuan mengembangkan fitur pengelolaan RPS yang terintegrasi dalam Sistem Penjamin Mutu untuk mendukung proses penyusunan, pembaruan, validasi, dan penyajian RPS secara digital dan terpusat. Metode yang digunakan adalah *Rapid Application Development (RAD)*. Sistem dikembangkan berbasis web menggunakan *framework* Laravel dan MySQL. Pengujian fungsional dilakukan menggunakan metode *black-box testing*, sedangkan pengujian kinerja sistem dilakukan menggunakan Apache Jmeter. Hasil penelitian menunjukkan bahwa fitur pengelolaan RPS berhasil dikembangkan dan diintegrasikan ke dalam sistem penjamin mutu. Seluruh skenario pengujian fungsional memperoleh status lulus, dan pengujian kinerja menunjukkan sistem stabil di bawah beban pengguna. Sistem dinyatakan siap digunakan untuk mendukung pengelolaan RPS dan peningkatan mutu akademik secara berkelanjutan.

Kata Kunci : RPS, *Rapid Application Development*, Sistem Penjamin Mutu, *Black-Box Testing*, Apache JMeter.

ABSTRACT

DEVELOPMENT OF A COURSE SYLLABUS MANAGEMENT FEATURE IN A QUALITY ASSURANCE INFORMATION SYSTEM FOR THE COMPUTER SCIENCE DEPARTMENT

By

ANNISA CITRA PRATIWI

Course Syllabus (RPS) management plays an important role in supporting academic quality assurance in higher education. At the Computer Science Department of Universitas Lampung, the RPS management process was previously conducted manually, resulting in issues related to format standardization, validation tracking, and document accessibility. This study aims to develop an RPS management feature integrated into the Quality Assurance System to support the digital and centralized processes of drafting, updating, validating, and presenting RPS documents. The development method used in this research is Rapid Application Development (RAD). The system was implemented as a web-based application using the Laravel framework and MySQL database. Functional testing was conducted using the black-box testing method, while system performance testing was carried out using Apache JMeter. The results show that the RPS management feature was successfully developed and integrated into the quality assurance system. All functional test scenarios achieved a pass status, and performance testing demonstrated system stability under user load. The system is considered ready for implementation to support RPS management and continuous academic quality improvement.

Keywords: Course Syllabus, Rapid Application Development, Quality Assurance System, Black-Box Testing, Apache JMeter.

Judul Skripsi : **PENGEMBANGAN FITUR PENGELOLAAN RENCANA PEMBELAJARAN SEMESTER (RPS) PADA SISTEM PENJAMIN MUTU JURUSAN ILMU KOMPUTER**

Nama Mahasiswa : **Annisa Citra Pratiwi**

Nomor Pokok Mahasiswa : **2217051008**

Program Studi : **S1-Ilmu Komputer**

Fakultas : **Matematika dan Ilmu Pengetahuan Alam**



2. Mengetahui

[Handwritten Signature]

Dwi Sakethi, S.Si., M.Kom.
NIP. 19680611 199802 1 001

[Handwritten Signature]

Tristyanto, S.Kom., M.I.S., Ph.D
NIP. 19810414 200501 1 001

MENGESAHKAN

1. Tim Penguji

Ketua : Tristiyanto, S.Kom., M.I.S., Ph.D

Penguji

Penguji : Bambang Hermanto, S.Kom., M.Cs

Pembahas I

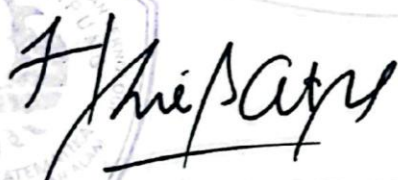
Penguji : Yunda Heningtyas, M.Kom

Pembahas II



2. Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam




Dr. Eng. Heri Satria, S.Si., M.Si.
NIP. 197110012005011002

Tanggal Lulus Ujian Skripsi: 07 April 2026

PERNYATAAN

Saya yang bertanda tangan di bawah ini:

Nama : Annisa Citra Pratiwi

NPM : 2217051008

Dengan ini menyatakan bahwa skripsi saya yang berjudul **“Pengembangan Fitur Pengelolaan Rencana Pembelajaran Semester (RPS) Pada Sistem Penjamin Mutu Jurusan Ilmu Komputer”** merupakan hasil karya saya sendiri dan bukan karya orang lain. Seluruh tulisan yang tertuang di skripsi ini telah mengikuti kaidah penulisan karya ilmiah Universitas Lampung. Apabila di kemudian hari terbukti skripsi saya merupakan hasil penjiplakan atau dibuat orang lain, maka saya bersedia menerima sanksi sesuai dengan hukum yang berlaku.

Bandar Lampung, 13 April 2026



Annisa Citra Pratiwi

NPM. 2217051008

RIWAYAT HIDUP



Penulis dilahirkan pada tanggal 15 Januari 2004 di Bandar Lampung sebagai anak kedua dari pasangan Bapak Kostalani dan Ibu Desi. Penulis menyelesaikan pendidikan formal pertama di Taman Kanak-kanak (TK) Kurnia Bandar Lampung pada Tahun 2010, kemudian melanjutkan pendidikan Sekolah Dasar (SD) di SD Negeri 1 Beringin Raya dan lulus pada Tahun 2016. Selanjutnya, penulis menempuh pendidikan di Sekolah Menengah Pertama (SMP) Negeri 2 Bandar Lampung dan lulus pada Tahun 2019, lalu melanjutkan ke Sekolah Menengah Atas (SMA) Negeri 7 Bandar Lampung dan menyelesaikannya pada Tahun 2022.

Pada Agustus 2022 penulis terdaftar sebagai mahasiswa Jurusan Ilmu Komputer, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Lampung melalui jalur SNMPTN. Selama menempuh pendidikan di perguruan tinggi, penulis melakukan beberapa kegiatan antara lain:

1. Menjadi Penerima Beasiswa Karya Salemba Empat Tahun 2024-2026.
2. Menjadi Asisten Dosen Jurusan Ilmu Komputer pada mata kuliah Logika, Basis Data, Pemrograman Desktop, Struktur Data Algoritma, dan Sistem Interaksi.
3. Menjadi Sekretaris Umum Paguyuban Karya Salemba Empat Universitas Lampung Periode 2025/2026.
4. Menjadi Sekretaris Bidang Pendidikan Riset & Teknologi Paguyuban Karya Salemba Empat Universitas Lampung Periode 2024/2025.
5. Menjadi Sekretaris Badan Khusus Himpunan Mahasiswa Jurusan Ilmu Komputer Periode 2023/2024.

6. Menjadi *Master of Ceremony* pada Kegiatan PKKMB Fakultas Matematika dan Ilmu Pengetahuan Alam Tahun 2024.
7. Melaksanakan Kerja Praktik Periode 2024/2025 di PT Bank Lampung.
8. Mengikuti Program MSIB *Batch 7* bersama Yayasan Dicoding Indonesia di Bidang *Cloud Computing* pada Tahun 2025.
9. Melaksanakan Kuliah Kerja Nyata (KKN) Periode I Universitas Lampung di Desa Suka Menanti Lampung Utara pada Tahun 2025.

MOTTO

فَإِنَّ مَعَ الْعُسْرِ يُسْرًا , إِنَّ مَعَ الْعُسْرِ يُسْرًا

*“Karena sesungguhnya bersama kesulitan ada kemudahan. Sesungguhnya
bersama kesulitan ada kemudahan.”*

(QS. Al-Insyirah: 5–6)

*“Yesterday is gone. Tomorrow has not yet arrived. Today is still unwritten.”
- Our Unwritten Soul 2025*

*“A real loser is somebody that is so afraid of not winning that they don't even try.
So, take a risk or lose the chance.”
- Dwayne Hoover*

*“Never let anyone including yourself, belittle your dream.”
- Joshua Hong*

PERSEMBAHAN

Alhamdulillahirobbilalamin

Puji Syukur kehadiran Allah Subhanahu Wa Ta'ala atas segala Rahmat dan karunia-Nya sehingga skripsi ini dapat diselesaikan dengan sebaik-baiknya.

Shalawat serta salam selalu tercurahkan kepada junjungan Nabi Agung Muhammad Shallallahu'Alaihi Wassalam.

Kupersembahkan karya ini kepada:

Kedua orang tua tercinta, sumber doa dan kekuatan yang tak pernah putus. Untuk para rekan yang telah kebersamai perjalanan ini dengan dukungan dan bantuan yang berarti. Serta untuk diri saya sendiri atas keberanian untuk terus melangkah hingga mampu menyelesaikan studi pada Program Studi S1 Ilmu Komputer Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung.

SANWACANA

Puji syukur kehadirat Allah SWT atas segala rahmat, nikmat, dan karunia-Nya sehingga penulis dapat menyelesaikan skripsi yang berjudul **“Pengembangan Fitur Pengelolaan Rencana Pembelajaran Semester (RPS) pada Sistem Penjamin Mutu Jurusan Ilmu Komputer”**. Dalam proses penyusunan skripsi ini, banyak pihak yang telah memberikan bantuan, dukungan, serta bimbingan kepada penulis. Oleh karena itu, dengan segala kerendahan hati dan rasa hormat, penulis mengucapkan terima kasih kepada:

1. Kedua orang tua penulis yaitu Bapak Kostalani dan Ibu Desi, yang selalu menjadi motivasi terbesar bagi penulis, serta senantiasa memberikan doa, dukungan, dan keyakinan kepada penulis hingga skripsi ini dapat diselesaikan.
2. Bapak Dr. Eng. Heri Satria, S.Si., M.Si. selaku Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung.
3. Bapak Dwi Sakethi, S.Si., M.Kom. selaku Ketua Jurusan Ilmu Komputer Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung.
4. Ibu Yunda Heningtyas, M.Kom. selaku Sekretaris Jurusan Ilmu Komputer sekaligus Dosen Pembahas yang telah memberikan kritik dan saran demi perbaikan skripsi ini.
5. Bapak Tristiyanto, S.Kom., M.I.S., Ph.D. selaku Ketua Program Studi S1-Ilmu Komputer sekaligus Dosen Pembimbing yang dengan penuh kesabaran membimbing penulis, serta memberikan dukungan, saran, dan kritik selama proses penyusunan skripsi ini.
6. Bapak M. Iqbal Parabi, S.Si., M.T. selaku Pembimbing Akademik yang senantiasa membimbing dan memberikan masukan selama masa perkuliahan.

7. Bapak Bambang Hermanto, S.Kom., M.Cs. selaku Dosen Pembahas yang telah memberikan kritik dan saran yang bermanfaat sehingga skripsi ini dapat diselesaikan dengan baik.
8. Bapak/Ibu Dosen dan Seluruh Staf Jurusan Ilmu Komputer FMIPA Universitas Lampung yang telah memberikan ilmu dan membantu dalam berbagai urusan akademik maupun administratif.
9. Kakak dan adik tercinta yaitu Ardira Bunga Pramesty, Alya Hendraning Prawestry dan Muhammad Radja Alghazali, yang selalu memberikan dukungan, motivasi, serta menjadi bagian penting dalam kehidupan penulis.
10. Eyang Uti yang selalu memberikan doa terbaik, mengharapkan kemudahan dan kelancaran bagi penulis dalam setiap langkah, termasuk dalam penyelesaian skripsi ini. Doa Eyang Uti yang penuh kasih telah menjadi sumber keberkahan dalam menyelesaikan skripsi ini.
11. Yayasan Karya Salemba Empat (KSE) yang telah memberikan beasiswa, motivasi, relasi serta berbagai program yang bermanfaat bagi penulis.
12. Teman-teman tersayang yaitu Laras, Tya, Via, Nazwa, Elen, Atikah dan Pradya yang selalu menjadi tempat berbagi canda, cerita dan pikiran selama masa perkuliahan. Terima kasih atas segala dukungan dan bantuan yang tak terhitung jumlahnya. Kehadiran kalian menjadi salah satu alasan penulis dapat menjalani dan menyelesaikan perkuliahan dengan baik.
13. Keluarga besar RT 17 dan Karang Taruna Perum Bhayangkara, Terima kasih atas kepercayaan, kebersamaan serta pengalaman yang telah membentuk penulis menjadi pribadi yang lebih berkembang. Semoga kita semua senantiasa dilancarkan dalam setiap langkah ke depan.
14. Kak Vidya, Meydina, Syauqii, Risnaz, Kak Edo dan Kak Widya yang selalu menjadi tempat berbagi cerita dan bertanya. Terima kasih atas dukungan dan energi positif yang selalu kalian berikan.
15. Paguyuban KSE Universitas Lampung beserta seluruh anggotanya atas pengalaman, kebersamaan dan dukungan yang telah diberikan selama penulis menjadi bagian dari keluarga besar ini.

16. Keluarga Ilmu Komputer 2022 yang telah menjadi bagian dari perjalanan penulis. Terima kasih atas kebersamaan dan dukungan selama masa perkuliahan di Jurusan Ilmu Komputer Universitas Lampung.
17. Seluruh pihak yang tidak dapat disebutkan satu per satu yang telah memberikan dukungan dan kontribusi dalam penyusunan skripsi ini, penulis mengucapkan terima kasih.

Penulis menyadari bahwa skripsi ini masih memiliki banyak kekurangan dan belum sepenuhnya sempurna. Meski demikian besar harapan penulis agar karya ini dapat memberikan manfaat bagi civitas akademika Universitas Lampung terkhusus mahasiswa Ilmu Komputer.

Bandar Lampung, 13 April 2026



Annisa Citra Pratiwi
NPM. 2217051008

DAFTAR ISI

	Halaman
DAFTAR ISI	xi
DAFTAR TABEL	xiv
DAFTAR GAMBAR	xv
DAFTAR KODE PROGRAM	xviii
I. PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Tujuan	2
1.4 Manfaat	3
1.5 Batasan Masalah	3
II. TINJAUAN PUSTAKA	4
2.1 Penelitian Terdahulu	4
2.2 Landasan Teori	9
2.2.1 Sistem Manajemen Mutu	9
2.2.2 Rencana Pembelajaran Semester (RPS)	10
2.2.3 PHP (<i>Hypertext Preprocessor</i>)	12
2.2.4 Laragon	13
2.2.5 Laravel	14
2.2.6 MySQL	15
2.2.7 <i>Unified Modeling Language</i> (UML)	15

2.2.8 Use Case Diagram.....	16
2.2.9 Activity Diagram.....	17
2.2.10 Class Diagram.....	18
2.2.11 Deployment Diagram.....	18
2.2.12 Entity Relationship Diagram (ERD).....	19
2.2.13 Software Development Life Cycle (SDLC).....	20
2.2.14 Rapid Application Development (RAD).....	20
2.2.15 Nilai Bisnis.....	22
2.2.16 Black-Box Testing.....	23
2.2.17 Apache Jmeter.....	23
III. METODOLOGI PENELITIAN.....	24
3.1 Waktu dan Tempat Penelitian.....	24
3.1.1 Waktu Penelitian.....	24
3.1.2 Tempat Penelitian.....	24
3.2 Tahapan Penelitian.....	25
3.3 Identifikasi Masalah.....	26
3.4 Studi Literatur.....	27
3.5 Perencanaan Kebutuhan (<i>Rapid Application Development</i>).....	28
3.5.1 Perangkat Penelitian.....	29
3.5.2 Ringkasan Eksekutif.....	29
3.5.3 Kebutuhan Bisnis.....	30
3.5.4 Kebutuhan Fungsional dan Non-Fungsional.....	31
3.5.5 Nilai Bisnis.....	32
3.5.6 Definisi dan Strategi Analisis Kebutuhan.....	33
3.5.7 Use Case Diagram.....	35
3.5.8 Spesifikasi Perangkat Lunak.....	53

3.5.9 Skenario Pengujian Fungsional	54
3.5.10 Skenario Pengujian Non-Fungsional	61
3.5.11 Validasi <i>Prototype</i>	63
3.5.12 <i>Activity Diagram</i>	63
3.5.13 <i>Class Diagram</i>	74
3.5.14 <i>Entity Relationship Diagram</i> (ERD).....	76
3.5.15 <i>Deployment Diagram</i>	78
3.5.16 Rancangan Tampilan Antarmuka Pengguna	78
IV. HASIL DAN PEMBAHASAN	89
4.1 Hasil.....	89
4.2 Validasi <i>Prototype</i>	90
4.3 Desain Sistem (<i>Rapid Application Development</i>)	90
4.3.1 <i>User Design 1/Input 1</i>	90
4.3.2 <i>User Design 2/Input 2</i>	95
4.3.3 <i>User Design 3/Input 3</i>	100
4.4 Pengembangan (<i>Rapid Application Development</i>).....	102
4.4.1 Pengembangan fitur Lihat Daftar RPS Publik dan Mengunduh RPS. 102	
4.4.2 Pengembangan fitur Mengelola RPS.....	106
4.4.3 Pengembangan fitur Mengelola Aktivitas.....	114
4.4.4 Pengembangan fitur Mengelola CPL ke MK.....	115
4.4.5 Pengembangan fitur Mengelola CPMK.....	117
4.4.6 Pengembangan fitur Mengelola Komponen	118
4.4.7 Pengembangan fitur Memvalidasi RPS	120
4.4.8 Pengembangan fitur Melihat Daftar RPS tervalidasi.....	123
4.5 Implementasi (<i>Rapid Application Development</i>)	125
4.6 Pembahasan	126

4.6.1 Pengujian Fungsional menggunakan <i>Black-box</i>	126
4.6.2 Pengujian Non-Fungsional	136
4.6.3 Keterbatasan Evaluasi Nilai Bisnis.....	141
V. SIMPULAN DAN SARAN	142
5.1 Kesimpulan.....	142
5.2 Saran.....	142
DAFTAR PUSTAKA	143
LAMPIRAN	147

DAFTAR TABEL

Tabel	Halaman
1. Tabel 2.1 Penelitian Terdahulu.....	4
2. Tabel 2.2 Komponen <i>Use Case</i>	16
3. Tabel 2.3 Komponen <i>Activity Diagram</i>	17
4. Tabel 2.4 Komponen <i>Entity Relationship Diagram (ERD)</i>	19
5. Tabel 3.1 Rencana Penelitian	24
6. Tabel 3.2 Tabel Kebutuhan Fungsional.....	31
7. Tabel 3.3 Tabel Kebutuhan Non Fungsional.....	32
8. Tabel 3.4 Matriks Kebutuhan Fungsional <i>As-Is</i> dan <i>To-Be</i>	34
9. Tabel 3.5 Matriks Kebutuhan Non-Fungsional <i>As-Is</i> dan <i>To-Be</i>	35
10. Tabel 3.6 Deskripsi <i>Use Case</i> Melihat Daftar RPS Publik.....	37
11. Tabel 3.7 Deskripsi <i>Use Case</i> Mengunduh RPS.....	38
12. Tabel 3.8 Deskripsi <i>Use Case</i> Mengelola RPS.....	39
13. Tabel 3.9 Deskripsi <i>Use Case</i> Mengelola Aktivitas.....	42
14. Tabel 3.10 Deskripsi <i>Use Case</i> Mengelola CPL ke MK.....	44
15. Tabel 3.11 Deskripsi <i>Use Case</i> Mengelola CPMK.....	46
16. Tabel 3.12 Deskripsi <i>Use Case</i> Mengelola Komponen	48
17. Tabel 3.13 Deskripsi <i>Use Case</i> Memvalidasi RPS	50
18. Tabel 3.14 Deskripsi <i>Use Case</i> Melihat Daftar RPS Tervalidasi.....	52
19. Tabel 3.15 Spesifikasi Perangkat Keras dan Perangkat Lunak.....	53
20. Tabel 3.16 Skenario Pengujian Fungsional.....	54
21. Tabel 3.17 Skenario Pengujian Non-Fungsional.....	61
22. Tabel 4.1 Hasil pengujian <i>black-box</i>	126
23. Tabel 4.2 Skenario Pengujian dengan Apache JMeter	138
24. Tabel 4.3 Hasil Uji Keseluruhan	140

DAFTAR GAMBAR

Gambar	Halaman
1. Gambar 2.1 Contoh Format RPS (APTIKOM, 2024)	11
2. Gambar 2.2 Contoh Format Rencana Pembelajaran Per Minggu.....	12
3. Gambar 2.3 Antarmuka Laragon.....	13
4. Gambar 2.4 Contoh Kode Laravel	14
5. Gambar 2.5 MySQL pada PHPMyAdmin.	15
6. Gambar 2.6 Tahapan <i>Rapid Application Development</i>	21
7. Gambar 3.1 Tahapan Penelitian	25
8. Gambar 3.2 <i>Use Case Diagram Global</i>	36
9. Gambar 3.3 <i>Use Case Diagram</i>	37
10. Gambar 3.4 <i>Activity Diagram</i> Melihat Daftar RPS Publik.....	64
11. Gambar 3.5 <i>Activity Diagram</i> Mengunduh RPS.....	65
12. Gambar 3.6 <i>Activity Diagram</i> Mengelola RPS.....	66
13. Gambar 3.7 <i>Activity Diagram</i> Mengelola Aktivitas.....	68
14. Gambar 3.8 <i>Activity Diagram</i> Mengelola CPLMK	69
15. Gambar 3.9 <i>Activity Diagram</i> Mengelola CPMK.....	70
16. Gambar 3.10 <i>Activity Diagram</i> Mengelola Komponen	71
17. Gambar 3.11 <i>Activity Diagram</i> Melihat Daftar RPS Tervalidasi.....	72
18. Gambar 3.12 <i>Activity Diagram</i> Memvalidasi RPS	73
19. Gambar 3.13 <i>Class Diagram</i> Fitur Rencana Pembelajaran Semester	75
20. Gambar 3.14 <i>Entity Relationship Diagram</i> Fitur Rencana Pembelajaran Semester.....	77
21. Gambar 3.15 <i>Deployment Diagram</i> Sistem Pengelolaan RPS	78
22. Gambar 3.16 Rancangan Tampilan Melihat Daftar RPS Publik.....	79
23. Gambar 3.17 Rancangan Tampilan Mengelola RPS bagian Daftar RPS.....	80
24. Gambar 3.18 Rancangan Tampilan Mengelola RPS bagian tambah RPS	81

25. Gambar 3.19 Rancangan Tampilan Mengelola RPS bagian edit RPS	81
26. Gambar 3.20 Rancangan Tampilan Mengelola RPS bagian <i>preview</i> RPS	82
27. Gambar 3.21 Rancangan Tampilan Mengelola Aktivitas bagian melihat daftar aktivitas mingguan	82
28. Gambar 3.22 Rancangan Tampilan Mengelola Aktivitas bagian <i>form</i> tambah aktivitas mingguan	83
29. Gambar 3.23 Rancangan Tampilan Mengelola Aktivitas bagian <i>form</i> edit aktivitas mingguan	83
30. Gambar 3.24 Rancangan Tampilan Mengelola CPLMK bagian melihat daftar CPLMK.....	84
31. Gambar 3.25 Rancangan Tampilan Mengelola CPLMK bagian <i>form</i> tambah CPLMK.....	85
32. Gambar 3.26 Rancangan Tampilan Mengelola CPMK bagian melihat daftar CPMK	85
33. Gambar 3.27 Rancangan Tampilan Mengelola CPMK bagian <i>form</i> tambah CPMK	86
34. Gambar 3.28 Rancangan Tampilan Mengelola Komponen bagian melihat daftar komponen penilaian.....	86
35. Gambar 3.29 Rancangan Tampilan Mengelola Komponen bagian <i>form</i> tambah komponen.....	87
36. Gambar 3.30 Rancangan Tampilan Memvalidasi RPS bagian melihat daftar RPS yang perlu di validasi	87
37. Gambar 3.31 Rancangan Tampilan Memvalidasi RPS bagian <i>form</i> tolak validasi RPS	87
38. Gambar 3.32 Rancangan Tampilan Melihat Daftar RPS Tervalidasi.....	88
39. Gambar 4.1 <i>Prototype</i> Melihat Daftar RPS Publik dan Mengunduh RPS.....	91
40. Gambar 4.2 <i>Prototype</i> Mengelola RPS	92
41. Gambar 4.3 <i>Prototype</i> Mengelola RPS bagian Tambah RPS	92
42. Gambar 4.4 <i>Prototype</i> Mengelola Aktivitas.....	93
43. Gambar 4.5 <i>Prototype</i> Mengelola Aktivitas bagian Tambah Aktivitas.....	93
44. Gambar 4.6 <i>Prototype</i> Mengelola CPL ke MK.....	97
45. Gambar 4.7 <i>Prototype</i> Daftar Pemetaan CPL ke Mata Kuliah	97

46. Gambar 4.8 <i>Prototype</i> Mengelola CPMK.....	98
47. Gambar 4.9 <i>Prototype</i> Daftar CPMK dan Sub CPMK	98
48. Gambar 4.10 <i>Prototype</i> Mengelola Komponen	98
49. Gambar 4.11 <i>Prototype</i> Daftar Kriteria Penilaian.....	99
50. Gambar 4.12 <i>Prototype</i> Memvalidasi RPS	101
51. Gambar 4.13 <i>Prototype</i> Daftar RPS untuk di Validasi.....	101
52. Gambar 4.14 Implementasi Halaman Melihat Daftar RPS Publik dan Mengunduh RPS	104
53. Gambar 4.15 Implementasi Halaman Hasil Unduh RPS (1)	105
54. Gambar 4.16 Implementasi Halaman Hasil Unduh RPS (2)	106
55. Gambar 4.17 Implementasi Halaman Mengelola RPS bagian Tambah RPS	112
56. Gambar 4.18 Implementasi Halaman Mengelola RPS bagian daftar RPS ...	112
57. Gambar 4.19 Implementasi Halaman Mengelola RPS bagian cetak RPS	113
58. Gambar 4.20 Implementasi Halaman Mengelola RPS bagian edit RPS.....	113
59. Gambar 4.21 Implementasi Halaman Mengelola Aktivitas	115
60. Gambar 4.22 Implementasi Halaman Mengelola CPL ke MK.....	116
61. Gambar 4.23 Implementasi Halaman Mengelola CPMK.....	118
62. Gambar 4.24 Implementasi halaman Mengelola Komponen bagian tambah komponen.....	119
63. Gambar 4.25 Implementasi halaman Mengelola Komponen bagian daftar komponen.....	120
64. Gambar 4.26 Implementasi <i>pop-up form</i> catatan penolakan RPS	122
65. Gambar 4.27 Implementasi notifikasi setelah penolakan RPS	122
66. Gambar 4.28 Implementasi notifikasi setelah penyetujuan RPS	123
67. Gambar 4.29 Implementasi Halaman Melihat Daftar RPS Tervalidasi	124
68. Gambar 4.30 Struktur <i>Test Plan</i> pada JMeter.....	137
69. Gambar 4.31 Konfigurasi <i>Thread Group</i> Pengujian.....	138
70. Gambar 4.32 <i>Summary Report</i> Pengujian Skenario 1	139
71. Gambar 4.33 <i>Summary Report</i> Pengujian Skenario 2	139
72. Gambar 4.34 <i>Summary Report</i> Pengujian Skenario 3	139

DAFTAR KODE PROGRAM

Kode Program	Halaman
1. Kode Program 4.1 Melihat Daftar RPS Publik (1)	102
2. Kode Program 4.2 Melihat Daftar RPS Publik (2)	103
3. Kode Program 4.3 Mengunduh RPS.....	104
4. Kode Program 4.4 Mengelola RPS bagian tambah RPS (1).....	107
5. Kode Program 4.5 Mengelola RPS bagian tambah RPS (2).....	107
6. Kode Program 4.6 Mengelola RPS bagian tambah RPS (3).....	108
7. Kode Program 4.7 Mengelola RPS bagian tambah RPS (4).....	108
8. Kode Program 4.8 Mengelola RPS bagian melihat daftar RPS	109
9. Kode Program 4.9 Mengelola RPS bagian cetak RPS (1)	110
10. Kode Program 4.10 Mengelola RPS bagian cetak RPS (2)	110
11. Kode Program 4.11 Mengelola RPS bagian cetak RPS (3)	111
12. Kode Program 4.12 Mengelola Aktivitas bagian tambah aktivitas.....	114
13. Kode Program 4.13 Mengelola CPL ke MK bagian tambah CPL ke MK....	116
14. Kode Program 4.14 Mengelola CPMK bagian tambah CPMK	117
15. Kode Program 4.15 Mengelola Komponen bagian daftar komponen	118
16. Kode Program 4.16 Mengelola Komponen bagian tambah komponen	119
17. Kode Program 4.17 Memvalidasi RPS	121
18. Kode Program 4.18 Memvalidasi RPS opsi disetujui.....	121
19. Kode Program 4.19 Memvalidasi RPS opsi ditolak	121
20. Kode Program 4.20 Melihat Daftar RPS Tervalidasi (1)	123
21. Kode Program 4.21 Melihat Daftar RPS Tervalidasi (2)	124

I. PENDAHULUAN

1.1 Latar Belakang

Di tengah persaingan global yang semakin ketat, mutu menjadi fondasi utama dalam penyelenggaraan pendidikan tinggi. Perguruan tinggi tidak lagi cukup hanya menjalankan fungsi akademik, melainkan juga harus mampu membuktikan kualitas layanannya melalui sistem manajemen yang terstruktur dan berorientasi pada peningkatan berkelanjutan. Oleh karena itu, penguatan sistem penjaminan mutu internal diperlukan agar standar akademik dan administratif dapat diterapkan secara konsisten serta dipertanggungjawabkan kepada para pemangku kepentingan.

Upaya peningkatan mutu tersebut tidak hanya berfokus pada pencapaian standar yang tinggi, tetapi juga pada bagaimana mutu itu dapat dirasakan secara merata. Pemerataan mutu dalam perguruan tinggi merupakan salah satu agenda bangsa Indonesia bila menginginkan kualitas mahasiswa tinggi, bukan hanya memberikan perluasan kesempatan (*equity*) dan meningkatkan mutu (*quality*) sebagai dua hal yang terpisah, melainkan pemerataan mutu (*equity of quality*) (Razak dkk., 2016). Namun demikian, pada kenyataannya salah satu permasalahan paling krusial dalam dunia pendidikan adalah rendahnya mutu pendidikan di berbagai jenis dan jenjang pendidikan (Noly dkk., 2022). Salah satu indikator penting dalam menjamin mutu pendidikan adalah keberadaan dokumen Rencana Pembelajaran Semester (RPS) yang tersusun secara sistematis, lengkap, dan mudah diakses oleh berbagai pihak, khususnya dosen, kepala program studi, dan penjamin mutu. RPS berfungsi sebagai pedoman utama dalam penyelenggaraan proses pembelajaran suatu mata kuliah selama satu semester. Oleh karena itu, penting untuk memastikan kualitas RPS melalui pengawasan dan evaluasi yang tepat. Perkuliahan yang dilaksanakan tanpa adanya RPS dapat dianggap sebagai kegiatan pembelajaran yang tidak terencana dengan baik (Aziira dkk., 2023).

Digitalisasi dokumen akademik seperti RPS dapat meningkatkan efisiensi kerja dosen, mengurangi kesalahan manusia (*human error*), serta mempercepat proses pengambilan keputusan di tingkat jurusan (Naufal, 2023). Selain itu, sistem digital juga memudahkan penjamin mutu dalam melakukan audit dan evaluasi, sehingga kualitas pembelajaran dapat terus ditingkatkan secara berkelanjutan.

Di lingkungan Jurusan Ilmu Komputer, proses pengelolaan RPS selama ini masih menghadapi beberapa kendala, seperti proses pengumpulan yang manual, ketidakteraturan format, hingga kesulitan dalam pelacakan revisi dan validasi oleh kepala program studi atau penjamin mutu. Kondisi ini menghambat efektivitas sistem penjaminan mutu internal, serta berdampak pada ketidakterpaduan antara perencanaan pembelajaran dengan pelaksanaan evaluasi mutu. Dalam rangka mendukung peningkatan mutu akademik jurusan, dibutuhkan solusi digital yang mampu mengintegrasikan, mendokumentasikan, serta mempermudah proses pengelolaan Rencana Pembelajaran Semester (RPS) secara efisien. Oleh karena itu, pengembangan fitur Pengelolaan RPS yang dirancang untuk menangani proses tersebut secara menyeluruh menjadi fokus utama dalam skripsi ini.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dijelaskan, penelitian ini berfokus pada pengembangan fitur pengelolaan Rencana Pembelajaran Semester (RPS) untuk mendukung sistem penjaminan mutu di lingkungan Jurusan Ilmu Komputer. Fitur ini dirancang untuk memfasilitasi proses penyusunan, pembaruan, dan dokumentasi RPS secara terintegrasi melalui sistem informasi berbasis web, sehingga Dosen, Kepala Program Studi, dan Penjamin Mutu Program Studi dapat mengakses dan memantau RPS dengan lebih efisien dan terdokumentasi dengan baik.

1.3 Tujuan

Adapun tujuan dari penelitian ini adalah:

- a. Mengembangkan fitur pengelolaan Rencana Pembelajaran Semester (RPS) yang memfasilitasi keseluruhan alur kerja, mulai dari penyusunan, pembaruan, penyimpanan, pemantauan, hingga validasi secara digital dan terpusat.
- b. Menyediakan fitur penyajian RPS dalam format PDF untuk keperluan administrasi.

1.4 Manfaat

Adapun manfaat dari penelitian ini antara lain sebagai berikut:

- a. Bagi Jurusan Ilmu Komputer, sistem yang dikembangkan akan mempermudah pengelolaan dan pendokumentasian Rencana Pembelajaran Semester secara terintegrasi dan efisien, sehingga mendukung penjaminan mutu.
- b. Bagi Dosen, sistem ini dapat memudahkan dalam menyusun, memperbarui, dan mengakses RPS kapan saja dan di mana saja.

1.5 Batasan Masalah

Adapun batasan masalah dalam penelitian ini adalah:

- a. Sistem dikembangkan menggunakan *framework* Laravel dan diimplementasikan sebagai aplikasi web.
- b. Pengembangan hanya mencakup fitur penyusunan, pembaruan, penyimpanan, dan penyajian RPS dalam format PDF, tanpa membahas fitur lain seperti rubrik penilaian, pengelolaan nilai ataupun evaluasi pembelajaran.

II. TINJAUAN PUSTAKA

2.1 Penelitian Terdahulu

Kajian terhadap penelitian sebelumnya dilakukan untuk membandingkan hasil-hasil yang telah dicapai dengan fokus penelitian saat ini. Penelitian-penelitian tersebut juga menjadi dasar dalam menelusuri referensi literatur yang mendukung. Ringkasan dari penelitian terdahulu ditampilkan pada Tabel 2.1.

Tabel 2.1 Penelitian Terdahulu

No.	Judul Penelitian	Metode	Hasil Penelitian	Research Gap
1.	Perancangan Aplikasi Pengembangan Rencana Pembelajaran Semester Dengan Menggunakan Metode <i>Prototype</i> (Syarif A, 2024).	<i>Prototype</i>	Hasil dari penelitian ini adalah sebuah aplikasi RPS (Rencana Pembelajaran Semester) berbasis <i>website</i> yang memfasilitasi pengelolaan Profil Lulusan, CPL, CPMK, Sub-CPMK, Bahan Kajian, Mata Kuliah, serta pembuatan dan pencetakan dokumen RPS sesuai format OBE/KKNI/SKKNI. Sistem membantu mempercepat pembuatan, penyimpanan, dan distribusi RPS kepada mahasiswa.	Pengembangan masih berfokus pada pengelolaan dan pembuatan RPS. Integrasi dengan mekanisme validasi berjenjang, serta siklus penjaminan mutu belum menjadi fokus utama..
2.	Rancang Bangun E-RPS Prodi Teknik Informatika IIB Darmajaya Berbasis Web (Naufal, 2023)	<i>Waterfall</i>	Hasil penelitian ini menghasilkan sebuah sistem E-RPS berbasis web yang berfungsi sebagai wadah bagi dosen Program Studi Teknik Informatika dalam melakukan input, pembaruan, dan pengelolaan RPS secara langsung	Sistem berfungsi sebagai wadah pengelolaan dan publikasi RPS, namun belum menerapkan mekanisme validasi formal berbasis peran

No.	Judul Penelitian	Metode	Hasil Penelitian	Research Gap
			di dalam <i>website</i> . Sistem ini dilengkapi kemudahan akses bagi mahasiswa tanpa perlu login. Berdasarkan pengujian fungsi dan antarmuka, website berjalan sesuai yang diharapkan dan mempermudah proses pengelolaan RPS dibandingkan metode sebelumnya yang masih berbasis <i>hardcopy</i> atau e-arsip.	(<i>role-based validation</i>) sebagai alat kendali mutu. Belum terintegrasi langsung ke dalam kerangka Sistem Penjaminan Mutu Internal.
3.	Pengembangan Modul Rencana Pembelajaran Semester Pada Sistem Informasi Portal Prodi Jurusan Teknik Elektro (Pramudya, 2023)	<i>Rapid Application Development</i>	Hasil penelitian ini adalah pengembangan modul Rencana Pembelajaran Semester (RPS) yang telah berhasil dibuat dengan beberapa fitur utama, yaitu mengajukan RPS, melakukan validasi RPS, melihat RPS, serta mencetak RPS dengan mengubahnya menjadi format PDF. Berdasarkan pengujian yang dilakukan menggunakan metode <i>blackbox testing</i> , modul ini terbukti berjalan dengan baik secara fungsional dengan hasil 100% OK dan sesuai dengan yang diharapkan.	Beberapa komponen RPS seperti CPL dan CPMK masih diinput manual melalui <i>database</i> , sehingga belum sepenuhnya mendukung pengelolaan komponen RPS secara dinamis dan terpadu melalui sistem.
4	Pembangunan Sistem Informasi Pengelolaan Rencana Pembelajaran Semester (RPS) (Aziira dkk., 2023)	Waterfall	Hasil penelitian ini adalah berhasil dibangunnya Sistem Informasi Pengelolaan Rencana Pembelajaran Semester (RPS) berbasis web dengan arsitektur MERN (MongoDB, Express.js, React.js, dan Node.js) yang mampu mendukung penyusunan RPS secara terstruktur, terintegrasi, dan terdokumentasi dengan baik. Sistem yang	Meskipun telah mengintegrasikan komponen pembelajaran, sistem belum dirancang sebagai bagian dari mekanisme SPMI. Tidak dijelaskan secara rinci alur validasi berjenjang dan kontrol mutu

No.	Judul Penelitian	Metode	Hasil Penelitian	Research Gap
			dikembangkan memungkinkan dosen untuk membuat, mengelola, memperbarui, serta mengarsipkan komponen RPS seperti CPMK, referensi, komponen penilaian, dan pertemuan mingguan secara online.	sebelum publikasi RPS.
5	Perancangan Aplikasi Rencana Pembelajaran Semester (RPS) Untuk Meningkatkan Pencapaian Pembelajaran Bagi Dosen (Syafarina & Setiawan, 2019)	Studi Kasus & SDLC	Hasil penelitian ini adalah aplikasi RPS berbasis web untuk mengatasi masalah format RPS yang tidak seragam karena dibuat dengan <i>Microsoft Word</i> dan kesulitan dalam perekapan berkas untuk keperluan akreditasi. Hasil evaluasi penggunaan sistem menunjukkan respon positif dari dosen.	Sistem berfokus pada standarisasi format dan penyimpanan dokumen RPS, belum mengintegrasikan mekanisme validasi formal oleh Kaprodi atau penjamin mutu. RPS masih berfungsi sebagai dokumen administratif, bukan sebagai instrumen pengendalian mutu dalam siklus SPMI.

Penelitian tahun 2024 yang dilakukan oleh Muh. Ridwan Syarif A dari Universitas Islam Sultan Agung merancang aplikasi pengembangan Rencana Pembelajaran Semester (RPS) berbasis *web* untuk menggantikan sistem administrasi kurikulum yang masih berbasis kertas. Permasalahan utama yang diangkat adalah risiko kerusakan dan kehilangan dokumen, serta lambatnya proses pencarian dan distribusi RPS kepada mahasiswa. Penelitian ini menggunakan metode *prototype* untuk mempercepat proses pengembangan dan memastikan sistem sesuai dengan kebutuhan program studi. Hasil penelitian berupa aplikasi yang mendukung pengelolaan Profil Lulusan, CPL, CPMK, Sub-CPMK, Bahan Kajian, Mata Kuliah, serta pembuatan dan pencetakan RPS sesuai standar OBE/KKNI/SKKNI. Sistem juga memungkinkan penyimpanan beberapa versi RPS dan mempermudah distribusinya. Pengujian menggunakan metode *black box testing* menunjukkan seluruh fungsi sistem berjalan sesuai dengan yang diharapkan.

Penelitian yang dilakukan pada tahun 2023 berjudul “Rancang Bangun E-RPS Prodi Teknik Informatika IIB Darmajaya Berbasis Web” bertujuan untuk mengatasi kendala pengelolaan RPS yang sebelumnya masih menggunakan *hardcopy*, e-arsip KRS, dan penggabungan RPS seluruh jurusan dalam satu *website*. Sistem ini dikembangkan menggunakan *framework* CodeIgniter dengan bahasa pemrograman PHP dan basis data MySQL melalui tahapan analisis kebutuhan, perancangan, pengembangan, implementasi, dan evaluasi. Hasil penelitian menghasilkan sistem E-RPS yang memungkinkan dosen melakukan input, pembaruan, dan pengelolaan RPS serta kegiatan pertemuan secara langsung di *website*, admin mengelola data dosen, mata kuliah, dan hak akses, serta mahasiswa mengakses RPS tanpa login. Pengujian menunjukkan seluruh fungsi berjalan sesuai harapan dan evaluasi menyatakan sistem layak digunakan, dengan kelebihan berupa tampilan *user-friendly*, akses fleksibel, dan penyimpanan terpusat, namun masih memiliki kekurangan seperti ketergantungan pada koneksi internet, belum adanya filter peminatan, dan belum adanya tanda tangan digital pada RPS.

Penelitian tahun 2023 yang dilakukan oleh Rivian Herdian Pramudya, berfokus pada pengembangan modul Rencana Pembelajaran Semester (RPS) pada Sistem Informasi Portal Program Studi Jurusan Teknik Elektro. Latar belakang penelitian ini adalah kebijakan kementerian pada tahun 2020 mengenai Indikator Kinerja Utama (IKU) perguruan tinggi, di mana RPS menjadi salah satu bukti utama untuk memenuhi IKU 7, yaitu Kelas Kolaboratif dan Partisipatif. Untuk itu, dikembangkan sebuah modul guna mengefektifkan dan menstandarkan pembuatan RPS yang sebelumnya dilakukan secara manual menggunakan *Microsoft Word*. Metode yang digunakan dalam pengembangan sistem ini adalah *Rapid Application Development* (RAD). Hasil dari penelitian ini adalah sebuah modul dengan fitur-fitur utama seperti mengajukan, memvalidasi, melihat, dan mencetak RPS ke dalam format PDF. Penelitian ini melakukan pengujian sistem menggunakan metode *black box testing*. Hasil pengujian menunjukkan bahwa seluruh modul telah berfungsi dengan baik dan sesuai dengan yang diharapkan, dengan tingkat keberhasilan mencapai 100%. Penelitian ini juga menemukan bahwa sistem portal prodi masih memerlukan pengembangan lebih lanjut, terutama pada menu penambahan data

Capaian Pembelajaran Prodi dan sinkronisasi data mata kuliah yang saat ini masih dilakukan secara manual.

Penelitian tahun 2023 yang dilakukan oleh Aina Hubby Aziira, Husnil Kamil, dan Afriyanti Dwi Kartika, berfokus pada pembangunan Sistem Informasi Pengelolaan Rencana Pembelajaran Semester (RPS) di Program Studi Sistem Informasi Universitas Andalas. Penelitian ini dilatarbelakangi oleh pengelolaan RPS yang masih dilakukan secara konvensional menggunakan aplikasi pengolah kata sehingga belum terstruktur dan belum memiliki sistem pengarsipan terpusat. Penelitian menggunakan metode SDLC dengan pendekatan Waterfall yang meliputi tahap analisis kebutuhan, perancangan sistem menggunakan UML, implementasi dengan memanfaatkan arsitektur MERN Stack, serta pengujian menggunakan metode black box testing. Hasil penelitian menunjukkan bahwa sistem yang dibangun mampu menyeragamkan format RPS, mengintegrasikan keterkaitan antara CPL dan CPMK, serta menyediakan penyimpanan dokumen RPS yang terstruktur dan aman sehingga mendukung efektivitas pengelolaan kurikulum dan evaluasi pembelajaran.

Penelitian yang dilakukan oleh Gita Ayu Syafarina dan Agus Setiawan pada tahun 2019 berfokus pada perancangan aplikasi RPS untuk meningkatkan pencapaian pembelajaran bagi dosen di Universitas Islam Kalimantan (UNISKA) Muhammad Arsyad Al Banjary. Latar belakang utama penelitian ini adalah proses pembuatan RPS yang masih manual menggunakan *Microsoft Word*, mengakibatkan ketidakseragaman format antar dosen dan menyulitkan proses perekapan berkas RPS oleh fakultas, terutama untuk keperluan akreditasi. Untuk mengatasi permasalahan ini, penelitian tersebut merancang sebuah sistem aplikasi berbasis web menggunakan tahapan *Software Development Life Cycle (SDLC)*. Hasil analisis kebutuhan melalui kuesioner menunjukkan bahwa 70% responden (dosen) menganggap perlu adanya pengembangan sistem yang terintegrasi dari sistem konvensional. Setelah implementasi, evaluasi akhir menunjukkan respon yang sangat positif (rata-rata 96%), mengindikasikan bahwa aplikasi RPS berbasis *online* tersebut layak digunakan untuk memudahkan dosen dalam pengisian dan membantu fakultas dalam perekapan berkas.

Dari analisis kelima penelitian terdahulu tersebut, teridentifikasi beberapa celah penelitian (*research gap*) kolektif yang mendasari urgensi pengembangan fitur dalam penelitian ini. Pertama, mayoritas sistem yang ada dikembangkan sebagai aplikasi mandiri atau modul portal prodi umum, belum secara spesifik terintegrasi sebagai komponen fungsional dalam Sistem Penjaminan Mutu jurusan, sehingga keterkaitannya dengan siklus mutu berkelanjutan belum optimal. Kedua, implementasi alur validasi formal oleh pemangku kepentingan seperti Kaprodi sebagai syarat publikasi RPS kepada mahasiswa seringkali belum menjadi fokus utama atau belum diuraikan secara rinci. Ketiga, terdapat keterbatasan dalam pengelolaan komponen pembelajaran secara komprehensif dan terintegrasi, di mana elemen-elemen detail seperti CPL, CPMK, asesmen, aktivitas mingguan, dan pemetaannya belum sepenuhnya dapat dikelola secara terpadu dalam satu *platform* RPS, bahkan terkadang masih memerlukan proses manual di luar sistem.

2.2 Landasan Teori

Berikut ini merupakan teori yang memiliki keterkaitan dengan penelitian yang akan dilakukan.

2.2.1 Sistem Manajemen Mutu

Sistem Manajemen Mutu merupakan suatu pendekatan sistematis yang dirancang untuk mengarahkan dan mengendalikan organisasi dalam menjamin tercapainya peningkatan mutu secara berkelanjutan. Sistem ini berfungsi sebagai kerangka kerja yang terdiri dari kebijakan, prosedur, proses, dan sumber daya yang saling berhubungan untuk memastikan bahwa produk atau jasa yang dihasilkan mampu memenuhi persyaratan serta harapan pengguna secara konsisten. Selain berfungsi sebagai alat peningkatan kualitas, sistem ini juga berperan sebagai mekanisme pengendalian dan pengawasan kegiatan internal organisasi agar target mutu dapat dicapai dan dipertahankan secara berkelanjutan (Farhan Azfa & Solihah, 2023).

Penerapan Sistem Manajemen Mutu dilakukan melalui siklus perencanaan, pelaksanaan, evaluasi, dan tindak lanjut sebagai upaya peningkatan kinerja proses dan sumber daya secara berkesinambungan. Pendekatan ini mengadaptasi prinsip *Total Quality Management* (TQM) yang menekankan pada perbaikan

berkelanjutan, keterlibatan seluruh pemangku kepentingan, serta orientasi pada kepuasan pengguna layanan (Sakdiyah, 2025).

2.2.2 Rencana Pembelajaran Semester (RPS)

RPS adalah salah satu perangkat pembelajaran yang berupa rancangan pembelajaran untuk semua Mata Kuliah, disertai perangkat pembelajaran lain seperti rencana tugas, instrumen penilaian dalam bentuk rubrik dan/atau portofolio, bahan ajar, dan lain-lain. RPS dirancang untuk menghasilkan lulusan yang memiliki kemampuan sesuai CPL yang telah ditetapkan, sehingga harus dapat dijalankan oleh mahasiswa pada setiap tahapan belajar pada mata kuliah terkait. RPS dititikberatkan pada bagaimana memandu mahasiswa untuk belajar agar memiliki kemampuan sesuai dengan CPL yang dibebankan pada mata kuliah, bukan pada kepentingan kegiatan dosen mengajar. Pembelajaran yang dirancang dalam RPS adalah pembelajaran yang berpusat pada mahasiswa (*Student Centered Learning*). Format RPS Mencakup Identitas dan otorisasi MK, Deskripsi Singkat MK, Capaian Pembelajaran Lulusan (CPL), Capaian Pembelajaran Mata Kuliah (CPMK), Sub CPMK, Bahan Kajian / Materi Pembelajaran, Bentuk Pembelajaran, Metode Pembelajaran, Estimasi Waktu, Pengalaman Belajar, Kriteria, Indikator Penilaian, Bobot Penilaian, dan Daftar Referensi (APTİKOM, 2024).

*LOGO UNIVERSITAS	RENCANA PEMBELAJARAN SEMESTER PROGRAM STUDI S1 INFORMATIKA/ILMU KOMPUTER FAKULTAS INFORMATIKA *NAMA UNIVERSITAS						
Identitas Mata Kuliah	NAMA MK	KODE MK	RUMPUN MATA KULIAH	BOBOT(SKS)		SEMESTER	Direvisi
	Analisis Jaringan Sosial (Social Network Analysis)	MKP02	Teknik	3	0	7/8	11/17/2022
Otoritas	Pengembang RPS			Ketua Kelompok Keahlian		Ka PRODI	
	(Nama Koordinator Dosen MK)			(Nama Ketua KK)		(Nama Kepala Program Studi)	
Deskripsi Mata Kuliah	<p>Pada mata kuliah ini mahasiswa mempelajari tentang: (1) definisi dan model fundamental Analisis Jaringan Sosial; (2) tipe jaringan, struktur, model, dan proses dinamis pada jaringan sosial; (3) metode perhitungan sentralitas jaringan sosial; (4) metode untuk mengidentifikasi komunitas dalam jaringan sosial; (5) perangkat lunak untuk menerapkan analisis jaringan sosial; (6) visualisasi jaringan sosial. Mata kuliah ini menggunakan studi kasus jaringan sosial Twitter.</p> <p><i>In this course students learn about: (1) the definition and fundamental models of Social Network Analysis; (2) network types, structures, models, and dynamic processes on social networks; (3) calculation methods of the social networks centrality; (4) methods for identifying communities in social networks; (5) software for implementing social network analysis; (6) visualization of social networks. This course uses Twitter social network case studies.</i></p>						
Capaian Pembelajaran Lulusan	Capaian Pembelajaran Lulusan (CPL) PRODI						
	CPL08	Kemampuan mengimplementasi kebutuhan <i>computing</i> dengan mempertimbangkan berbagai metode/algorithm yang sesuai.					
&Capaian Pembelajaran Mata Kuliah	Capaian Pembelajaran Mata Kuliah (CPMK)					CPL yang di dukung	
	CPMK081	Kemampuan menganalisis dan mendesain solusi untuk proyek terkait analisis jejaring sosial dengan benar.				CPL08	
	CPMK082	Kemampuan mengimplementasikan solusi untuk proyek terkait analisis jejaring sosial dengan sistematis				CPL08	
Penilaian	Id CPMK	Bobot per Bentuk Penilaian					TOTAL BOBOT PER CPMK
		Tugas 1	Tugas 2	Tugas 3	Proyek 1	Proyek 2	
	CPMK081	5	5	0	10	0	20
	CPMK082	5	5	0	20	0	30
	CPMK083	0	0	10	0	10	20
	CPMK084	0	0	10	0	20	30
	Total per penilaian	10	10	20	30	30	100
Pustaka	Utama:						
	[WAS94] Social Network Analysis: Methods and Applications, Stanley Wasserman and Katherine Faust, Cambridge University Press, 1994						
	[MAK11] Social Network Analysis for Start Up, Maksim Tsvetov and Alexander Kouznetsov, O'Reilly, 2011						
	Pustaka Pendukung:						
	-						
	Computer Science Papers						
Media Pembelajaran	Software:					Hardware :	
	Gephi; NetworkX, python					Komputer/Laptop; Projector	
Team Teaching	(Nama-Nama Dosen Pengampu MK)						
Matakuliah Syarat							
Ambang Batas Kelulusan Mahasiswa	50.01						
Ambang Batas Kelulusan MK	85.00%						

Gambar 2.1 Contoh Format RPS (APTIKOM, 2024)

MINGGU KE-	ID CPMK	DESKRIPSI SUB CPMK	INDIKATOR KETERCAPAIAN CPMK	BENTUK ASSESMEN	MATERI	METODE	LUAR JARINGAN (TATAP MUKA)	DALAM JARINGAN (DARING)
1	CPMK081 CPMK082	Mahasiswa mampu menjelaskan konsep dasar Analisis Jaringan Sosial (AJS) [C2, A1, P1]	Ketepatan mahasiswa dalam menjelaskan konsep dasar Analisis Jaringan Sosial (AJS)	Proyek 1	a) Pendahuluan: Pengenalan Analisis Jaringan Sosial (AJS) b) Konsep dasar AJS dan definisi-definisi dalam AJS c) Konsep umum pengumpulan dan analisis data jaringan sosial	Ceramah dan Diskusi [TM: 1x(3x50')] Tugas 1 Literature Review: Membaca paper SNA dan menyusun resume sesuai format yang telah ditentukan [BM:1x(3x60')] Pustaka: [WAS94] [MAK11]	--	1) Mengikuti perkuliahan sinkronus 2) Penjelasan aturan perkuliahan 3) Penjelasan materi ke-1 yang belum jelas ke dosen 5) Diskusi
2	CPMK081 CPMK082	a) Mahasiswa mampu menerapkan beberapa tipe dan struktur jaringan dalam Tugas dan Proyek [C3, A1, P1] b) Mahasiswa mampu mengumpulkan data jaringan sosial	a) Ketepatan mahasiswa dalam menerapkan beberapa tipe dan struktur jaringan dalam Tugas dan Proyek b) Ketepatan mahasiswa dalam mengumpulkan data jaringan sosial menggunakan tools perangkat lunak AJS	Tugas 1	a) Tipe jaringan b) Struktur jaringan c) Model jaringan d) Penjelasan Rencana Proyek e) Tutorial NetworkX	Blended learning (flipped classroom) [DR: 1x(3x50')] Tugas 1: crawling data dan analisis Penjelasan Rencana Proyek 1	--	1) Mengikuti perkuliahan sinkron/Mempelajari video materi pertemuan ke-2 secara online 2) Menanyakan materi yang belum jelas ke dosen 3) Memperhatikan penjelasan Project 1 dan menerima tugas penyusunan proposal Project 1

Gambar 2.2 Contoh Format Rencana Pembelajaran Per Minggu

Berdasarkan Gambar 2.2, rencana pembelajaran per-minggu mencakup ID dan deskripsi sub CPMK, indikator ketercapaian, bentuk asesmen, materi, metode pembelajaran, serta aktivitas luar jaringan dan dalam jaringan. Format detail RPS dan rencana pembelajaran per-minggu pada penelitian ini disusun mengikuti struktur sebagaimana yang ditampilkan pada Gambar 2.1 dan Gambar 2.2.

2.2.3 PHP (*Hypertext Preprocessor*)

PHP (*Hypertext Preprocessor*) adalah bahasa pemrograman *server-side* yang digunakan untuk membangun aplikasi web dinamis. Laravel sendiri dibangun menggunakan PHP, sehingga PHP menjadi komponen utama dalam pengembangan backend (Qomariah & Jumiyati, 2025). Alasan penggunaan bahasa pemrograman PHP adalah :

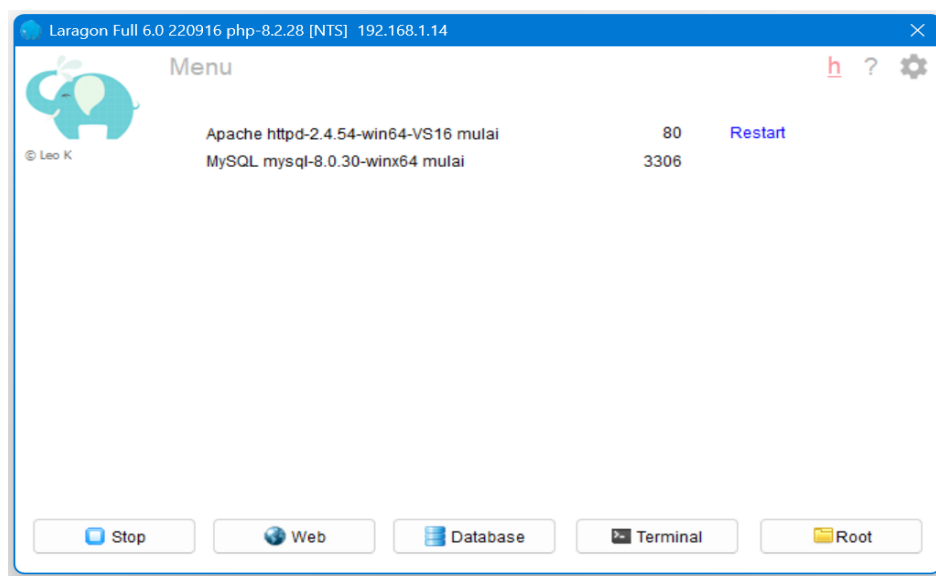
1. Kompatibel dengan Berbagai Database: PHP mendukung berbagai sistem basis data, termasuk MySQL.
2. Keamanan yang Kuat: PHP memiliki fitur keamanan seperti hashing password, proteksi XSS, dan proteksi SQL Injection.
3. Performa Tinggi dengan Laravel: Laravel membantu mengoptimalkan penggunaan PHP dalam pengembangan aplikasi.

4. Banyak Dukungan Komunitas: PHP memiliki komunitas yang besar sehingga memudahkan dalam mencari solusi jika terjadi kendala.

2.2.4 Laragon

Laragon merupakan perangkat lunak bersifat *open source* yang mendukung banyak sekali sistem operasi. Tugas laragon sebagai server virtual atau sering disebut sebagai *localhost* yang menggunakan domain sesuai dengan keinginan atau bisa disebut dengan *pretty url's*, menyediakan beragam layanan dan fitur seperti Apache, PHP Server, PhpMyAdmin, MySQL, Memcached, Redis, Composer, Xdebug, Cmdr, serta dukungan untuk Laravel. Laragon juga sebagai pengganti XAMPP sangat baik untuk pengelolaan aplikasi yang berbasis *website* (Sultan Adensa dkk., 2023). Kelebihan dari laragon yaitu:

1. *Portable*, project yang dapat dipindahkan dengan mudahnya tanpa dengan merusak sistem.
2. *Isolated*, sistem pada laragon terisolasi dengan sistem operasi maka apa yang pengguna lakukan pada aplikasi tidak mempengaruhi komputer lokal pengguna.
3. *Easy Operation*, aplikasi yang otomatis dengan memiliki banyak yang konfigurasi sehingga sangat mudah untuk digunakan.
4. Modern dan Powerful, aplikasi yang memiliki arsitektur *modern* sehingga dapat memudahkan digunakan saat membangun web yang modern.



Gambar 2.3 Antarmuka Laragon

2.2.5 Laravel

Laravel adalah kerangka kerja PHP *open-source* yang dirancang untuk mengoptimalkan dan mempercepat proses pengembangan aplikasi web sisi server. Dengan desain *Model-View-Controller* (MVC) yang dipakai agar membangun aplikasi website. Framework ini pertama sekali diciptakan oleh Taylor Otwell pada tanggal 22 Februari 2012. Selain itu, Laravel ialah pengembangan *website* berlandaskan MVP yang ditulis dalam PHP yang dirancang untuk meningkatkan kualitas *software* dengan mengurangi *cost* pengembangan awal, biaya perawatan, serta untuk mengoptimalkan pengalaman bekerja dengan aplikasi dengan menyediakan sintaks ekspresif, jelas dan efisien. Laravel menawarkan berbagai fitur kunci, seperti sistem *routing* yang fleksibel, manajemen otentikasi yang terintegrasi, dan kemampuan migrasi basis data yang memudahkan pengelolaan skema basis data. Salah satu fitur paling menonjol dari Laravel adalah *Eloquent*, ORM bawaan yang memungkinkan pengembang berinteraksi dengan basis data menggunakan sintaksis PHP yang bersih dan intuitif. Selain itu, *Blade*, mesin *templating* Laravel, mempermudah pengembangan tampilan dengan sintaksis yang mudah dipahami. *Framework* ini juga menawarkan alat-alat untuk pengelolaan antrian, sistem notifikasi, dan *caching*, semuanya dirancang untuk mempercepat dan mengoptimalkan kinerja aplikasi web. Dengan ekosistem paket tambahan yang kuat dan dukungan terus-menerus dari komunitas pengembang yang aktif, Laravel menjadi pilihan yang sangat baik bagi siapa pun yang ingin mengembangkan aplikasi web dengan cepat dan efisien (Alfarisi dkk., 2023).

```
namespace App\Http\Controllers;

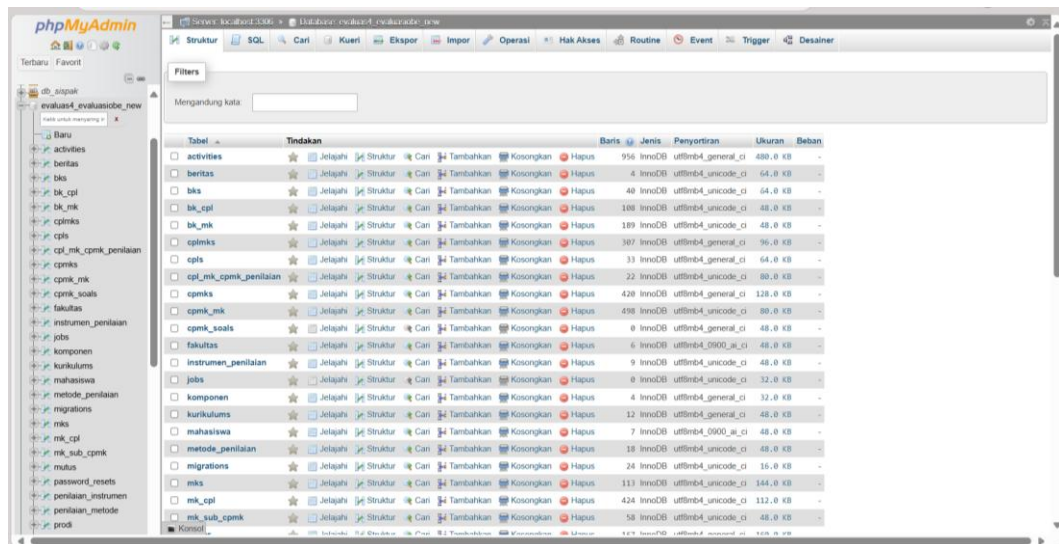
use Illuminate\Http\Request;
use App\Models\RPS;
use App\Models\Activity;
use App\Models\CPMK;
use App\Models\CPLMK;

class DosenController extends Controller
{
    public function dashboard(){
        return view('dosen.dashboard');
    }
}
```

Gambar 2.4 Contoh Kode Laravel

2.2.6 MySQL

MySQL merupakan *database engine* atau server *database* yang mendukung bahasa *database SQL* sebagai bahasa interaktif dalam mengelola data. MySQL adalah *database* yang populer di kalangan *programmer web* karena dianggap lebih stabil dan memiliki kekuatan yang lebih baik dalam menyimpan data dibandingkan dengan *database* lainnya. MySQL termasuk dalam kategori *software DBMS (Database Management System)* yang berfungsi sebagai server *database*. Sebagai DBMS, MySQL menggunakan lisensi *GNU General Public License (GPL)* yang bersifat *open source*, menawarkan fitur *multiplatform*, keandalan, kecepatan, kemudahan penggunaan, serta keamanan akses. Selain itu, MySQL mendukung perintah SQL (Putri dkk., 2023).



Gambar 2.5 MySQL pada PHPMyAdmin.

2.2.7 Unified Modeling Language (UML)


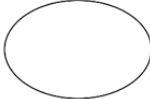


Unified Modeling Language (UML) adalah bahasa spesifikasi yang digunakan untuk memvisualisasikan dan membuat model abstrak dari suatu sistem atau produk akhir. UML menyediakan representasi visual atau blueprint dari langkah-langkah implementasi yang akan dilakukan dalam sebuah proyek, terutama yang berskala besar. Penggunaan diagram dalam UML merupakan bagian penting dari proses pemodelan dan perancangan, yang bertujuan untuk menyajikan gambaran umum sistem dan mendefinisikan semua fase yang terlibat dalam proses bisnis, seperti skema basis data dan fungsi sistem. Hal ini berperan penting untuk


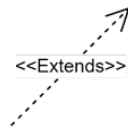
menyederhanakan kompleksitas selama proses implementasi (Bhatt & Nandu, 2021). Untuk memodelkan sistem, penelitian ini akan menguraikan penggunaan diagram UML yang relevan untuk menggambarkan fungsionalitas serta alur kerja sistem.

2.2.8 Use Case Diagram

Use case adalah gambaran yang menjelaskan bagaimana pengguna atau aktor berinteraksi dengan sistem untuk mencapai tujuan tertentu. Diagram ini menunjukkan fungsi-fungsi utama yang disediakan oleh sistem dari sudut pandang pengguna, serta hak akses dan peran masing-masing aktor. Dengan menggunakan *use case*, pengembang dapat memahami kebutuhan pengguna secara lebih jelas, menentukan fitur yang harus disediakan, dan memastikan sistem dapat memenuhi kebutuhan tersebut secara efisien dan efektif (Azmi Ginting dkk., 2024).

Tabel 2.2 Komponen *Use Case*



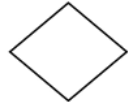

No	Simbol	Nama	Keterangan
1	 Actor	Aktor	Aktor merepresentasikan pengguna atau entitas yang berinteraksi dengan sebuah sistem. Aktor bisa berupa pengguna manusia, perangkat, atau sistem lain yang terlibat dalam use case.
2		<i>Use case</i>	<i>Use case</i> menggambarkan fungsionalitas atau layanan yang disediakan oleh sistem kepada aktor. Biasanya berisi tindakan-tindakan atau skenario yang dilakukan oleh aktor dengan sistem.
3		Asosiasi	Asosiasi menggambarkan hubungan antara aktor atau <i>use case</i> .
4		Generalisasi	Generalisasi menggambarkan hubungan hierarki antara aktor atau use case. Aktor atau use case yang lebih spesifik mewarisi atau memperluas fungsionalitas dari aktor atau use case yang lebih umum.

No	Simbol	Nama	Keterangan
5		<i>Include</i>	<i>Include</i> menunjukkan bahwa suatu <i>use case</i> selalu menyertakan fungsionalitas <i>use case</i> lain sebagai bagian dari pelaksanaannya.
6		<i>Extends</i>	<i>Extends</i> menunjukkan bahwa suatu <i>use case</i> dapat memperpanjang fungsionalitas <i>use case</i> lain di bawah kondisi tertentu.

2.2.9 Activity Diagram

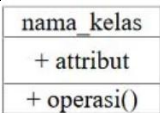



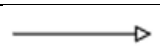
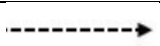

Activity diagram adalah sebuah diagram yang menunjukkan urutan tindakan atau aktivitas yang terjadi dalam sebuah sistem. Diagram ini memvisualisasikan langkah-langkah yang dilakukan, keputusan yang diambil, serta hubungan antar aktivitas tersebut secara berurutan, sehingga alur kerja dari sistem dengan lebih jelas dan mudah untuk dipahami (Hafsari dkk., 2024). Oleh karena itu *activity diagram* lebih menggambarkan proses-proses dan jalur-jalur aktivitas dari level atas secara umum (Hasanah & Sri Untari, 2020).

Tabel 2.3 Komponen *Activity Diagram*

No	Simbol	Nama	Keterangan
1		Status Awal	Status awal merupakan titik pertama dalam diagram aktivitas, menunjukkan di mana proses dimulai.
2		Aktivitas	Aktivitas menggambarkan tindakan atau langkah yang dilakukan dalam proses. Setiap aktivitas menunjukkan suatu pekerjaan atau operasi yang dilakukan dalam alur kerja.
3		Percabangan	Percabangan menunjukkan titik keputusan di mana alur dapat bercabang menjadi dua atau lebih jalur tergantung pada kondisi tertentu.
4		Status Akhir	Status akhir menunjukkan titik akhir dari suatu alur aktivitas, menandakan bahwa proses atau aliran kerja telah selesai.

2.2.10 Class Diagram

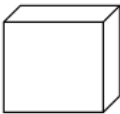


Class Diagram merupakan salah satu jenis *static diagram* yang berfungsi membantu dalam membangun kode eksekusi dari perangkat lunak. *Class Diagram* digunakan untuk memperlihatkan bagaimana berbagai elemen dalam sistem seperti *class*, *interface*, *data type*, maupun *component* dimodelkan. Diagram ini tidak hanya bermanfaat bagi pengembang perangkat lunak, tetapi juga dapat membantu pemangku kepentingan lain, seperti analis bisnis, untuk memahami struktur sistem yang sedang dibangun. Karena diagram kelas dapat dipetakan ke dalam struktur pemrograman, maka setiap elemen dan relasinya perlu diidentifikasi dengan baik sejak awal. Distribusi tanggung jawab dalam kelas harus jelas. Tiga elemen utama dalam sebuah diagram kelas adalah nama kelas, atribut, dan operasi. *Class* biasanya digambarkan dalam bentuk persegi panjang dengan nama *class* pada bagian atas. Bagian tengah persegi panjang berisi atribut, sedangkan bagian bawah berisi operasi atau fungsi yang dimiliki oleh kelas (Bhatt & Nandu, 2021).

No	Simbol	Nama	Keterangan
1		<i>Class</i>	<i>Class</i> yang ada di sistem.
2		<i>Interface</i>	<i>Interface</i> dalam pemrograman OOP
3		<i>Association</i>	Hubungan antarkelas yang bersifat umum
4		<i>Directed Association</i>	Hubungan antarkelas yang menunjukkan suatu kelas digunakan oleh kelas lain.
5		<i>Generalisasi</i>	Hubungan antarkelas dengan makna generalisasi-spesialisasi (umum khusus).
6		<i>Dependency</i>	Hubungan antarkelas dengan makna kebergantungan antarkelas.
7		<i>Aggregation</i>	Hubungan antarkelas dengan makna semua-bagian (<i>whole-part</i>).

2.2.11 Deployment Diagram

Deployment diagram adalah diagram UML yang menggambarkan konfigurasi fisik sistem, dengan menunjukkan bagaimana *artifact* atau komponen perangkat lunak ditempatkan pada node perangkat keras serta hubungan komunikasi di antara node


tersebut selama eksekusi sistem (Heriyanto & Rozi, 2023). Pada sistem berbasis web, *deployment* diagram umumnya merepresentasikan node seperti web server, *application server*, dan *database server* beserta artefak perangkat lunak yang berjalan di dalamnya, termasuk mekanisme komunikasi yang digunakan. Dengan demikian, tujuan utama diagram ini adalah untuk memberikan gambaran yang jelas mengenai topologi fisik sistem saat dijalankan (Niqotaini dkk., 2023).

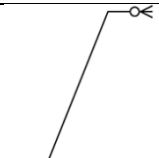
No	Simbol	Nama	Keterangan
1		<i>Node</i>	Elemen fisik atau perangkat keras tempat perangkat lunak dijalankan.
2		<i>Artifact</i>	Komponen perangkat lunak yang dihasilkan dari proses pengembangan.
3		<i>Communication Path</i>	Menunjukkan hubungan komunikasi antara dua node.

2.2.12 Entity Relationship Diagram (ERD)

Entity Relationship Diagram (ERD) adalah salah satu teknik pemodelan data yang sangat penting dalam proses perancangan basis data. ERD digunakan untuk merepresentasikan data dalam bentuk konseptual sebelum diimplementasikan ke dalam skema basis data yang sebenarnya. ERD dapat mengilustrasikan hubungan antar entitas, atribut yang dimiliki setiap entitas, dan jenis hubungan yang terjadi di antara entitas-entitas tersebut (Palinggi dkk., 2024).

Tabel 2.4 Komponen *Entity Relationship Diagram* (ERD)

No	Simbol	Nama	Keterangan
1		Entitas	Entitas merepresentasikan objek atau hal penting yang ingin dimodelkan dalam basis data. Setiap entitas biasanya mewakili tabel dalam basis data, yang berisi kumpulan data yang berhubungan.

No	Simbol	Nama	Keterangan
2		Relasi	Relasi menunjukkan hubungan antara dua entitas dalam model ERD. Relasi dapat menggambarkan jenis hubungan seperti <i>one-to-one</i> , <i>one-to-many</i> , atau <i>many to-many</i> .

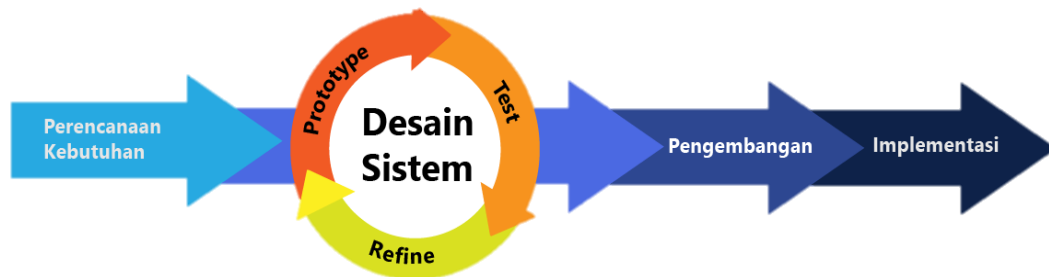
2.2.13 *Software Development Life Cycle (SDLC)*

Software Development Life Cycle (SDLC) adalah suatu pendekatan sistematis yang digunakan dalam pengembangan perangkat lunak. Pendekatan ini membantu tim pengembangan perangkat lunak dalam mengatur proses pembuatan, pengembangan, pengujian, dan pemeliharaan perangkat lunak secara efisien dan terstruktur. Tujuan dari SDLC adalah untuk menghasilkan perangkat lunak berkualitas tinggi yang memenuhi kebutuhan pengguna, mengikuti tenggat waktu, dan berada dalam batas anggaran yang ditentukan (Permana dkk., 2023). Dalam proses pengembangan perangkat lunak, *Software Development Life Cycle (SDLC)* menyediakan berbagai kerangka kerja dengan pendekatan yang bervariasi. Terdapat model klasik seperti *Waterfall* yang tahapannya berjalan secara linear dan berurutan, serta model yang lebih fleksibel seperti pendekatan iteratif pada *Prototyping*. Ada pula model *Rapid Application Development (RAD)*, yang dirancang untuk mempercepat pengembangan melalui siklus singkat dan pemanfaatan komponen (Gupta, 2021).

2.2.14 *Rapid Application Development (RAD)*

Rapid Application Development (RAD) mengacu pada strategi pengembangan perangkat lunak adaptif yang berpusat di sekitar pembuatan *prototype* dan umpan balik yang dipercepat, sementara kurang fokus pada desain yang teliti. Teknik RAD dicirikan oleh fokus pada pengembangan dan pembuatan *prototype*, berlawanan dengan perencanaan ekstensif, dalam keseluruhan strateginya. Pengembangan aplikasi yang cepat memungkinkan pengembang untuk mengimplementasikan beberapa iterasi dan perubahan secara efisien pada produk tanpa perlu memulai proses pengembangan lagi. Praktik ini membantu menjamin bahwa hasil akhir berpusat pada kualitas dan sesuai dengan spesifikasi pengguna akhir (Permana dkk.,

2023). Salah satu faktor utama yang membuat metode RAD (*Rapid Application Development*) mampu mempercepat proses pengembangan karena melibatkan pengguna secara aktif pada setiap tahapan, di mana pengguna berperan langsung dalam pengambilan keputusan sehingga hasil sistem lebih cepat selesai dan sesuai kebutuhan.



Gambar 2.6 Tahapan *Rapid Application Development*.

Tahapan RAD terdiri dari 4 tahap yang terstruktur dan saling bergantung disetiap tahap, yaitu:

1. *Requirements Planning*

Requirements Planning dalam *Rapid Application Development* (RAD) berfokus pada identifikasi tujuan sistem guna memahami secara jelas kebutuhan yang harus dipenuhi oleh perangkat lunak yang dikembangkan.

2. Desain Sistem

Tahap ini merupakan fase iteratif yang berfokus pada proses perancangan dan penyempurnaan sistem, yang terdiri dari tiga langkah utama:

- 1) *Prototype*

Pengembang merancang *prototype* awal berdasarkan hasil analisis kebutuhan pengguna. *Prototype* ini mencakup desain antarmuka pengguna (*user interface*), alur kerja sistem (*workflow*), serta fungsionalitas utama sebagai representasi awal sistem yang akan dikembangkan.

- 2) *Test*

Prototype yang telah dibuat kemudian diuji dan dievaluasi langsung oleh calon pengguna untuk menilai kesesuaian desain, kemudahan penggunaan, serta kecocokan alur kerja dengan kebutuhan yang diharapkan.

3) *Refine*

Umpan balik yang terkumpul dari sesi pengujian dianalisis oleh pengembang untuk melakukan penyempurnaan (*refinement*) sistem, baik melalui perbaikan desain antarmuka, penyesuaian alur kerja, maupun penambahan fitur yang diperlukan hingga *prototype* dinilai optimal.

3. Pengembangan

Setelah desain sistem disepakati melalui workshop, tahap pengembangan pada metode RAD dilakukan secara iteratif dan inkremental dengan menerjemahkan *prototype* menjadi modul-modul fungsional. Proses ini menekankan kecepatan implementasi sekaligus fleksibilitas terhadap perubahan, dengan tetap menjaga komunikasi aktif bersama pengguna agar setiap fitur dapat diuji, divalidasi, dan disempurnakan sebelum sistem diimplementasikan secara menyeluruh.

4. *Implementation* (Penerapan)

Tahap Implementasi merupakan fase akhir dalam siklus RAD, di mana komponen-komponen sistem yang telah dikembangkan, diuji, dan disetujui oleh pengguna digabungkan menjadi satu produk akhir yang siap digunakan. Pada tahap ini, dilakukan pengujian akhir secara menyeluruh untuk memastikan semua modul dapat berintegrasi dan berfungsi dengan baik dalam lingkungan operasional yang sesungguhnya. Setelah sistem berhasil diterapkan dan beroperasi di lingkungan produksi, siklus pengembangan dianggap selesai.

2.2.15 Nilai Bisnis

Nilai bisnis dalam konteks sistem informasi dapat didefinisikan sebagai manfaat terukur maupun tidak terukur yang diperoleh organisasi dari investasi, pengembangan, atau implementasi sistem informasi, yang berkontribusi terhadap peningkatan kinerja operasional, efisiensi biaya, keunggulan bersaing, dan pencapaian tujuan organisasi (ZareRavasan & Krčál, 2021). Secara umum, nilai bisnis ini dapat dikategorikan menjadi dua jenis utama yaitu *tangible* dan *intangible*.

a. Nilai Bisnis *Tangible*

Nilai bisnis *tangible* (berwujud) adalah manfaat yang dapat diidentifikasi, diukur secara kuantitatif, dan seringkali dapat dinyatakan dalam satuan

moneter atau unit terukur lainnya. Manfaat ini bersifat konkret dan dampaknya dapat diobservasi secara langsung pada kinerja operasional.

b. Nilai Bisnis *Intangible*

Nilai bisnis intangible (tidak berwujud) adalah manfaat yang bersifat kualitatif dan lebih sulit untuk diukur secara langsung dalam nilai kuantitatif atau moneter, namun tetap memberikan dampak strategis yang signifikan bagi organisasi. Manfaat ini seringkali berkaitan dengan peningkatan kualitas layanan, reputasi, kepuasan pemangku kepentingan, atau perbaikan dalam proses pengambilan keputusan.

2.2.16 Black-Box Testing

Metode *Black Box Testing* merupakan salah satu teknik pengujian perangkat lunak yang bertujuan untuk mengidentifikasi kesalahan pada sistem aplikasi. Pengujian ini berfokus pada fungsionalitas sistem, seperti memastikan bahwa setiap fungsi berjalan sesuai kebutuhan. Dalam pelaksanaannya, *Black Box Testing* dilakukan dengan memberikan berbagai data masukan (*input*), termasuk data acak, untuk memperoleh hasil yang *valid*. Hasil pengujian dinyatakan *valid* apabila sistem mampu menolak masukan yang salah sehingga tidak dapat disimpan ke dalam basis data, serta menerima masukan yang benar sehingga dapat tersimpan dengan baik dalam sistem informasi (Nur Ichsanudin dkk., 2022).

2.2.17 Apache Jmeter

Apache JMeter adalah sebuah *tool open-source* yang dikembangkan oleh Apache Software Foundation dan digunakan secara luas untuk melakukan pengujian kinerja (*performance testing*) pada aplikasi berbasis web. Apache JMeter bekerja dengan cara mensimulasikan sejumlah besar pengguna virtual yang mengirimkan permintaan (*request*), seperti HTTP request, ke server atau aplikasi yang diuji, kemudian mengukur dan menganalisis respons sistem terhadap beban tersebut. Melalui mekanisme ini, JMeter mampu menilai apakah suatu sistem informasi dapat memberikan waktu respons yang cepat, throughput yang stabil, serta tingkat kesalahan (*error rate*) yang rendah ketika diakses secara bersamaan oleh banyak pengguna (Indrianto, 2023).

III. METODOLOGI PENELITIAN

3.1 Waktu dan Tempat Penelitian

3.1.1 Waktu Penelitian

Penelitian ini dimulai pada bulan September 2025 hingga bulan Januari 2026.

Waktu penelitian akan dijelaskan pada Tabel 3.1

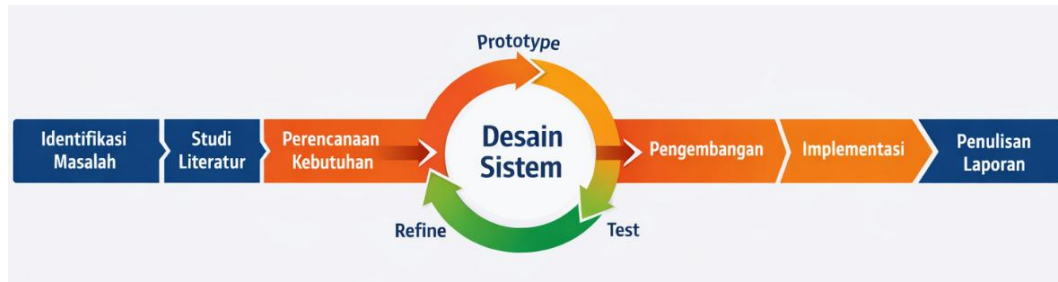
Tabel 3.1 Rencana Penelitian

Nama Kegiatan	2025																2026			
	Sep				Okt				Nov				Des				Jan			
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
Identifikasi Masalah																				
Studi Literatur																				
Perencanaan Kebutuhan																				
Desain Sistem																				
Pengembangan																				
Implementasi																				
Penulisan Laporan																				

3.1.2 Tempat Penelitian

Penelitian ini dilaksanakan di Jurusan Ilmu Komputer, Universitas Lampung, yang berlokasi di Jalan Prof. Dr. Ir. Sumantri Brojonegoro No. 1, Gedong Meneng, Kecamatan Rajabasa, Kota Bandar Lampung, Lampung 35141 dan kediaman Kemiling, Bandar Lampung.

3.2 Tahapan Penelitian



Gambar 3.1 Tahapan Penelitian

Dalam pelaksanaannya, penelitian ini mengacu pada kerangka kerja *Software Development Life Cycle* (SDLC) untuk memastikan proses pengembangan berjalan secara terstruktur. Model yang dipilih secara spesifik adalah *Rapid Application Development* (RAD), yang diintegrasikan dengan alur penelitian akademis. Terdapat tujuh tahapan kunci yang dilalui secara sistematis, mulai dari identifikasi masalah, studi literatur, perencanaan kebutuhan, desain sistem, pengembangan, implementasi, hingga penyusunan laporan akhir.

1. Identifikasi Masalah

Tahap ini diawali dengan analisis mendalam terhadap sistem penjaminan mutu yang sedang berjalan untuk mengidentifikasi kendala serta kebutuhan yang belum terpenuhi. Temuan dari analisis ini kemudian menjadi dasar untuk merumuskan spesifikasi kebutuhan fungsional sistem baru, dengan fokus utama pada pengembangan fitur Rencana Pembelajaran Semester (RPS).

2. Studi Literatur

Studi literatur dilakukan untuk menelusuri berbagai referensi yang berkaitan dengan sistem penjaminan mutu di perguruan tinggi, digitalisasi RPS, pengelolaan RPS hingga dapat diakses oleh mahasiswa, serta metode pengembangan sistem informasi dengan pendekatan RAD. Hal ini bertujuan untuk membangun landasan teori yang kuat sekaligus memahami perkembangan terbaru dalam bidang tersebut.

3. Perencanaan Kebutuhan (RAD)

Tahap ini termasuk dalam metode RAD, yang berfokus pada perencanaan kebutuhan fungsional maupun non-fungsional sistem. Perencanaan tersebut disusun berdasarkan hasil identifikasi permasalahan serta masukan dari

stakeholder, sehingga dapat menjadi acuan utama dalam proses perancangan dan pengembangan sistem.

4. Desain Sistem (RAD)

Pada tahap ini dilakukan perancangan awal sistem sesuai dengan kebutuhan yang telah ditetapkan. Proses perancangan dilakukan secara iteratif melalui metode *live prototyping*, dengan menekankan pada pengembangan antarmuka serta alur sistem yang dapat langsung diuji oleh pengguna.

5. Pengembangan (RAD)

Tahap ini berfokus pada pembangunan sistem secara teknis dengan menggunakan Laravel sebagai *framework* utama. Proses pengembangan dilakukan secara bertahap, dimulai dari implementasi fitur inti hingga penyempurnaan antarmuka dan fungsionalitas, sesuai dengan rancangan yang telah disusun sebelumnya.

6. Implementasi (RAD)

Pada tahap ini, sistem yang sudah dikembangkan diuji dan dijalankan pada lingkungan yang lebih nyata, misalnya melalui server lokal. Pengguna mulai memanfaatkan sistem sekaligus memberikan umpan balik yang akan digunakan untuk penyempurnaan lebih lanjut.

7. Penulisan Laporan

Tahap akhir berupa penyusunan laporan yang mendokumentasikan seluruh rangkaian penelitian, meliputi identifikasi masalah, studi literatur, metodologi, hasil pengembangan sistem, analisis evaluasi, hingga kesimpulan dan rekomendasi untuk pengembangan di masa depan.

3.3 Identifikasi Masalah

Pada Sistem Penjaminan Mutu yang sedang dikembangkan, belum terdapat modul terintegrasi untuk pengelolaan Rencana Pembelajaran Semester (RPS). Saat ini, proses penyusunan dan pengelolaan RPS masih dilakukan secara manual menggunakan *Microsoft Word* dan didistribusikan secara terpisah. Kondisi ini menimbulkan berbagai kendala yang menghambat efektivitas penjaminan mutu. Beberapa masalah utama yang teridentifikasi antara lain:

1. Potensi Keberagaman Format RPS.

Proses penyusunan yang bersifat manual membuka peluang adanya variasi dalam format dan struktur RPS yang digunakan. Hal ini dapat menjadi tantangan dalam upaya memastikan keselarasan penuh dengan panduan standar seperti dari APTIKOM, serta dalam melakukan evaluasi dan rekapitulasi yang konsisten di tingkat program studi.

2. Tidak Adanya Alur Validasi dan Persetujuan yang Terdokumentasi.

Proses peninjauan dan persetujuan oleh Kepala Program Studi (Kaprodi) berjalan lambat dan tidak terpusat. Dokumen harus diserahkan secara terpisah, sehingga sulit untuk melacak status persetujuan dan tidak ada catatan riwayat revisi yang jelas.

3. Sulitnya Pengelolaan Arsip dan Versi Dokumen.

Setiap dosen menyimpan file RPS secara mandiri. Hal ini menyulitkan pengelolaan arsip terpusat dan sering kali menimbulkan kebingungan dalam menentukan versi dokumen yang paling mutakhir, terutama saat ada pembaruan kurikulum atau pergantian semester.

4. Keterbatasan Aksesibilitas RPS bagi Pihak Terkait.

Belum tersedianya sistem terpusat untuk publikasi RPS membuat akses informasi bagi mahasiswa menjadi terbatas. Akibatnya, dokumen penting ini berpotensi sulit ditemukan kembali oleh mahasiswa seiring berjalannya semester, sehingga prinsip transparansi akademik belum berjalan secara optimal.

Secara keseluruhan, ketiadaan modul RPS yang terintegrasi dalam sistem penjaminan mutu menyebabkan proses pengelolaan dokumen menjadi tidak efisien, tidak konsisten, dan kurang transparan. Kondisi ini tidak hanya membebani dosen secara administratif, tetapi juga menjadi kendala signifikan dalam upaya memastikan kualitas, standarisasi, dan keberlanjutan RPS sebagai instrumen vital dalam penjaminan mutu pembelajaran.

3.4 Studi Literatur

Rencana Pembelajaran Semester (RPS) merupakan dokumen penting dalam proses perencanaan dan pelaksanaan kegiatan pembelajaran di perguruan tinggi. RPS disusun oleh dosen sebagai pedoman untuk menjamin bahwa proses pembelajaran sesuai dengan capaian pembelajaran lulusan (CPL) dan Capaian Pembelajaran Mata Kuliah (CPMK) yang telah ditetapkan. Oleh karena itu, pengelolaan RPS secara

digital menjadi bagian penting dalam sistem penjamin mutu akademik, agar proses penyusunan, evaluasi, dan pembaruan RPS dapat dilakukan secara efisien dan terdokumentasi dengan baik.

Beberapa penelitian sebelumnya telah mengembangkan sistem berbasis web untuk mendukung penyusunan dan pengelolaan RPS secara digital. Sistem-sistem tersebut umumnya memfasilitasi penyimpanan dokumen dan pengisian format standar RPS, namun masih terbatas pada fungsi dasar. Sebagian besar belum mengakomodasi keterkaitan antara CPL, CPMK, dan indikator penilaian, serta belum menyediakan mekanisme validasi oleh pihak terkait sebelum RPS digunakan dalam proses pembelajaran.

Dalam implementasinya, sistem pengelolaan RPS tidak hanya dituntut untuk menyediakan format yang sesuai standar, tetapi juga harus mampu memfasilitasi proses validasi oleh pihak terkait, seperti Kaprodi, sebelum RPS diakses mahasiswa. Hal ini penting untuk menjamin konsistensi, kesesuaian, dan kualitas isi RPS dengan ketentuan akademik yang berlaku.

Selain itu, fleksibilitas bagi dosen dalam memperbarui RPS juga menjadi kebutuhan, mengingat adanya kemungkinan perubahan isi dokumen sesuai perkembangan mata kuliah pada setiap semester. Sistem juga perlu memberikan akses RPS kepada mahasiswa agar dapat memahami alur pembelajaran, materi, serta mekanisme penilaian sejak awal perkuliahan. Dengan demikian, sistem pengelolaan RPS tidak hanya berfungsi sebagai repositori dokumen, tetapi juga sebagai instrumen penunjang dalam peningkatan kualitas dan penjaminan mutu pembelajaran di perguruan tinggi.

3.5 Perencanaan Kebutuhan (*Rapid Application Development*)

Penelitian ini menerapkan pendekatan *Rapid Application Development* (RAD) dalam tahap perencanaan kebutuhan, dengan fokus pada percepatan proses dan iterasi berulang melalui pembuatan prototipe langsung. Penulis membangun prototipe berdasarkan masukan pengguna dan melakukan penyempurnaan secara bertahap sesuai hasil evaluasi.

3.5.1 Perangkat Penelitian

Untuk mendukung seluruh rangkaian tahapan penelitian sesuai prosedur, digunakan sejumlah alat bantu yang terbagi ke dalam dua jenis, yakni perangkat keras dan perangkat lunak. Berikut ini merupakan daftar alat yang digunakan selama proses penelitian berlangsung:

3.5.1.1 Perangkat Keras

Untuk menunjang proses pengembangan dan pengujian sistem, digunakan perangkat laptop dengan spesifikasi yang sesuai kebutuhan penelitian. Adapun komponen perangkat keras yang digunakan adalah sebagai berikut:

1. *System Manufacturer* : ASUS
2. *System Model* : Asus Vivobook A1402ZA
3. *Processor* : intel® Core™ i5-1235U CPU @ 1.30GHz
4. RAM : 8GB DDR4
5. Penyimpanan : SSD 256GB

3.5.1.2 Perangkat Lunak

Selama pelaksanaan penelitian, sejumlah perangkat lunak dimanfaatkan untuk memfasilitasi pengembangan sistem secara menyeluruh. Setiap perangkat lunak memiliki peran tertentu dalam mendukung tahap-tahap pengembangan, mulai dari penulisan kode hingga pengujian dan manajemen versi. Berikut adalah daftar perangkat lunak yang digunakan:

1. Sistem Operasi: Windows 11 Pro 64-bit
2. Code Editor: Visual Studio Code 1.99.3
3. Browser: Google Chrome 136.0.7103.49
4. Local Server: Laragon 6.0
5. Framework Backend: Laravel 8
6. Framework Frontend: Bootstrap
7. Version Control: GitHub

3.5.2 Ringkasan Eksekutif

Penelitian ini berfokus pada pengembangan fitur pengelolaan Rencana Pembelajaran Semester (RPS) dalam Sistem Penjamin Mutu Jurusan Ilmu

Komputer Universitas Lampung. Sistem dikembangkan untuk memungkinkan dosen memperbarui atau membuat RPS sesuai kebutuhan setiap semester, serta menyediakan mekanisme validasi oleh Kaprodi atau Penjamin Mutu Prodi sebelum RPS dapat diakses tamu/mahasiswa. Dengan demikian, pengembangan ini mendukung proses penjaminan mutu akademik yang lebih terkontrol, transparan, dan berkelanjutan, sekaligus memastikan mahasiswa memiliki akses terhadap RPS terbaru sebagai pedoman pembelajaran.

Metodologi pengembangan yang digunakan adalah *Rapid Application Development* (RAD). Proses pengembangan dimulai dari tahap perencanaan kebutuhan, dilanjutkan dengan desain sistem (iteratif), pengembangan, hingga implementasi. Setiap siklus pengembangan akan melibatkan umpan balik dari *stakeholder* untuk memastikan sistem memenuhi kebutuhan yang ada.

Hasil akhir dari proyek ini diharapkan dapat memberikan kontribusi signifikan terhadap efisiensi dan efektivitas pengelolaan dokumen RPS di Jurusan Ilmu Komputer, serta membantu meningkatkan kualitas pelaporan dan evaluasi capaian pembelajaran secara menyeluruh.

3.5.3 Kebutuhan Bisnis

Sistem yang dibutuhkan harus mampu mengelola Rencana Pembelajaran Semester (RPS) secara fleksibel dan terstruktur di Jurusan Ilmu Komputer Universitas Lampung. Fitur utama yang diperlukan mencakup kemampuan bagi dosen untuk membuat, memperbarui, dan mengelola versi RPS sesuai kebutuhan setiap semester. Sistem juga harus menyediakan mekanisme validasi oleh Kaprodi/Penjamin Mutu sehingga RPS yang ditampilkan kepada mahasiswa telah melalui proses peninjauan dan sesuai ketentuan akademik.

Selain itu, sistem perlu memberikan akses RPS kepada mahasiswa agar mereka dapat memahami alur pembelajaran, materi, dan mekanisme penilaian sejak awal perkuliahan. Dengan adanya pengelolaan RPS yang terdokumentasi dan terkontrol, proses penjaminan mutu akademik dapat dilakukan secara lebih efektif, transparan, dan berkelanjutan.

3.5.4 Kebutuhan Fungsional dan Non-Fungsional

Dalam merancang dan mengembangkan sistem pengelolaan Rencana Pembelajaran Semester (RPS), diperlukan identifikasi kebutuhan sistem secara menyeluruh, baik dari sisi fungsional maupun non-fungsional. Kebutuhan fungsional mencakup fungsionalitas utama yang harus disediakan oleh sistem agar dapat digunakan sesuai dengan tujuan pengelolaan RPS. Sementara itu, kebutuhan non-fungsional berkaitan dengan kualitas sistem seperti performa, keamanan, dan kemudahan penggunaan. Berikut adalah rincian kebutuhan fungsional dan non-fungsional yang disajikan pada Tabel 3.2 dan Tabel 3.3.

3.5.4.1 Kebutuhan Fungsional

Tabel 3.2 Tabel Kebutuhan Fungsional

No	ID	Kebutuhan Fungsional	Use Case
1	FR-001	Sistem harus dapat menampilkan daftar RPS publik dan menyediakan fitur pencarian untuk menyaring daftar tersebut.	UC-001
2	FR-002	Sistem harus menyediakan <i>button</i> unduh RPS yang dipilih tamu dari daftar RPS.	UC-002
3	FR-003	Sistem harus memungkinkan dosen untuk melakukan pengelolaan data RPS serta menyediakan fitur untuk mencetak dan mengajukan validasi RPS.	UC-003
4	FR-004	Sistem harus menyediakan halaman detail RPS dan fitur pengelolaan data aktivitas mingguan.	UC-004
5	FR-005	Sistem harus menyediakan fitur mengelola pemetaan CPL ke mata kuliah,.	UC-005
6	FR-006	Sistem harus menyediakan fitur pengelolaan data CPMK.	UC-006
7	FR-007	Sistem harus menyediakan fitur pengelolaan komponen penilaian.	UC-007
8	FR-008	Sistem harus menyediakan fitur validasi rps sebelum dapat diakses tamu.	UC-008
9	FR-009	Sistem harus menyediakan fitur yang menampilkan daftar rps yang sudah tervalidasi.	UC-009

3.5.4.2 Kebutuhan Non-Fungsional

Tabel 3.3 Tabel Kebutuhan Non Fungsional

No	ID	Kebutuhan Non-Fungsional
1	NFR-001	Sistem harus menampilkan umpan balik yang jelas dan informatif, seperti notifikasi keberhasilan atau pesan kesalahan yang mudah dipahami, untuk setiap tindakan pengguna. Hal ini bertujuan memandu pengguna dalam berinteraksi dengan sistem.
2	NFR-002	Sistem harus menjamin bahwa hanya dokumen RPS yang telah melewati tahap validasi dan berstatus " <i>Published</i> " yang dapat diakses oleh pengguna publik.
3	NFR-003	Sistem harus menerapkan mekanisme kontrol versi otomatis, di mana setiap RPS baru yang dibuat untuk mata kuliah yang sama akan ditandai sebagai versi terbaru, sementara versi sebelumnya tetap tersimpan.
4	NFR-004	Sistem harus mampu menjaga stabilitas dan kinerja di bawah beban pengguna simultan (<i>load testing</i>), serta memberikan waktu respons (<i>response time</i>) yang cepat dan wajar untuk setiap interaksi pengguna.

3.5.5 Nilai Bisnis

Dalam pengembangan fitur ini, nilai bisnis yang dapat diperoleh terbagi menjadi dua, yaitu nilai *tangible* dan *intangible*. Berikut adalah penjelasannya:

3.5.5.1 *Tangible*

1. Efisiensi proses dokumentasi RPS

Fitur pengelolaan RPS memungkinkan proses penyimpanan, pembaruan, dan pengarsipan dokumen dilakukan secara terpusat dan terstruktur, sehingga mengurangi aktivitas manual serta mempercepat distribusi dokumen kepada dosen dan pihak terkait.

2. Peningkatan efisiensi administratif

Sistem memungkinkan pengelolaan data RPS secara digital sehingga dapat mengurangi penggunaan kertas, meminimalkan kesalahan input,

menghindari kemungkinan perbedaan format RPS, serta mempermudah proses pencarian dokumen.

3. Optimalisasi Alokasi Sumber Daya Mengajar.

Dengan basis data RPS yang terstruktur, pimpinan jurusan dapat dengan mudah menganalisis keselarasan materi ajar, distribusi beban SKS, dan kebutuhan sumber daya pendukung pembelajaran, sehingga mendukung pengambilan keputusan yang lebih efisien dalam perencanaan akademik dan anggaran.

3.5.5.2 Intangible

1. Meningkatkan transparansi dan akuntabilitas

Pengelolaan RPS yang terdigitalisasi dan terstandarisasi memungkinkan pihak jurusan dan pimpinan fakultas melakukan pemantauan terhadap kesesuaian isi RPS dengan kurikulum. Hal ini mendukung transparansi dalam pelaksanaan pembelajaran dan tanggung jawab dosen dalam menyusun RPS.

2. Mendorong konsistensi kualitas pembelajaran

Dengan tersedianya fitur pengelolaan RPS yang sistematis, dosen memiliki referensi yang jelas dan terdokumentasi. Ini membantu menjaga keseragaman standar pembelajaran antar kelas dan semester yang berbeda.

3. Meningkatkan reputasi jurusan dalam hal tata kelola akademik

Adanya sistem RPS yang terkelola dengan baik menunjukkan komitmen jurusan terhadap peningkatan kualitas tata kelola pembelajaran. Hal ini dapat memperkuat citra jurusan di mata mahasiswa, dosen, dan pihak eksternal seperti asesor akreditasi atau mitra industri.

3.5.6 Definisi dan Strategi Analisis Kebutuhan

Untuk mengumpulkan informasi terkait kebutuhan sistem dalam penelitian ini, dilakukan diskusi langsung dengan calon pengguna. Hasil diskusi tersebut kemudian diolah dan disajikan dalam bentuk Matriks Kebutuhan Fungsional dan Matriks Kebutuhan Non-Fungsional yang dapat dilihat pada Tabel 3.4 dan

Tabel 3.5. Selain itu, analisis juga meliputi pemetaan proses saat ini (*As-Is*) dan proses yang diharapkan (*To-Be*) untuk menggambarkan perubahan yang diinginkan dalam pengelolaan Rencana Pembelajaran Semester (RPS).

Tabel 3.4 Matriks Kebutuhan Fungsional *As-Is* dan *To-Be*

Kebutuhan Fungsional		
No	Proses Berjalan (<i>As-Is</i>)	Sistem yang akan dibuat (<i>To-Be</i>)
1	Pembuatan dan pembaruan RPS dilakukan secara manual menggunakan <i>software</i> terpisah seperti <i>Microsoft Word</i> dan disimpan masing-masing dosen.	Sistem memungkinkan untuk membuat, mengubah, menyimpan, dan mengelola versi RPS serta aktivitas minggunya secara terpusat langsung di dalam sistem.
2	Proses validasi RPS tidak terstruktur. Dosen menyerahkan <i>file</i> secara manual kepada Kaprodi tanpa adanya catatan revisi yang terdokumentasi.	Sistem memungkinkan Kaprodi dapat menerima pengajuan, meninjau, serta memberikan persetujuan atau penolakan validasi RPS dengan catatan revisi langsung melalui sistem.
3	Tidak ada repositori terpusat untuk memantau RPS yang sudah disetujui. Pengarsipan dilakukan secara manual dan terpisah.	Sistem memungkinkan Kaprodi dan Penjamin Mutu dapat melihat rekapitulasi seluruh RPS yang telah berhasil divalidasi dalam satu halaman khusus.
4	Proses penyebaran RPS kepada pihak berkepentingan belum memiliki sistem terpusat, sehingga distribusinya bergantung pada masing-masing dosen.	Pihak berkepentingan dapat menelusuri atau mencari RPS berdasarkan mata kuliah tanpa perlu login, setelah RPS divalidasi Kaprodi.

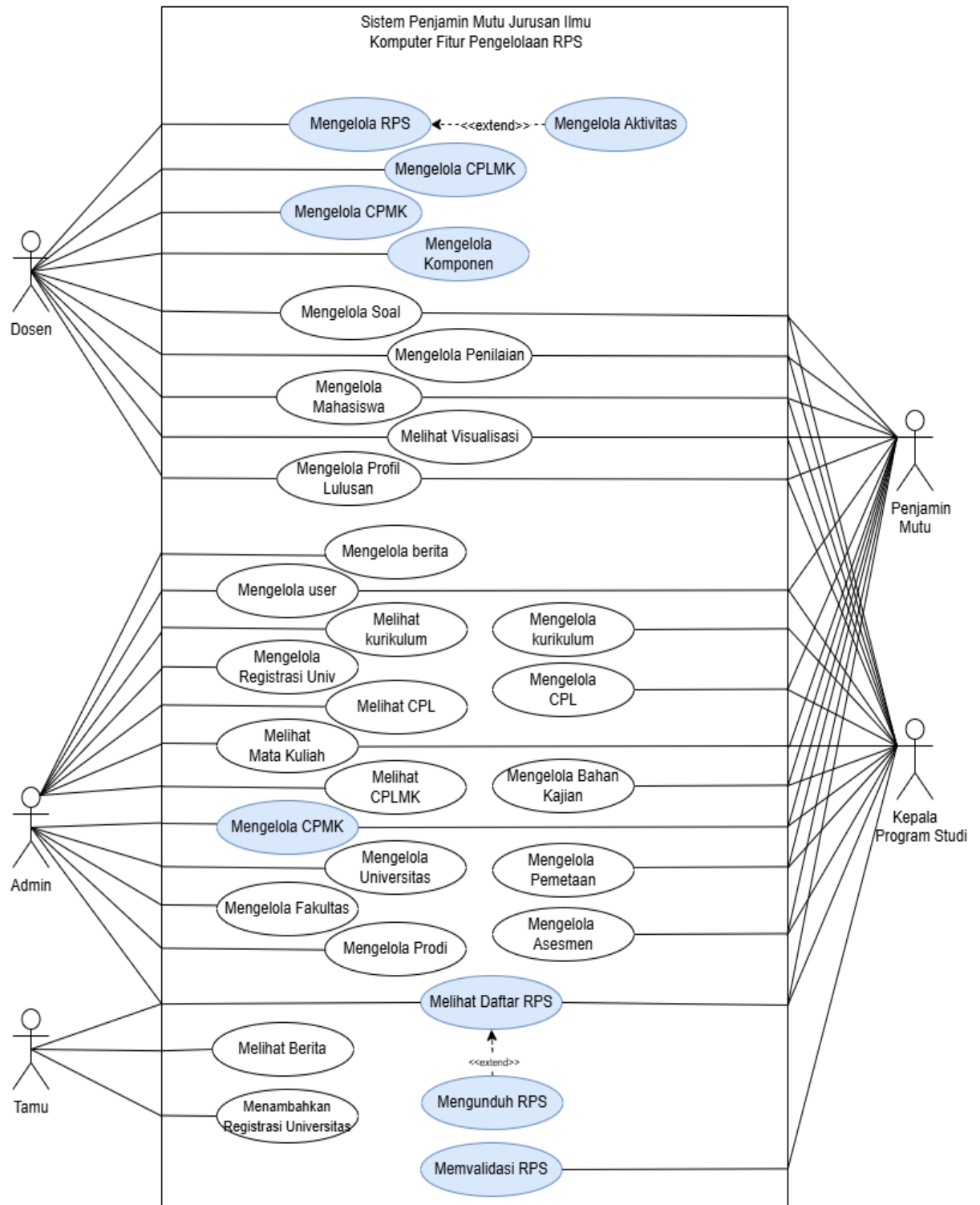
Tabel 3.5 Matriks Kebutuhan Non-Fungsional *As-Is* dan *To-Be*

Kebutuhan Non-Fungsional		
No	Proses Berjalan (<i>As-Is</i>)	Sistem yang akan dibuat (<i>To-Be</i>)
1	<i>Operational Requirements</i>	
	- Proses sepenuhnya bergantung pada perangkat lunak yang terpasang di komputer masing-masing.	- Sistem harus berbasis web dan dapat diakses melalui peramban modern tanpa perlu instalasi <i>software</i> tambahan di sisi pengguna.
2	<i>Performance Requirements</i>	
	- Waktu untuk menemukan RPS tertentu bervariasi dan bisa memakan waktu beberapa menit karena harus mencari file secara manual pada <i>device</i> masing-masing dosen.	- Waktu <i>respons</i> sistem untuk memuat halaman utama dan menampilkan daftar RPS harus relatif cepat pada koneksi internet standar.
3	<i>Safety Requirements</i>	
	- Pengelolaan dokumen rentan terhadap kesalahan pengguna. File dapat dihapus atau tertimpa tanpa peringatan maupun riwayat versi sebelumnya.	- Sistem menerapkan konfirmasi penghapusan dan penyimpanan versi lama (<i>versioning</i>) agar dokumen tidak terhapus atau tertimpa tanpa sengaja.
4	<i>Security Requirements</i>	
	- Keamanan dokumen hanya sebatas keamanan perangkat lokal. File dapat dengan mudah disalin dan didistribusikan tanpa kontrol.	- Hak akses ke RPS akan dibedakan berdasarkan perannya. misalnya, Dosen hanya bisa mengelola RPS miliknya, sementara Kaprodi bisa melakukan validasi.

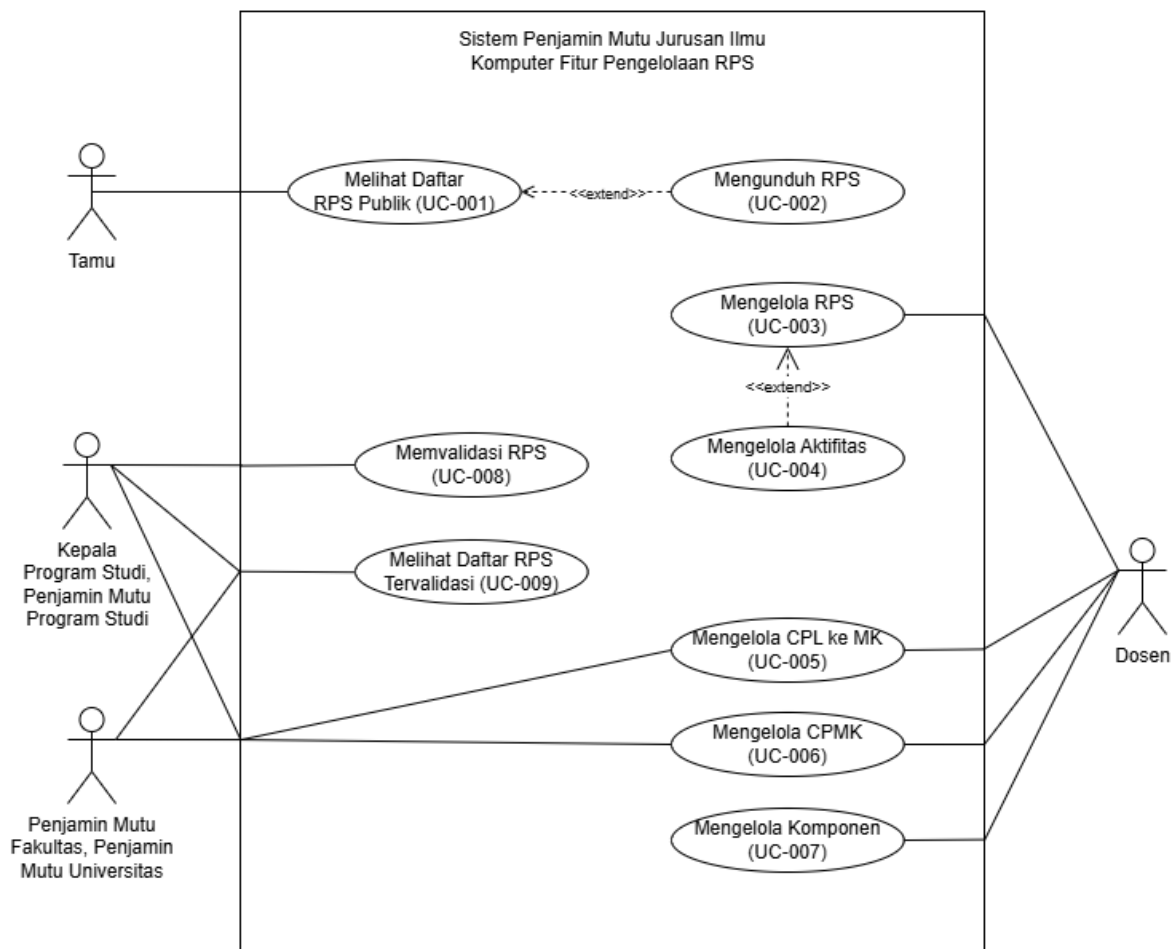
3.5.7 Use Case Diagram

Use Case Diagram pada penelitian ini menggambarkan interaksi antara pengguna dengan sistem dalam pengelolaan Rencana Pembelajaran Semester (RPS) sebagai bagian dari sistem penjamin mutu. Diagram ini memvisualisasikan fitur-fitur utama yang dikembangkan serta aktor-aktor yang terlibat. Diagram disusun menggunakan

Draw.io dan difokuskan pada modul pengelolaan RPS, dengan penandaan khusus untuk membedakan fitur yang dikembangkan dalam penelitian ini dari sistem yang telah ada sebelumnya.



Gambar 3.2 Use Case Diagram Global



Gambar 3.3 Use Case Diagram

Use case pada Gambar 3.3 merupakan fitur yang dikembangkan pada penelitian ini. Untuk deskripsi dari use case akan dijelaskan beberapa tabel berikut ini:

3.5.7.1 Melihat Daftar RPS Publik (UC-001)

Tabel 3.6 Deskripsi Use Case Melihat Daftar RPS Publik

Nama Use Case : Melihat Daftar RPS Publik	ID : UC-001
Aktor : Tamu	
Deskripsi : Use case ini menggambarkan proses pengguna (tamu) dalam melihat, mencari, dan mengunduh rps publik yang tersedia.	
Pemicu (Trigger) : Pengguna berada di <i>landing page</i> sistem, dan mengklik menu “RPS” di <i>navbar</i> .	
Kondisi Awal (Preconditions) : Sistem memiliki setidaknya satu RPS yang dapat ditampilkan.	

Alur Normal (Melihat Daftar RPS) :

1. Pengguna mengakses sistem melalui browser.
2. Sistem menampilkan halaman yang sedang diakses oleh pengguna.
3. Pengguna memilih menu “RPS” pada navbar.
4. Sistem menampilkan halaman Daftar RPS Publik yang berisi seluruh data RPS yang tersedia.

Alur Normal (Mencari RPS) :

1. Pengguna mengakses sistem melalui browser.
2. Sistem menampilkan halaman yang sedang diakses oleh pengguna.
3. Pengguna memilih menu “RPS” pada navbar.
4. Sistem menampilkan halaman Daftar RPS Publik yang berisi seluruh data RPS yang tersedia.
5. Pengguna memasukkan kata kunci (kode atau nama matkul) pada *search bar*.
6. Sistem menyaring dan menampilkan daftar RPS yang sesuai dengan kata kunci.

Kondisi Akhir (Postconditions) :

1. Sistem berhasil menampilkan Daftar RPS (lengkap atau hasil pencarian) kepada pengguna.

3.5.7.2 Mengunduh RPS (UC-002)Tabel 3.7 Deskripsi *Use Case* Mengunduh RPS

Nama Use Case : Mengunduh RPS	ID : UC-002
Aktor : Tamu	
Deskripsi : <i>Use case</i> ini menggambarkan proses pengguna (tamu) mengunduh file RPS dari daftar yang ditampilkan.	

Pemicu (<i>Trigger</i>) : Pengguna menekan tombol “Unduh RPS” pada salah satu RPS yang ada di daftar.
Kondisi Awal (<i>Preconditions</i>) : Halaman Daftar RPS Publik sedang ditampilkan kepada pengguna.
Alur Normal (<i>Normal Course</i>) : <ol style="list-style-type: none"> 1. Pengguna mengakses sistem melalui browser. 2. Sistem menampilkan halaman yang sedang diakses oleh pengguna. 3. Pengguna memilih menu “RPS” pada navbar. 4. Sistem menampilkan halaman Daftar RPS Publik yang berisi seluruh data RPS yang tersedia. 5. Pengguna menekan tombol “Unduh RPS” pada salah satu RPS yang diinginkan. 6. Sistem memulai proses pengunduhan file RPS ke perangkat pengguna.
Kondisi Akhir (<i>Postconditions</i>) : <ol style="list-style-type: none"> 1. File dokumen RPS berhasil diunduh ke perangkat pengguna.

3.5.7.3 Mengelola RPS (UC-003)

Tabel 3.8 Deskripsi *Use Case* Mengelola RPS

Nama <i>Use Case</i> : Mengelola RPS	ID : UC-003
Aktor : Dosen	
Deskripsi : <i>Use case</i> ini menggambarkan proses mengelola RPS yang dilakukan oleh pengguna (dosen), meliputi proses menambah, melihat detail, mengubah, mencetak, mengajukan validasi dan menghapus data RPS.	
Pemicu (<i>Trigger</i>) : Pengguna memilih menu “RPS” di <i>sidebar</i> .	
Kondisi Awal (<i>Preconditions</i>) : Pengguna telah masuk ke dalam sistem dengan otoritas dosen.	

Alur Normal (Melihat Daftar RPS) :

1. Pengguna mengakses *sidebar* “RPS”.
2. Sistem menampilkan halaman yang berisi daftar RPS yang telah dibuat oleh pengguna (dosen).

Alur Alternatif 1 (Menambahkan RPS) :

1. Pengguna mengakses *sidebar* “RPS”.
2. Sistem menampilkan halaman yang berisi daftar RPS yang telah dibuat oleh pengguna (dosen).
3. Pengguna mengklik tombol “Tambah RPS”.
4. Sistem menampilkan *form* tambah RPS
5. Pengguna mengklik “Simpan RPS”.
6. Sistem menyimpan data RPS baru dan menampilkan notifikasi bahwa RPS berhasil ditambahkan.

Alur Alternatif 2 (Mengubah RPS) :

1. Pengguna mengakses *sidebar* “RPS”.
2. Sistem menampilkan halaman yang berisi daftar RPS yang telah dibuat oleh pengguna (dosen).
3. Pada RPS tertentu yang ingin diubah, pengguna mengklik tombol berikon pensil.
4. Sistem menampilkan *form* edit RPS dengan data yang sudah ada.
5. Pengguna mengubah data RPS dan mengklik “Simpan Perubahan”.
6. Sistem mengubah RPS dan menampilkan notifikasi bahwa RPS berhasil diperbarui.

Alur Alternatif 3 (Menghapus RPS) :

1. Pengguna mengakses *sidebar* “RPS”.
2. Sistem menampilkan halaman yang berisi daftar RPS yang telah dibuat oleh pengguna (dosen).
3. Pada RPS tertentu yang ingin dihapus, pengguna mengklik tombol berikon *trash*.
4. Sistem menampilkan konfirmasi penghapusan.

5. Pengguna mengonfirmasi penghapusan.
6. Sistem menghapus RPS dan menampilkan notifikasi bahwa RPS telah dihapus.

Alur Alternatif 4 (Mencetak RPS)

1. Pengguna mengakses sidebar “RPS”.
2. Sistem menampilkan halaman yang berisi daftar RPS yang telah dibuat oleh pengguna (dosen).
3. Pada RPS tertentu yang ingin dilihat dan dicetak, pengguna mengklik tombol berikon *printer*.
4. Sistem menampilkan preview dokumen RPS sekaligus dialog pencetakan.
5. Pengguna mengklik cetak.

Alur Alternatif 5 (Mengajukan Validasi RPS)

1. Pengguna mengakses sidebar “RPS”.
2. Sistem menampilkan halaman yang berisi daftar RPS yang telah dibuat oleh pengguna (dosen).
3. Pada RPS tertentu yang ingin diajukan untuk di validasi, pengguna mengklik tombol biru gelap berikon *up*.
4. Sistem menampilkan konfirmasi pengajuan.
5. Pengguna mengonfirmasi pengajuan.
6. Sistem mengubah status RPS menjadi *Pending*, dan menonaktifkan *button* lihat detail, edit, dan hapus.
7. Sistem meneruskan RPS ke Kaprodi untuk di validasi.

Kondisi Akhir (Postconditions) :

1. Jika dosen menambah RPS, RPS baru tersimpan dan muncul di daftar.
2. Jika dosen mengubah RPS, perubahan langsung tersimpan di sistem.
3. Jika dosen menghapus RPS, data RPS tersebut hilang dari sistem.
4. Jika dosen mencetak RPS, file PDF dapat diunduh.

5. Jika dosen mengajukan RPS, status berubah menjadi *pending*.
6. Sistem selalu menampilkan notifikasi sukses setiap kali aksi berhasil dilakukan.

3.5.7.4 Mengelola Aktivitas (UC-004)

Tabel 3.9 Deskripsi *Use Case* Mengelola Aktivitas

Nama <i>Use Case</i> : Mengelola Aktivitas	ID : UC-004
Aktor : Dosen	
Deskripsi : <i>Use case</i> ini menggambarkan proses pengelolaan aktivitas mingguan pada sebuah RPS yang dilakukan oleh pengguna (dosen), termasuk menambahkan, melihat, mengubah, dan menghapus aktivitas yang ditampilkan pada sistem.	
Pemicu (<i>Trigger</i>) : Pengguna menekan tombol “Lihat Detail” atau tombol berikon <i>eye</i> pada salah satu RPS di halaman daftar RPS.	
Kondisi Awal (<i>Preconditions</i>) : Pengguna telah masuk ke dalam sistem dengan otoritas Dosen.	
Alur Normal (Melihat Aktivitas) : <ol style="list-style-type: none"> 1. Pengguna mengakses sidebar “RPS”. 2. Sistem menampilkan halaman yang berisi daftar RPS yang telah dibuat oleh pengguna (dosen). 3. Pengguna memilih RPS yang ingin ditambahkan aktivitas mingguan dengan mengklik ikon <i>eye</i>. 4. Sistem menampilkan halaman detail yang berisi informasi umum RPS dan sebuah tabel yang menampilkan daftar aktivitas mingguan di mata kuliah tersebut. 	
Alur Alternatif 1 (Menambah Aktivitas) : <ol style="list-style-type: none"> 1. Pengguna mengakses sidebar “RPS”. 2. Sistem menampilkan halaman yang berisi daftar RPS yang telah dibuat oleh pengguna (dosen). 	

3. Pengguna memilih RPS yang ingin ditambahkan aktivitas mingguan dengan mengklik ikon *eye*.
4. Sistem menampilkan halaman detail yang berisi informasi umum RPS dan sebuah tabel yang menampilkan daftar aktivitas mingguan di mata kuliah tersebut.
5. Pada halaman detail RPS, pengguna menekan tombol “Tambah Aktivitas”.
6. Sistem menampilkan modal *form* untuk menambah aktivitas.
7. Pengguna mengisi data aktivitas dan menekan “Simpan Aktivitas”.
8. Sistem memvalidasi, menyimpan data, menampilkan notifikasi keberhasilan dan memperbarui tabel aktivitas dengan data baru.

Alur Alternatif 2 (Mengubah Aktivitas) :

1. Pengguna mengakses sidebar “RPS”.
2. Sistem menampilkan halaman yang berisi daftar RPS yang telah dibuat oleh pengguna (dosen).
3. Pengguna memilih RPS yang ingin ditambahkan aktivitas mingguan dengan mengklik ikon *eye*.
4. Sistem menampilkan halaman detail yang berisi informasi umum RPS dan sebuah tabel yang menampilkan daftar aktivitas mingguan di mata kuliah tersebut.
5. Pengguna menekan tombol berikon pensil pada salah satu baris data di tabel aktivitas.
6. Sistem menampilkan modal *form* yang sudah terisi dengan data aktivitas tersebut.
7. Pengguna mengubah data dan menekan “Simpan Perubahan”.
8. Sistem memvalidasi, memperbarui data, menampilkan notifikasi keberhasilan dan memperbarui tabel aktivitas.

Alur Alternatif 3 (Menghapus Aktivitas) :

1. Pengguna mengakses sidebar “RPS”.
2. Sistem menampilkan halaman yang berisi daftar RPS yang telah dibuat oleh pengguna (dosen).

<ol style="list-style-type: none"> 3. Pengguna memilih RPS yang ingin ditambahkan aktivitas mingguan dengan mengklik ikon <i>eye</i>. 4. Sistem menampilkan halaman detail yang berisi informasi umum RPS dan sebuah tabel yang menampilkan daftar aktivitas mingguan di mata kuliah tersebut. 5. Pengguna menekan tombol berikon <i>trash</i> pada salah satu baris data di tabel aktivitas. 6. Sistem menampilkan dialog konfirmasi penghapusan. 7. Pengguna mengonfirmasi penghapusan. 8. Sistem menghapus data, memperbarui tabel aktivitas, dan menampilkan notifikasi keberhasilan.
<p>Kondisi Akhir (<i>Postconditions</i>) :</p> <ol style="list-style-type: none"> 1. Jika dosen menambah aktivitas, data aktivitas baru tersimpan dan muncul di dalam tabel aktivitas mingguan. 2. Jika dosen mengubah aktivitas, perubahan data langsung tersimpan dan diperbarui pada tabel. 3. Jika dosen menghapus aktivitas, data aktivitas tersebut hilang dari sistem dan tabel. 4. Sistem selalu menampilkan notifikasi sukses setiap kali aksi tambah, ubah, hapus berhasil dilakukan.

3.5.7.5 Mengelola CPL ke MK (UC-005)

Tabel 3.10 Deskripsi *Use Case* Mengelola CPL ke MK

Nama <i>Use Case</i> : Mengelola CPL ke MK	ID : UC-005
Aktor : Dosen	
Deskripsi : <i>Use case</i> ini menggambarkan proses pengelolaan pemetaan Capaian Pembelajaran Lulusan (CPL) dengan Mata Kuliah (MK), yang meliputi proses menambah, melihat, dan menghapus data CPL ke MK.	
Pemicu (<i>Trigger</i>) : Pengguna memilih menu CPLMK pada <i>sidebar</i> .	

Kondisi Awal (Preconditions) :

1. Pengguna telah masuk ke dalam sistem dengan otoritas dosen.
2. Data mata kuliah dan CPL untuk program studi terkait telah terdefinisi di sistem.

Alur Normal (Melihat Daftar CPLMK) :

1. Pengguna login dengan otoritas Dosen.
2. Sistem menampilkan halaman utama untuk otoritas Dosen.
3. Pengguna memilih menu CPLMK pada *sidebar*, lalu memilih sub-menu “Daftar CPLMK”
4. Sistem menampilkan halaman “Daftar CPLMK” yang berisi tabel daftar pemetaan CPL ke MK yang sudah ada, dengan kolom Kode MK, Nama MK, dan Kode CPL.

Alur Alternatif 1 (Menambah CPLMK):

1. Pengguna memilih menu “CPLMK” dan memilih sub-menu “Tambah CPLMK”.
2. Sistem menampilkan halaman *form* tambah CPLMK.
3. Pengguna memilih satu mata kuliah dari *dropdown* yang tersedia.
4. Pengguna memilih CPL Prodi dari daftar *dropdown*.
5. Pengguna menekan tombol “*Submit*”.
6. Sistem menyimpan data pemetaan baru dan menampilkan notifikasi keberhasilan.

Alur Alternatif 2 (Menghapus CPLMK):

1. Pengguna login dengan otoritas Dosen.
2. Sistem menampilkan halaman utama untuk otoritas Dosen.
3. Pengguna memilih menu CPLMK pada sidebar, lalu memilih sub-menu “Daftar CPLMK”
4. Sistem menampilkan halaman “Daftar CPLMK” yang berisi tabel daftar pemetaan CPL ke MK yang sudah ada, dengan kolom Kode MK, Nama MK, dan Kode CPL.
5. Pada data CPLMK yang ingin dihapus, pengguna menekan tombol “*delete*”

<ol style="list-style-type: none"> 6. Sistem menampilkan dialog konfirmasi penghapusan. 7. Pengguna mengonfirmasi penghapusan. 8. Sistem menghapus data pemetaan dari <i>database</i> dan menampilkan notifikasi keberhasilan.
<p>Kondisi Akhir (<i>Postconditions</i>) :</p> <ol style="list-style-type: none"> 1. Jika dosen menambah CPLMK, data pemetaan baru antara CPL dan MK tersimpan di sistem. 2. Jika dosen menghapus CPLMK, data pemetaan tersebut terhapus dari sistem. 3. Sistem selalu menampilkan notifikasi sukses setiap kali aksi tambah atau hapus berhasil dilakukan.

3.5.7.6 Mengelola CPMK (UC-006)

Tabel 3.11 Deskripsi *Use Case* Mengelola CPMK

Nama <i>Use Case</i> : Mengelola CPMK	ID : UC-006
Aktor : Dosen, Penjamin Mutu, Kepala Program Studi	
Deskripsi : <i>Use case</i> ini menggambarkan proses pengelolaan Capaian Pembelajaran Mata Kuliah (CPMK) oleh Aktor, yang meliputi proses melihat, menambah, mengubah, dan menghapus data CPMK.	
Pemicu (<i>Trigger</i>) : Pengguna memilih menu CPMK pada <i>sidebar</i> .	
Kondisi Awal (<i>Preconditions</i>) : Pengguna telah masuk ke dalam sistem dengan otoritas dosen, penjamin mutu, atau kepala program studi.	
Alur Normal (Melihat Daftar CPMK) :	
<ol style="list-style-type: none"> 1. Pengguna memilih menu CPMK pada <i>sidebar</i>, lalu memilih sub-menu “Daftar CPMK” 2. Sistem menampilkan halaman “Daftar CPMK” yang berisi data dengan kolom kode CPL, kode CPMK, rincian CPMK. 	

Alur Alternatif 1 (Menambah CPLMK):

1. Pengguna memilih menu “CPMK” dan memilih sub-menu “Tambah CPMK”.
2. Sistem menampilkan halaman *form* tambah CPMK.
3. Pengguna memilih satu mata kuliah dari *dropdown* yang tersedia.
4. Pengguna memilih CPL Prodi dari daftar *dropdown*.
5. Pengguna menekan tombol “*Submit*”.
6. Sistem menyimpan data pemetaan baru dan menampilkan notifikasi keberhasilan.

Alur Alternatif 2 (Mengubah CPMK):

1. Pengguna memilih menu CPMK pada sidebar, lalu memilih sub-menu “Daftar CPMK”
2. Sistem menampilkan halaman “Daftar CPMK” yang berisi data dengan kolom kode CPL, kode CPMK, rincian CPMK.
3. Pada data CPLMK yang ingin diubah, pengguna menekan tombol “Edit”.
4. Sistem menampilkan halaman *form* edit yang sudah terisi dengan data CPMK yang dipilih.
5. Pengguna melakukan perubahan pada data cpmk tersebut dan menekan “Submit”.
6. Sistem memperbarui data CPMK dan menampilkan notifikasi keberhasilan.

Alur Alternatif 3 (Menghapus CPMK):

1. Pengguna memilih menu CPMK pada sidebar, lalu memilih sub-menu “Daftar CPMK”
2. Sistem menampilkan halaman “Daftar CPMK” yang berisi data dengan kolom kode CPL, kode CPMK, rincian CPMK.
3. Pada data CPLMK yang ingin diubah, pengguna menekan tombol “Hapus”.
4. Sistem menampilkan dialog konfirmasi penghapusan.
5. Pengguna mengonfirmasi penghapusan.

6. Sistem menghapus data CPMK beserta semua rinciannya dan menampilkan notifikasi keberhasilan.
<p>Kondisi Akhir (<i>Postconditions</i>) :</p> <ol style="list-style-type: none"> 1. Jika pengguna menambah CPMK, data CPMK baru beserta rinciannya tersimpan di sistem. 2. Jika pengguna mengubah CPMK, perubahan data tersimpan di sistem. 3. Jika pengguna menghapus CPMK, data tersebut terhapus dari sistem. 4. Sistem selalu menampilkan notifikasi sukses setelah aksi tambah, ubah, atau hapus berhasil dilakukan.

3.5.7.7 Mengelola Komponen (UC-007)

Tabel 3.12 Deskripsi *Use Case* Mengelola Komponen

Nama <i>Use Case</i> : Mengelola Komponen	ID : UC-007
Aktor : Dosen	
Deskripsi : <i>Use case</i> ini menggambarkan proses pengelolaan komponen penilaian oleh dosen, meliputi proses melihat daftar, menambah, dan menghapus data komponen.	
Pemicu (<i>Trigger</i>) : Pengguna memilih menu Komponen pada <i>sidebar</i> .	
Kondisi Awal (<i>Preconditions</i>) : Pengguna telah masuk ke dalam sistem dengan otoritas dosen.	

Alur Normal (Melihat Daftar Komponen) :

1. Pengguna login dengan otoritas Dosen.
2. Sistem menampilkan halaman utama untuk otoritas Dosen.
3. Pengguna memilih menu “Komponen” dan memilih sub-menu “Lihat Komponen”.
4. Sistem menampilkan halaman yang berisi tabel daftar komponen penilaian yang sudah ada, dilengkapi dengan fitur pencarian dan paginasi.

Alur Alternatif 1 (Menambah Komponen):

1. Pengguna login dengan otoritas Dosen.
2. Sistem menampilkan halaman utama untuk otoritas Dosen.
3. Pengguna memilih menu “Komponen” dan memilih sub-menu “Tambah Komponen”.
4. Sistem menampilkan halaman *form* tambah komponen dengan satu *field* isian “Jenis”.
5. Pengguna mengisi nama komponen pada *field* “Jenis” dan menekan tombol “Submit”.
6. Sistem menyimpan data komponen baru dan menampilkan notifikasi keberhasilan.

Alur Alternatif 2 (Menghapus Komponen):

1. Pengguna login dengan otoritas Dosen.
2. Sistem menampilkan halaman utama untuk otoritas Dosen.
3. Pengguna memilih menu “Komponen” dan memilih sub-menu “Lihat Komponen”.
4. Sistem menampilkan halaman yang berisi tabel daftar komponen penilaian yang sudah ada, dilengkapi dengan fitur pencarian dan paginasi.
5. Pengguna menekan tombol “Delete” pada salah satu baris data komponen.
6. Sistem menampilkan dialog konfirmasi penghapusan.
7. Pengguna mengonfirmasi penghapusan.

8. Sistem menghapus data komponen dari <i>database</i> , memperbarui tabel, dan menampilkan notifikasi keberhasilan.
Kondisi Akhir (<i>Postconditions</i>) :
<ol style="list-style-type: none"> 1. Jika dosen menambah komponen, data komponen baru tersimpan di sistem dan muncul di daftar. 2. Jika dosen menghapus komponen, data komponen tersebut terhapus dari sistem. 3. Sistem selalu menampilkan notifikasi yang sesuai setelah setiap aksi berhasil dilakukan.

3.5.7.8 Memvalidasi RPS (UC-008)

Tabel 3.13 Deskripsi *Use Case* Memvalidasi RPS

Nama <i>Use Case</i> : Memvalidasi RPS	ID : UC-008
Aktor : Kepala Program Studi, Penjamin Mutu Program Studi	
Deskripsi : <i>Use case</i> ini menggambarkan proses Kaprodi/PM prodi meninjau detail sebuah RPS yang diajukan, kemudian memberikan persetujuan atau menolaknya dengan memberikan catatan revisi.	
Pemicu (<i>Trigger</i>) : Pengguna memilih menu Validasi RPS pada <i>sidebar</i> .	
Kondisi Awal (<i>Preconditions</i>) : Pengguna telah masuk ke dalam sistem dengan otoritas kaprodi.	
Alur Normal (Menyetujui RPS) :	
<ol style="list-style-type: none"> 1. Pengguna login dengan otoritas Kaprodi/ Penjamin Mutu Program Studi. 2. Sistem menampilkan halaman <i>dashboard</i> untuk otoritas Kaprodi atau penjamin mutu. 3. Pengguna memilih menu “Validasi RPS” dan sub-menu “Validasi RPS”. 4. Sistem menampilkan halaman berisi tabel daftar RPS yang menunggu untuk di validasi. 	

5. Dari halaman daftar validasi, di tiap RPS tersedia tombol “Setujui” dan “Tolak”.
6. Untuk meninjau RPS, pengguna dapat mengklik tombol berikon printer.
7. Pengguna meninjau konten RPS dan menekan tombol “Setujui”.
8. Sistem menampilkan dialog konfirmasi persetujuan.
9. Pengguna mengonfirmasi.
10. Sistem mengubah status RPS menjadi “*Published*” dan menampilkan notifikasi keberhasilan.

Alur Alternatif 1 (Menolak RPS) :

1. Pengguna login dengan otoritas Kaprodi.
2. Sistem menampilkan halaman *dashboard* untuk otoritas Kaprodi atau penjamin mutu.
3. Pengguna memilih menu “Validasi RPS” dan sub-menu “Validasi RPS”.
4. Sistem menampilkan halaman berisi tabel daftar RPS yang menunggu untuk di validasi.
5. Dari halaman daftar validasi, di tiap RPS tersedia tombol “Setujui” dan “Tolak”.
6. Untuk meninjau RPS, pengguna dapat mengklik tombol berikon printer.
7. Pengguna meninjau konten RPS dan menekan tombol “Tolak”.
8. Sistem menampilkan *form* yang mewajibkan pengguna mengisi kolom “Catatan Revisi”.
9. Pengguna mengisi alasan penolakan atau catatan perbaikan, lalu menekan tombol “Kirim Penolakan”.
10. Sistem mengubah status RPS menjadi “*rejected*”, menyimpan catatan revisi, dan menampilkan notifikasi keberhasilan.

Kondisi Akhir (*Postconditions*) :

1. Jika disetujui, status RPS berubah menjadi “*Published*” dan dapat diakses publik.

2. Jika ditolak, status RPS berubah menjadi “*Rejected*” dan catatan revisi tersimpan di sistem untuk dilihat oleh Dosen dan dosen dapat kembali melakukan edit terhadap RPS tersebut.

3.5.7.9 Melihat Daftar RPS Tervalidasi (UC-009)

Tabel 3.14 Deskripsi *Use Case* Melihat Daftar RPS Tervalidasi

Nama Use Case : Melihat Daftar RPS Tervalidasi	ID : UC-009
Aktor : Kepala Program Studi, Penjamin Mutu	
Deskripsi : <i>Use case</i> ini menggambarkan proses bagi Kaprodi dan Penjamin Mutu untuk melihat daftar semua RPS yang telah diajukan oleh Dosen dan sudah di validasi.	
Pemicu (<i>Trigger</i>) : Pengguna memilih menu Validasi RPS pada <i>sidebar</i> .	
Kondisi Awal (<i>Preconditions</i>) : Pengguna telah masuk ke dalam sistem dengan otoritas kaprodi atau penjamin mutu.	
Alur Normal (Melihat Daftar RPS Tervalidasi) : <ol style="list-style-type: none"> 1. Pengguna login dengan otoritas Kaprodi atau Penjamin Mutu. 2. Sistem menampilkan halaman <i>dashboard</i> untuk otoritas Kaprodi atau penjamin mutu. 3. Pengguna memilih menu “RPS” dan sub-menu “Daftar RPS”. 4. Sistem menampilkan halaman berisi tabel daftar RPS yang berstatus "<i>published</i>" 5. Tabel menampilkan informasi ringkas seperti Kode & Nama Mata Kuliah, Pengembang RPS, dan Tanggal Validasi. 6. Pengguna dapat menggunakan fitur pencarian atau paginasi untuk menemukan RPS tertentu. 	
Kondisi Akhir (<i>Postconditions</i>) : Sistem berhasil menampilkan daftar RPS yang sudah divalidasi oleh Kaprodi.	

3.5.8 Spesifikasi Perangkat Lunak

Spesifikasi perangkat keras dan perangkat lunak merupakan dokumen yang menjelaskan kebutuhan perangkat yang diperlukan untuk mendukung jalannya aplikasi. Rincian spesifikasi perangkat keras dan perangkat lunak dapat dilihat pada

Tabel 3.15 Spesifikasi Perangkat Keras dan Perangkat Lunak

No	Spesifikasi	Server	Client
1	Sistem	Linux (Kernel Version 4.18.0-	Windows 10/11/macOS/
	Operasi	553.27.1.lve.el8.x86_64)	Linux
2	Perangkat Lunak	<ul style="list-style-type: none"> • Database Server (MySQL 8.0.41) • Framework (Laravel 8) 	Web browser (Google Chrome, Mozilla Firefox, Safari, Microsoft Edge, dll)

3.5.9 Skenario Pengujian Fungsional

Untuk memastikan fungsionalitas sistem, penelitian ini mengaplikasikan metode pengujian *black-box*. Serangkaian skenario pengujian *black-box* telah disusun dan disajikan pada Tabel 3.16.

Tabel 3.16 Skenario Pengujian Fungsional

No	Testing ID	Objective/ Output	Kasus Uji	Tipe Uji	Role	Hasil yang diharapkan	Hasil Sebenarnya	Status Pengujian
1a	BBT-001a/UC-001/FR-001	Sistem harus menampilkan RPS publik saat halaman diakses oleh tamu.	User mengakses halaman utama dan melihat daftar RPS.	Positif	Tamu	Sistem menampilkan halaman utama (<i>landing page</i>) dengan daftar RPS berstatus published. Setiap RPS menampilkan nama, kode, semester, jumlah sks, deskripsi mata kuliah, dan <i>button</i> unduh RPS.		
1b	BBT-001b/UC-001/FR-001	Sistem harus berhasil menyaring daftar RPS sesuai kata kunci pencarian yang valid.	User memasukkan kode atau nama mata kuliah yang <i>valid</i> dan ada di sistem ke dalam kolom pencarian.	Positif	Tamu	Sistem menyaring Daftar RPS dan hanya menampilkan data yang cocok dengan kata kunci pencarian.		
1c	BBT-001c/UC-001/FR-001	Sistem harus menampilkan pesan 'data tidak ditemukan' jika hasil pencarian kosong.	User memasukkan kode atau nama mata kuliah yang tidak ada di sistem ke dalam kolom pencarian.	Negatif	Tamu	Sistem menampilkan pesan informatif seperti "RPS tidak ditemukan".		

No	Testing ID	Objective/ Output	Kasus Uji	Tipe Uji	Role	Hasil yang diharapkan	Hasil Sebenarnya	Status Pengujian
2	BBT-002/UC-002/FR-002	Sistem harus berhasil mengunduh file RPS yang dipilih dari daftar.	Setelah daftar RPS (baik dari daftar awal maupun hasil pencarian) tampil, <i>user</i> menekan tombol “Unduh RPS” pada salah satu RPS.	Positif	Tamu	Sistem memproses permintaan dan <i>file</i> RPS yang sesuai berhasil diunduh.		
3a	BT-003a/UC-003/FR-003	Sistem dapat menampilkan daftar RPS milik <i>user</i> (dosen) yang login.	<i>User</i> login dan mengakses menu “RPS” di <i>sidebar</i> .	Positif	Dosen	Sistem menampilkan halaman yang berisi daftar RPS yang hanya dimiliki oleh Dosen tersebut.		
3b	BT-003b/UC-003/FR-003	<i>User</i> dapat menambahkan data RPS baru.	<i>User</i> membuka form “Tambah RPS”, mengisi seluruh kolom wajib dengan data yang valid, lalu menekan tombol “Simpan RPS”.	Positif	Dosen	Sistem menyimpan data RPS baru ke dalam <i>database</i> , menampilkan notifikasi keberhasilan, dan RPS baru tersebut muncul dalam daftar.		
3c	BBT-003c/UC-003/FR-003	<i>User</i> gagal menambahkan RPS jika terdapat data wajib yang tidak diisi.	<i>User</i> mencoba menyimpan form “Tambah RPS” dengan membiarkan satu atau lebih kolom wajib dalam keadaan kosong.	Negatif	Dosen	Sistem menolak penyimpanan data, tidak menambahkan RPS baru, dan menampilkan pesan error yang relevan pada isian <i>form</i> yang tidak valid.		

No	Testing ID	Objective/ Output	Kasus Uji	Tipe Uji	Role	Hasil yang diharapkan	Hasil Sebenarnya	Status Pengujian
3d	BBT-003d/UC-003/FR-003	User dapat mengubah data pada RPS yang sudah ada.	User memilih salah satu RPS dari daftar, menekan <i>button</i> berikon pensil untuk memodifikasi data pada form, dan menekan “Simpan Perubahan”.	Positif	Dosen	Sistem memperbarui data RPS di <i>database</i> , dan memunculkan notifikasi keberhasilan.		
3e	BBT-003e/UC-003/FR-003	User dapat menghapus RPS yang sudah ada.	User memilih RPS, menekan <i>button</i> berikon <i>trash</i> , dan menekan “Ya” atau “Konfirmasi” pada dialog konfirmasi penghapusan.	Positif	Dosen	Sistem menghapus data RPS dari <i>database</i> , menghilangkan RPS tersebut dari daftar, dan memunculkan notifikasi keberhasilan.		
3f	BBT-003f/UC-003/FR-003	User dapat mencetak atau menyimpan RPS.	User memilih RPS dan menekan <i>button</i> berikon <i>printer</i> pada RPS tersebut.	Positif	Dosen	Sistem menghasilkan pratinjau dokumen RPS dan membuka dialog pencetakan browser.		
3g	BBT-003g/UC-003/FR-003	User dapat mengajukan validasi RPS yang berstatus “Draft”.	User memilih RPS yang siap divalidasi, menekan <i>button</i> biru gelap berikon <i>up</i> (ajukan validasi), dan	Positif	Dosen	Sistem mengubah status RPS menjadi “Pending”, menonaktifkan fungsi ubah, lihat dan hapus untuk RPS tersebut, dan meneruskan data ke alur kerja validasi Kaprodi.		

No	Testing ID	Objective/ Output	Kasus Uji	Tipe Uji	Role	Hasil yang diharapkan	Hasil Sebenarnya	Status Pengujian
			mengonfirmasi tindakan tersebut.					
4a	BBT-004a/UC-004/FR-004	User dapat melihat detail informasi sebuah RPS beserta daftar aktivitas mingguannya.	User menekan tombol berikon mata pada salah satu RPS dari daftar.	Positif	Dosen	Sistem menampilkan halaman detail yang berisi informasi umum RPS dan tabel aktivitas mingguan yang terkait dengan RPS tersebut.		
4b	BBT-004b/UC-004/FR-004	User dapat menambahkan aktivitas mingguan baru.	Pada halaman detail RPS, User menekan "Tambah Aktivitas", mengisi form pada modal dengan data valid, dan menekan "Simpan Aktivitas".	Positif	Dosen	Sistem menyimpan data dan menampilkan aktivitas baru tersebut di dalam tabel aktivitas mingguan beserta notifikasi sukses.		
4c	BBT-004c/UC-004/FR-004	User gagal menambahkan aktivitas jika data pada form tidak lengkap atau tidak valid.	User mencoba menyimpan form tambah aktivitas dengan membiarkan kolom wajib seperti deskripsi sub cpmk dalam keadaan kosong.	Negatif	Dosen	Sistem menolak penyimpanan dan menampilkan pesan error pada isian form yang wajib diisi namun kosong,		
4d	BBT-004d/UC-004/FR-004	User dapat mengubah data aktivitas	User menekan tombol berikon pensil pada salah satu aktivitas,	Positif	Dosen	Sistem memperbarui data di database, menampilkan perubahan tersebut pada tabel		

No	Testing ID	Objective/ Output	Kasus Uji	Tipe Uji	Role	Hasil yang diharapkan	Hasil Sebenarnya	Status Pengujian
		mingguan yang sudah ada.	memodifikasi data di form, dan menekan "Simpan Perubahan".			aktivitas, dan memunculkan notifikasi sukses.		
4e	BBT-004e/UC-004/FR-004	User dapat menghapus data aktivitas mingguan yang sudah ada.	User menekan tombol berikon <i>trash</i> pada salah satu aktivitas dan memberikan konfirmasi pada dialog penghapusan yang muncul.	Positif	Dosen	Sistem menghapus data dari <i>database</i> , menghilangkan baris aktivitas tersebut dari tabel, dan memunculkan notifikasi sukses.		
5a	BBT-005a/UC-005/FR-005	User dapat melihat daftar pemetaan CPL ke MK yang sudah ada.	User mengakses menu "CPLMK" dan sub-menu "Daftar CPLMK".	Positif	Dosen	Sistem menampilkan halaman berisi tabel pemetaan CPLMK dengan kolom Kode MK, Nama MK, dan Kode CPL.		
5b	BBT-005b/UC-005/FR-005	User dapat menambahkan CPL baru ke sebuah Mata Kuliah.	User mengakses sub-menu "Tambah CPLMK", lalu memilih Mata Kuliah dan CPL dari <i>dropdown</i> , lalu menekan "Submit".	Positif	Dosen	Sistem menyimpan data pemetaan baru, menampilkannya di daftar CPLMK, dan memunculkan notifikasi sukses.		
5c	BBT-005c/UC-005/FR-005	User gagal menambahkan pemetaan jika salah	User mencoba menyimpan <i>form</i> "Tambah CPLMK" tanpa memilih Mata	Negatif	Dosen	Sistem menolak penyimpanan dan menampilkan pesan <i>error</i> yang mengindikasikan kolom mana yang wajib diisi.		

No	Testing ID	Objective/ Output	Kasus Uji	Tipe Uji	Role	Hasil yang diharapkan	Hasil Sebenarnya	Status Pengujian
		satu data (MK atau CPL) tidak dipilih.	Kuliah atau CPL pada <i>dropdown</i> .					
5d	BBT-005d/UC-005/FR-005	User dapat menghapus pemetaan CPL ke MK yang sudah ada.	User menekan tombol "delete" pada salah satu baris data CPLMK dan memberikan konfirmasi.	Positif	Dosen	Sistem menghapus data tersebut dari <i>database</i> , menghilangkan data tersebut dari tabel, dan memunculkan notifikasi sukses.		
6a	BBT-006a/UC-006/FR-006	User dapat melihat daftar CPMK beserta Sub-CPMK.	User mengakses menu "Daftar CPMK".	Positif	Dosen, Kaprodi, Penjamin Mutu	Sistem menampilkan daftar CPMK dan Sub-CPMK.		
6b	BBT-006b/UC-006/FR-006	User dapat menambahkan CPMK dan Sub-CPMK baru.	User mengakses menu "Tambah CPMK".	Positif	Dosen, Kaprodi, Penjamin Mutu	Sistem menyimpan data CPMK dan Sub-CPMK baru.		
6c	BBT-006d/UC-006/FR-006	User dapat menghapus data CPMK atau Sub-CPMK yang sudah ada.	User menekan tombol "Delete" pada salah satu CPMK dan memberikan konfirmasi.	Positif	Dosen, Kaprodi, Penjamin Mutu	Sistem menghapus data CPMK dan Sub-CPMK atau hanya Sub-CPMK tergantung yang dipilih dari <i>database</i> dan memunculkan notifikasi sukses.		
7a	BBT-007a/UC-007/FR-007	User dapat Menambah komponen penilaian	Menambahkan komponen penilaian.	Positif	Dosen	Sistem menerima data komponen penilaian baru, menyimpan ke		

No	Testing ID	Objective/ Output	Kasus Uji	Tipe Uji	Role	Hasil yang diharapkan	Hasil Sebenarnya	Status Pengujian
						database, dan menampilkan komponen baru tersebut beserta notifikasi sukses.		
7b	BBT-007b/UC-007/FR-007	User dapat melihat daftar seluruh komponen penilaian yang ada.	Melihat daftar komponen penilaian.	Positif	Dosen	Sistem menampilkan halaman berisi tabel yang memuat semua data komponen penilaian (jenis dan aksi hapus).		
7c	BBT-007c/UC-007/FR-007	User dapat menghapus komponen penilaian yang sudah ada.	Menghapus daftar komponen penilaian.	Positif	Dosen	Sistem menghapus data komponen dari <i>database</i> , menghilangkan baris tersebut dari tabel, dan menampilkan notifikasi sukses.		
8a	BBT-009a/UC-009/FR-009	User dapat melihat daftar RPS yang perlu divalidasi.	Mengakses sub-menu "Validasi RPS" pada menu RPS.	Positif	Kaprodi, PM Prodi	Sistem menampilkan halaman yang berisi daftar RPS yang berstatus "Pending".		
8b	BBT-009b/UC-009/FR-009	User dapat menyetujui (memvalidasi) RPS yang diajukan.	Memilih salah satu RPS dari daftar, meninjau isinya, lalu menekan tombol "Setujui" dan memberikan konfirmasi.	Positif	Kepala Program Studi, PM Prodi	Sistem mengubah status RPS menjadi "Published", menampilkan notifikasi keberhasilan, dan RPS tersebut hilang dari daftar validasi.		
8c	BBT-009c/UC-009/FR-009	User dapat menolak RPS dan memberikan catatan revisi	Memilih salah satu RPS, menekan tombol "Tolak",	Positif	Kepala Program	Sistem mengubah status RPS menjadi "Rejected", mengembalikan RPS ke dosen		

No	Testing ID	Objective/ Output	Kasus Uji	Tipe Uji	Role	Hasil yang diharapkan	Hasil Sebenarnya	Status Pengujian
			mengisi kolom "Catatan Penolakan", lalu menekan "Kirim".		Studi, PM Prodi	untuk diperbaiki beserta catatan revisi, dan menampilkan notifikasi keberhasilan.		
9	BBT-008/UC-008/FR-008	User dapat melihat daftar seluruh RPS yang telah tervalidasi.	Mengakses sub-menu "Daftar RPS" pada menu RPS.	Positif	Penjamin Mutu, Kepala Program Studi	Sistem menampilkan halaman berisi tabel yang memuat semua RPS yang berstatus "Published", lengkap dengan fitur pencarian dan filter.		

3.5.10 Skenario Pengujian Non-Fungsional

Untuk memperoleh bukti validasi empiris mengenai keandalan, keamanan, dan performa sistem, serangkaian skenario pengujian non-fungsional telah disusun pada Tabel 3.17. Setiap skenario di bawah ini merepresentasikan kasus uji spesifik yang dirancang untuk menjawab kebutuhan non-fungsional yang telah diidentifikasi sebelumnya.

Tabel 3.17 Skenario Pengujian Non-Fungsional

No	ID Kebutuhan	Aspek yang diuji	Skenario Pengujian	Alat Bantu	Kriteria Kelulusan
1	NFR-001	Usability	Memastikan sistem memberikan umpan balik (notifikasi sukses atau	Observasi manual	- Setelah berhasil menyimpan/menghapus data, muncul notifikasi sukses yang jelas.

No	ID Kebutuhan	Aspek yang diuji	Skenario Pengujian	Alat Bantu	Kriteria Kelulusan
		(Kemudahan Penggunaan)	pesan <i>error</i>) yang jelas dan informatif setelah setiap aksi penting pengguna.		- Saat terjadi kesalahan input (<i>form</i> wajib kosong), muncul pesan error spesifik di dekat kolom yang salah.
2	NFR-002	<i>Security</i> (Kontrol Akses)	Memverifikasi bahwa pengguna publik (Tamu) hanya dapat mengakses RPS yang statusnya " <i>Published</i> " dan tidak dapat melihat RPS yang masih " <i>Draft</i> ".	Observasi manual	- RPS dengan status " <i>Draft</i> " tidak ditemukan di halaman publik. - Setelah status menjadi " <i>Published</i> ", RPS tersebut berhasil ditemukan dan dapat diakses dari halaman publik.
3	NFR-003	<i>Auditability</i>	Memastikan sistem secara otomatis membuat versi baru saat RPS untuk mata kuliah yang sama ditambahkan, tanpa menimpa versi yang sudah ada.	Observasi manual	- Sistem berhasil menyimpan RPS kedua sebagai entri baru. - RPS yang pertama masih ada di dalam sistem sebagai arsip dan dapat diakses. - Terdapat penanda yang jelas untuk membedakan kedua versi.
4	NFR-004	<i>Performance</i>	Mengukur waktu respons dan stabilitas sistem di bawah beban pengguna simultan untuk memastikan sistem tetap dapat diakses dan berfungsi dengan wajar.	Apache Jmeter	- Waktu respons rata-rata (<i>average response time</i>) tidak melebihi 5 detik. - Tingkat kesalahan (<i>error rate</i>) yang tercatat tidak lebih dari 2%. - Sistem tidak mengalami <i>crash</i> selama pengujian.

3.5.11 Validasi *Prototype*

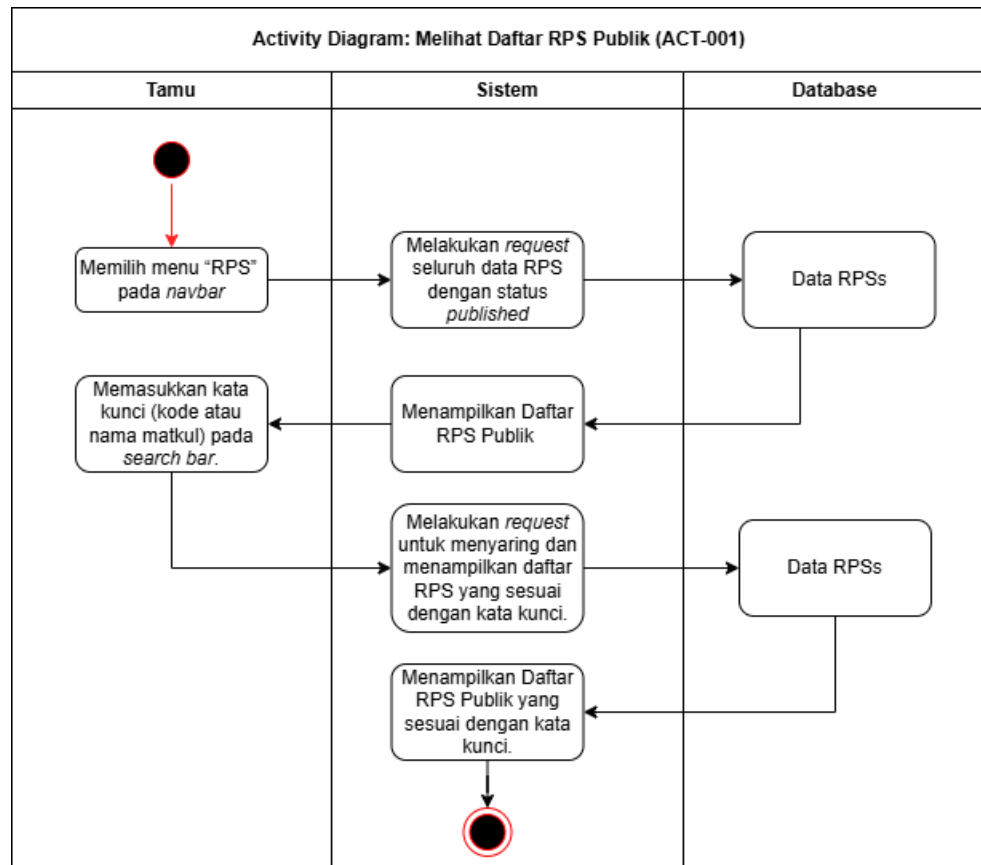
Validasi *prototype* pada penelitian ini dilakukan melalui diskusi langsung dengan *stakeholder*. Setiap hasil pengembangan *prototype* diuji, kemudian dikonsultasikan dalam sesi bimbingan untuk memperoleh masukan secara iteratif. Validasi dinyatakan berhasil apabila masukan telah diterapkan dan tidak terdapat koreksi lebih lanjut. Pendekatan ini dipilih karena proses pengembangan dilakukan secara langsung dengan metode *live prototyping*.

3.5.12 *Activity Diagram*

Activity diagram digunakan untuk memodelkan alur kerja (*workflow*) dari setiap fungsionalitas sistem. Diagram ini menjabarkan secara rinci langkah-langkah proses yang sebelumnya telah diidentifikasi dalam *use case* diagram. Berikut adalah diagram aktivitas yang relevan untuk sistem yang dikembangkan.

a. **Melihat Daftar RPS Publik (ACT-001)**

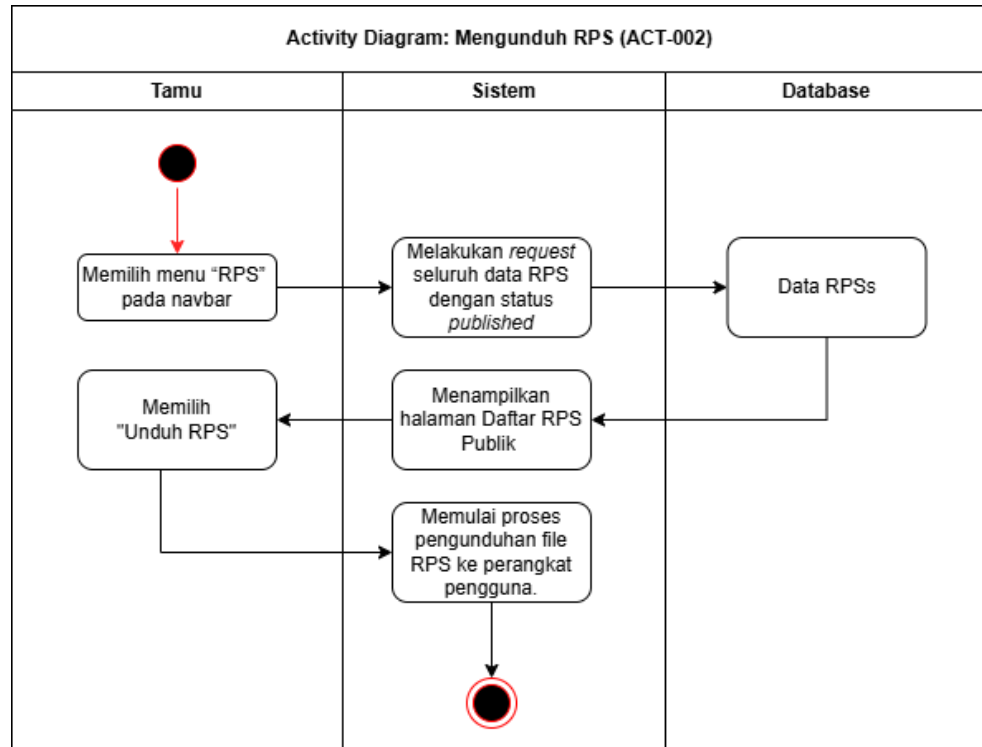
Activity Diagram pada Gambar 3.4 menjelaskan alur untuk melihat daftar RPS Publik oleh pengguna tamu. Proses dimulai ketika tamu mengklik menu “RPS” pada navbar, lalu sistem mengirim *request* untuk mengambil seluruh data RPS dari *database* yang berstatus *published*. Selanjutnya, alur aktivitas memberikan dua kemungkinan. Pertama, pengguna dapat selesai melihat daftar tersebut. Kedua, pengguna dapat berinteraksi dengan memasukkan kata kunci (kode atau nama mata kuliah) pada *search bar* untuk melakukan pencarian. Sistem kemudian akan memproses permintaan tersebut dengan menyaring data dan menampilkan kembali daftar RPS publik yang relevan dengan kata kunci pencarian pengguna.



Gambar 3.4 *Activity Diagram* Melihat Daftar RPS Publik

b. Mengunduh RPS (ACT-002)

Activity diagram pada Gambar 3.5 menjelaskan alur bagi pengguna tamu untuk mengunduh Rencana Pembelajaran Semester (RPS). Proses diawali saat tamu memilih menu “RPS” pada navbar. Sistem kemudian melakukan *request* untuk mengambil seluruh data RPS yang berstatus *published* dari *database* dan menampilkannya dalam halaman Daftar RPS Publik. Selanjutnya, pengguna memilih opsi “Unduh RPS”, yang memicu sistem untuk memulai proses pengunduhan file RPS ke perangkat pengguna.



Gambar 3.5 *Activity Diagram* Mengunduh RPS

c. Mengelola RPS (ACT-003)

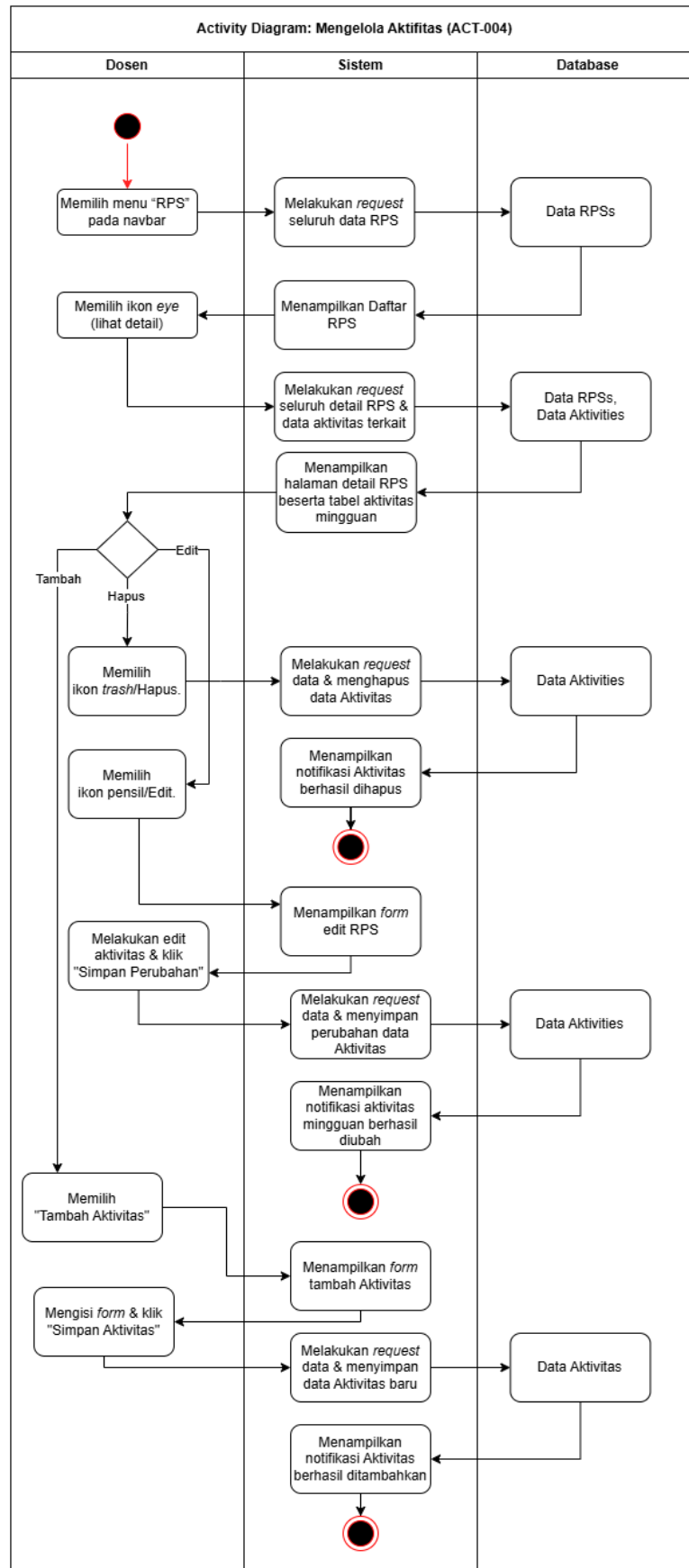
Activity Diagram pada Gambar 3.6 merincikan alur bagi Dosen dalam mengelola RPS. Proses diawali saat Dosen memilih menu “RPS”, kemudian sistem akan menampilkan Daftar RPS yang telah dibuat. Dari halaman ini, terdapat empat kemungkinan alur. Pertama menambah RPS, Dosen memilih “Tambah RPS”, mengisi *form*, dan menyimpannya. Kedua mengedit, Dosen memilih ikon pensil, melakukan perubahan pada *form*, dan menyimpannya. Ketiga menghapus, Dosen memilih ikon hapus, dan sistem akan menghapus data terkait. Keempat mencetak RPS, Dosen memilih ikon printer, sistem dengan menampilkan pratinjau beserta dialog pencetakan. Kelima mengajukan validasi, Dosen memilih ikon “Ajukan Validasi”, melakukan konfirmasi, dan sistem akan mengubah status RPS menjadi “*pending*”. Setiap proses edit, hapus, dan pengajuan validasi akan diakhiri dengan notifikasi keberhasilan dari sistem.

d. Mengelola Aktivitas (ACT-004)

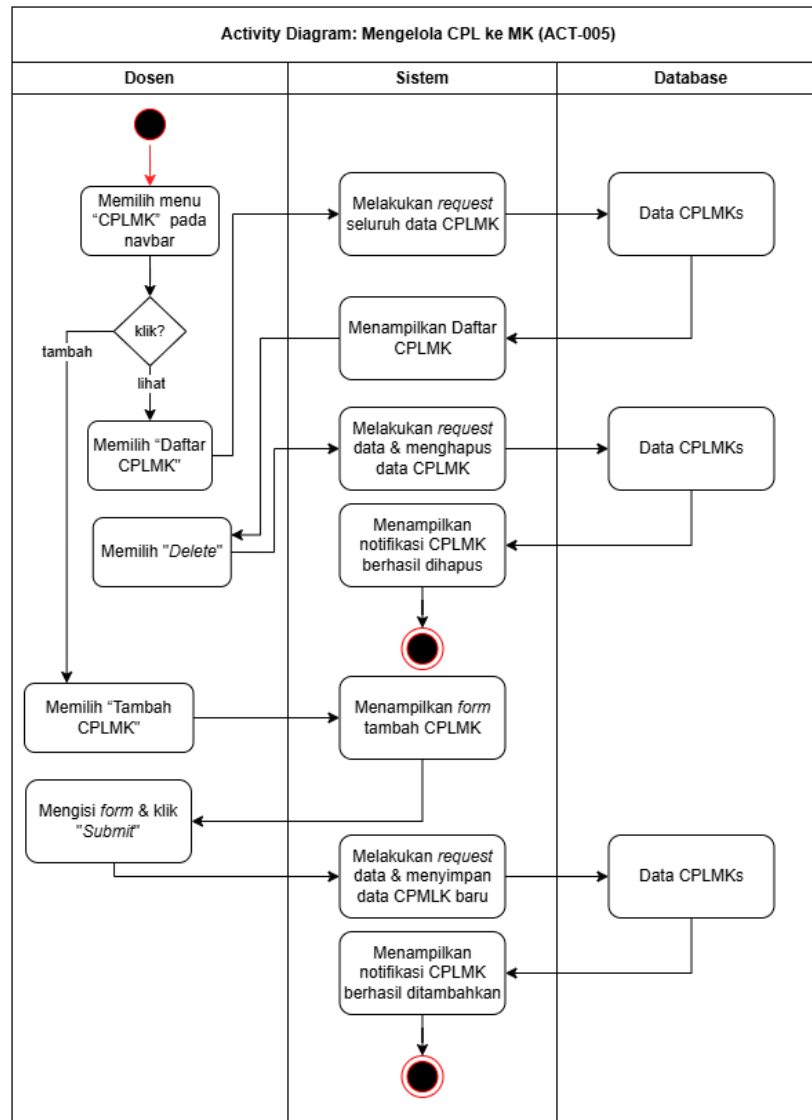
Activity Diagram pada Gambar 3.7 menjelaskan alur pengelolaan aktivitas mingguan dalam sebuah RPS oleh Dosen. Proses dimulai saat Dosen memilih ikon *eye* untuk melihat detail sebuah RPS. Sistem akan menampilkan halaman detail RPS beserta tabel aktivitas mingguan. Dari sini, Dosen memiliki tiga opsi yaitu, menambah, mengedit, atau menghapus aktivitas. Untuk menambah, Dosen memilih “Tambah Aktivitas”, mengisi form yang ditampilkan, lalu menyimpannya. Untuk mengedit, Dosen memilih ikon pensil, melakukan perubahan pada *form* edit, lalu menyimpannya. Untuk menghapus, Dosen memilih ikon hapus. Setiap tindakan tersebut akan diproses oleh sistem dengan mengirimkan *request* ke *database* dan diakhiri dengan penampilan notifikasi bahwa data aktivitas berhasil ditambahkan, diubah, atau dihapus.

e. Mengelola CPL ke MK (ACT-005)

Activity Diagram pada Gambar 3.8 menggambarkan alur bagi Dosen untuk mengelola data Capaian Pembelajaran Lulusan Mata Kuliah (CPLMK). Alur dimulai ketika Dosen memilih menu untuk melihat daftar CPLMK. Sistem kemudian akan menampilkan seluruh data CPLMK. Pada halaman ini, terdapat dua opsi utama. Pertama, Dosen dapat menghapus data dengan memilih opsi “Delete”, yang akan diproses oleh sistem dan diakhiri notifikasi berhasil. Kedua, Dosen dapat menambah data dengan memilih “Tambah CPLMK”, mengisi *form*, dan menekan “Submit”. Sistem akan menyimpan data baru tersebut dan menampilkan notifikasi bahwa komponen berhasil ditambahkan.



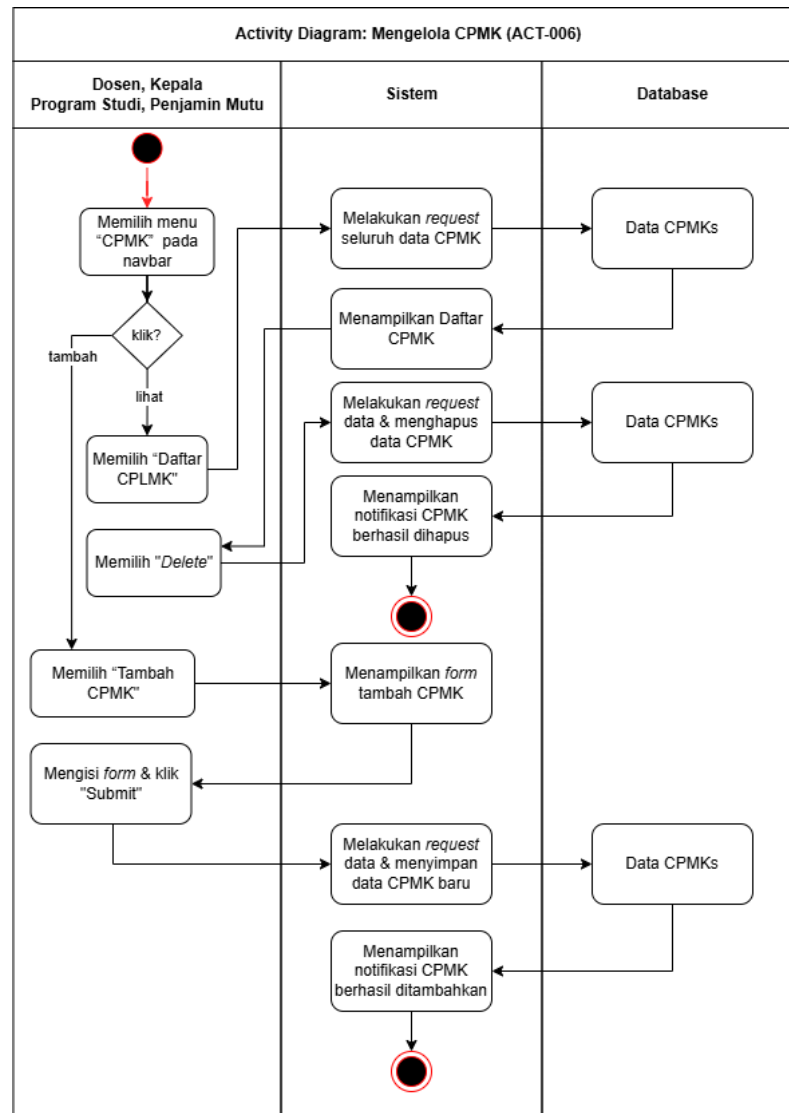
Gambar 3.7 Activity Diagram Mengelola Aktifitas



Gambar 3.8 *Activity Diagram* Mengelola CPLMK

f. Mengelola CPMK (ACT-006)

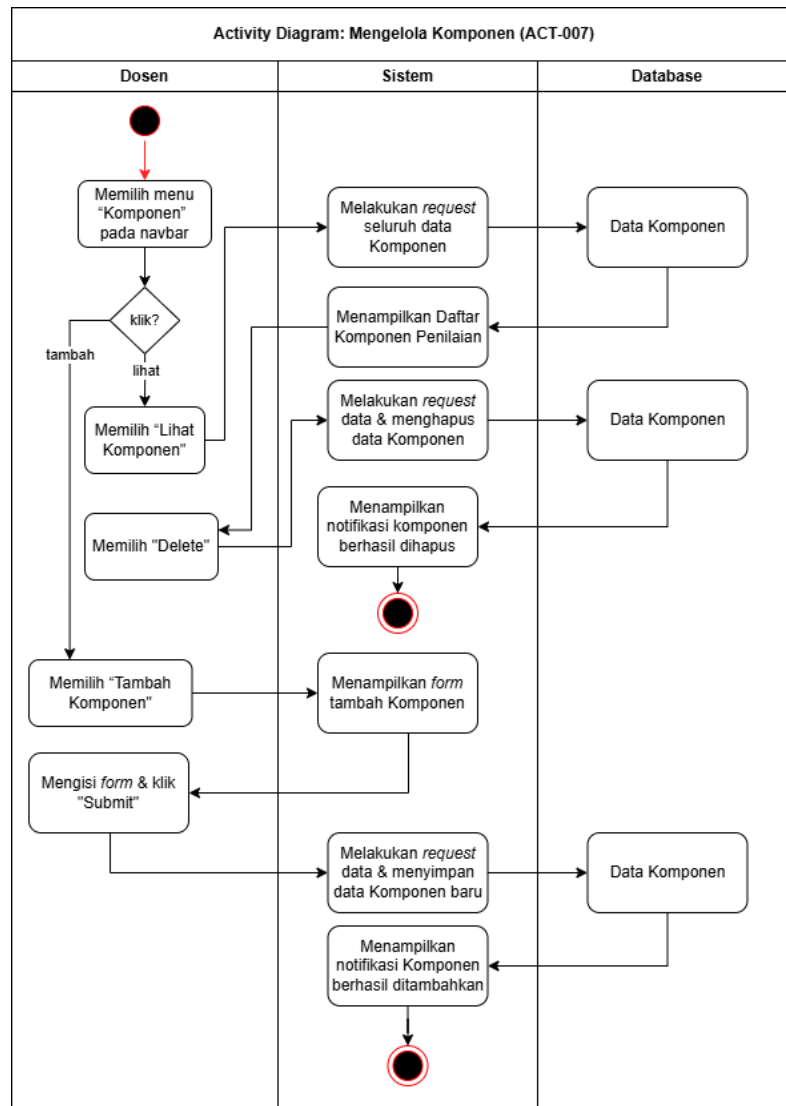
Activity Diagram pada Gambar 3.9 menjelaskan alur bagi Dosen, Kepala Program Studi, dan Penjamin Mutu dalam mengelola Capaian Pembelajaran Mata Kuliah (CPMK). Proses dimulai saat pengguna memilih untuk melihat “Daftar CPMK”. Sistem akan mengambil data dari *database* dan menampilkannya. Selanjutnya, pengguna memiliki dua pilihan. Untuk menghapus, pengguna memilih “Delete”, lalu sistem akan memproses penghapusan dan menampilkan notifikasi berhasil. Untuk menambah, pengguna memilih “Tambah CPMK”, mengisi *form* yang tersedia, lalu menekan “Submit”. Sistem akan menyimpan data CPMK yang baru dan menampilkan notifikasi keberhasilan.



Gambar 3.9 *Activity Diagram* Mengelola CPMK

g. Mengelola Komponen (ACT-007)

Activity Diagram pada Gambar 3.10 merincikan alur bagi Dosen untuk mengelola komponen penilaian. Proses diawali ketika Dosen memilih "Lihat Komponen", lalu sistem menampilkan daftar komponen penilaian. Dari tampilan ini, alur memberikan dua kemungkinan. Pertama, Dosen dapat memilih "Delete" untuk menghapus komponen, di mana sistem akan memrosesnya dan menampilkan notifikasi berhasil dihapus. Kedua, Dosen dapat memilih "Tambah Komponen" untuk membuat komponen baru. Setelah itu, Dosen akan mengisi *form* dan menekan "Submit", yang akan direspons sistem dengan menyimpan data baru dan menampilkan notifikasi komponen berhasil ditambahkan.



Gambar 3.10 *Activity Diagram* Mengelola Komponen

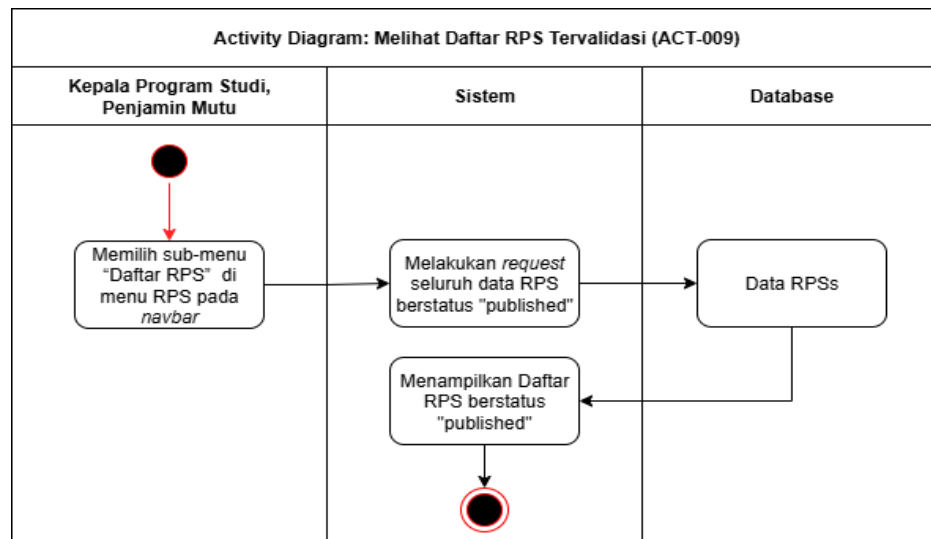
h. Memvalidasi RPS (ACT-008)

Activity Diagram pada Gambar 3.12 menjelaskan proses validasi RPS oleh Kepala Program Studi. Alur dimulai ketika Kepala Program Studi memilih sub-menu “Validasi RPS”. Sistem akan menampilkan daftar RPS yang berstatus “*pending*”. Pengguna dapat memilih untuk melihat detail RPS, yang akan menampilkan *preview* dokumen. Untuk validasi, terdapat dua pilihan utama. Pertama, memilih “Setujui RPS”, di mana sistem akan menyimpan perubahan status RPS menjadi “*published*” dan menampilkan notifikasi berhasil. Kedua, memilih “Tolak RPS”, di mana sistem akan menampilkan *pop up form* untuk mengisi catatan penolakan. Setelah diisi

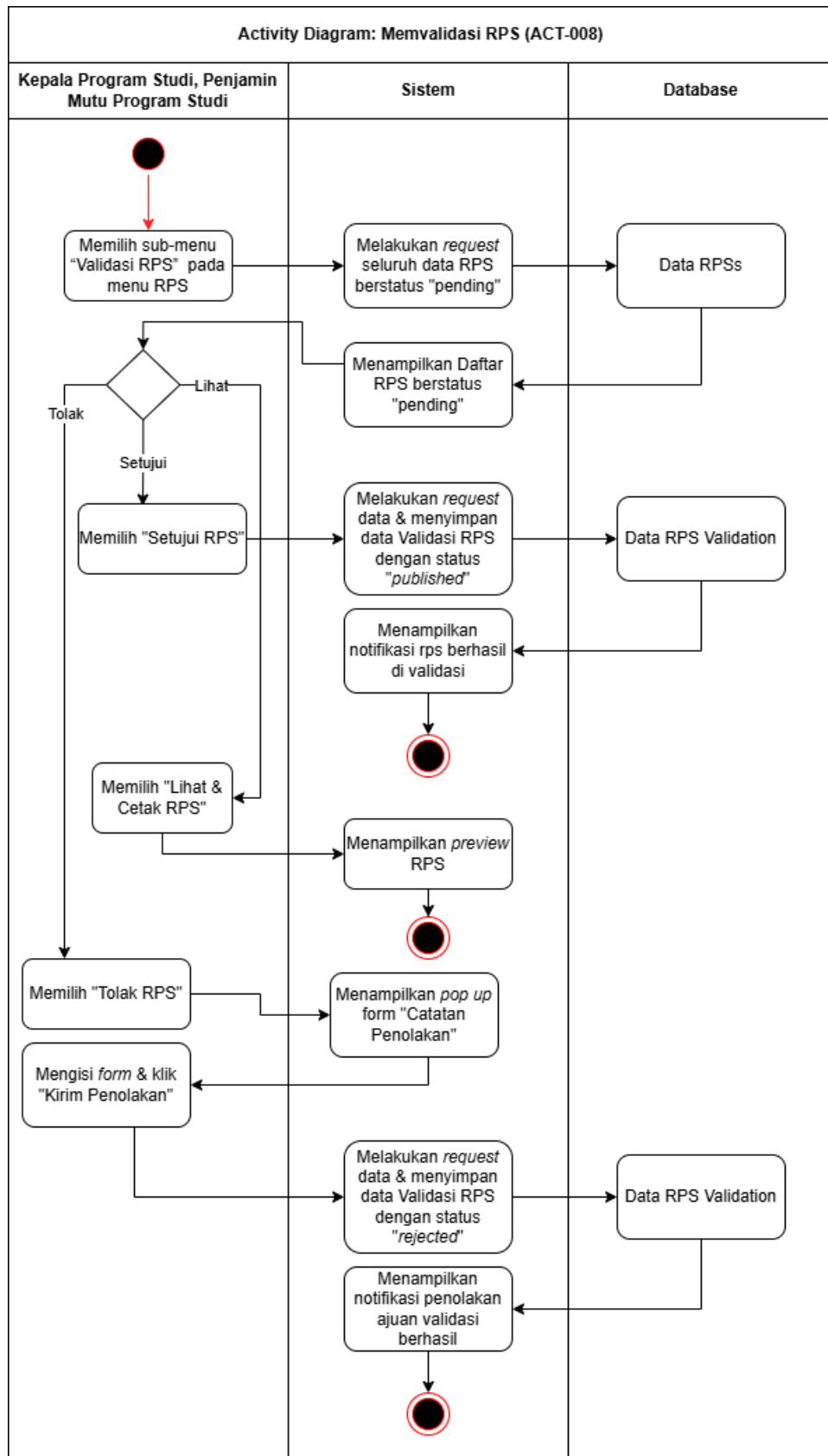
dan dikirim, sistem akan menyimpan data validasi dengan status “*rejected*” dan menampilkan notifikasi bahwa penolakan berhasil diajukan.

i. Melihat Daftar RPS Tervalidasi (ACT-009)

Activity Diagram pada Gambar 3.11 menunjukkan alur bagi Kepala Program Studi dan Penjamin Mutu untuk melihat daftar RPS yang telah tervalidasi. Prosesnya sangat sederhana, dimulai ketika pengguna memilih sub-menu “Daftar RPS” di dalam menu RPS pada navbar. Sistem kemudian akan langsung melakukan request ke database untuk mengambil seluruh data RPS yang berstatus “*published*”. Setelah data diterima, sistem akan menampilkannya dalam bentuk daftar kepada pengguna, dan alur selesai.



Gambar 3.11 *Activity Diagram* Melihat Daftar RPS Tervalidasi



Gambar 3.12 Activity Diagram Memvalidasi RPS

3.5.13 Class Diagram

Class Diagram merupakan salah satu diagram struktur dalam *Unified Modeling Language* (UML) yang berfungsi untuk merepresentasikan kelas, atribut, metode (*operations*), serta hubungan antar kelas dalam suatu sistem perangkat lunak. Diagram ini memberikan gambaran statis mengenai arsitektur sistem dengan menunjukkan komponen utama serta keterkaitannya.

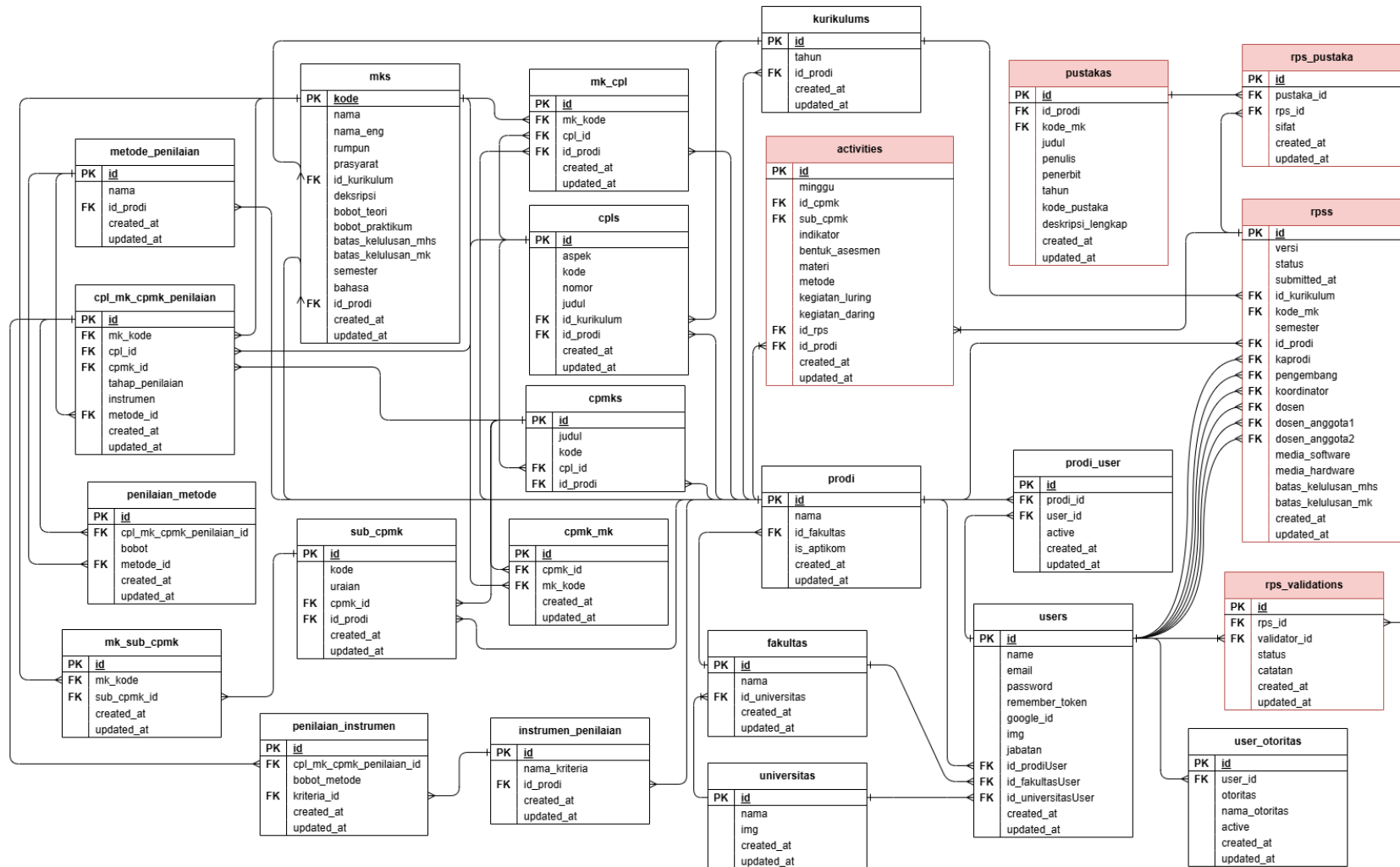
Secara umum, sebuah *Class Diagram* divisualisasikan dalam bentuk kotak yang terbagi menjadi tiga bagian yaitu nama kelas, atribut kelas, dan metode kelas. Selain itu, hubungan antar kelas seperti asosiasi (*association*), agregasi (*aggregation*), komposisi (*composition*), maupun pewarisan (*inheritance* atau *generalization*) turut digambarkan untuk memperlihatkan pola interaksi dan ketergantungan antar kelas.

Dalam penelitian ini, *Class Diagram* digunakan untuk memodelkan struktur sistem Rencana Pembelajaran Semester (RPS). Diagram yang disajikan memuat kelas-kelas utama, atribut penting, serta metode utama yang mendukung pengelolaan RPS, mulai dari proses pembuatan, pengeditan, validasi, hingga distribusi kepada pihak terkait. Dengan demikian, *Class Diagram* ini membantu memberikan pemahaman menyeluruh mengenai struktur logis sistem yang dikembangkan. Gambaran lengkap *Class Diagram* dari fitur RPS yang dikembangkan dapat dilihat pada Gambar 3.13.

3.5.14 *Entity Relationship Diagram (ERD)*

Entity Relationship Diagram (ERD) merupakan salah satu model data konseptual yang digunakan dalam perancangan basis data untuk merepresentasikan struktur logis data yang akan dikelola. ERD memvisualisasikan entitas sebagai objek utama yang relevan dalam sistem, beserta atribut-atribut yang mendeskripsikan karakteristiknya. Relasi antar entitas digambarkan melalui garis penghubung yang menjelaskan keterkaitan antar data, dengan disertai kardinalitas yang menunjukkan jumlah keterlibatan antar entitas, seperti *one-to-one*, *one-to-many*, atau *many-to-many*. Model ini memberikan landasan yang jelas dan sistematis dalam merancang skema basis data fisik sehingga mendukung konsistensi, integritas, dan efisiensi penyimpanan data.

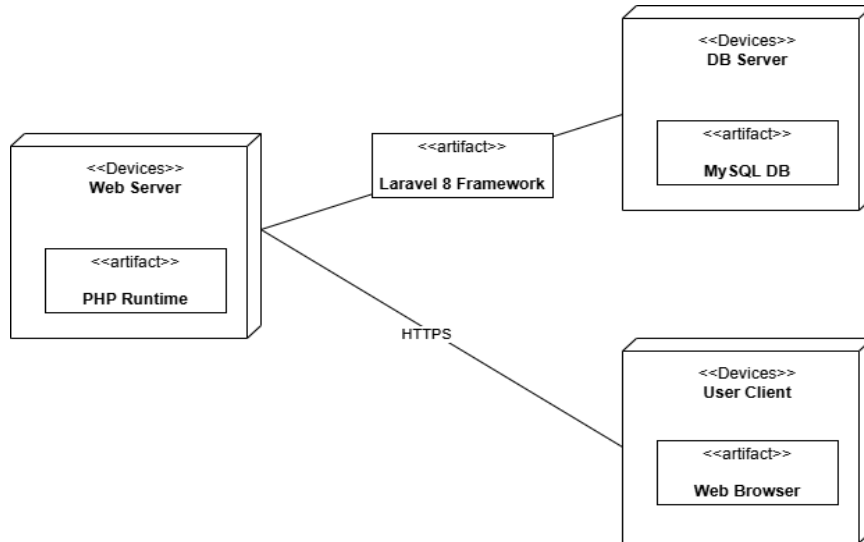
Dalam penelitian ini, ERD digunakan untuk memodelkan hubungan antar data yang berkaitan dengan pengelolaan Rencana Pembelajaran Semester (RPS). Diagram ini menggambarkan entitas utama, seperti *User*, RPS, Mata Kuliah, Prodi, Fakultas, CPMK, CPL, dan Sub-CPMK, beserta relasi yang menghubungkannya. Dengan adanya ERD ini, struktur basis data dapat dipahami secara lebih komprehensif sehingga mempermudah proses implementasi sistem. *Entity Relationship Diagram* yang lengkap, yang merepresentasikan keterhubungan antar entitas pada sistem RPS, dapat dilihat pada Gambar 3.14



Gambar 3.14 Entity Relationship Diagram Fitur Rencana Pembelajaran Semester

3.5.15 Deployment Diagram

Deployment diagram adalah diagram UML yang menggambarkan konfigurasi fisik sistem, seperti pada Gambar 3.15.



Gambar 3.15 *Deployment Diagram* Sistem Pengelolaan RPS

Diagram *deployment* ini menunjukkan komponen *software* (*artifact*) diletakkan pada perangkat kerasnya (*node*). Terdapat tiga perangkat utama yang terlibat dalam arsitektur ini:

- Server Web (*Web Server*), merupakan server utama yang menjalankan *artifact* *PHP Runtime*. Server ini bertugas mengeksekusi logika aplikasi yang dibangun menggunakan *Laravel 8 Framework*.
- Server Database (*Database Server*), merupakan server (*node* kanan atas) yang khusus menjadi tempat penyimpanan data *MySQL DB*. Terdapat jalur koneksi yang menghubungkan server ini dengan *Server Web* agar aplikasi bisa mengambil dan menyimpan data.
- Perangkat Pengguna (*User Client*), merupakan perangkat yang dipakai oleh pengguna, seperti komputer atau laptop, yang menjalankan *Web Browser*. Melalui browser inilah pengguna bisa mengakses dan berinteraksi dengan sistem.

3.5.16 Rancangan Tampilan Antarmuka Pengguna

Rancangan tampilan antarmuka sistem ini dibangun dengan pendekatan *live prototyping*, di mana desain langsung diterapkan pada sistem. Hal ini

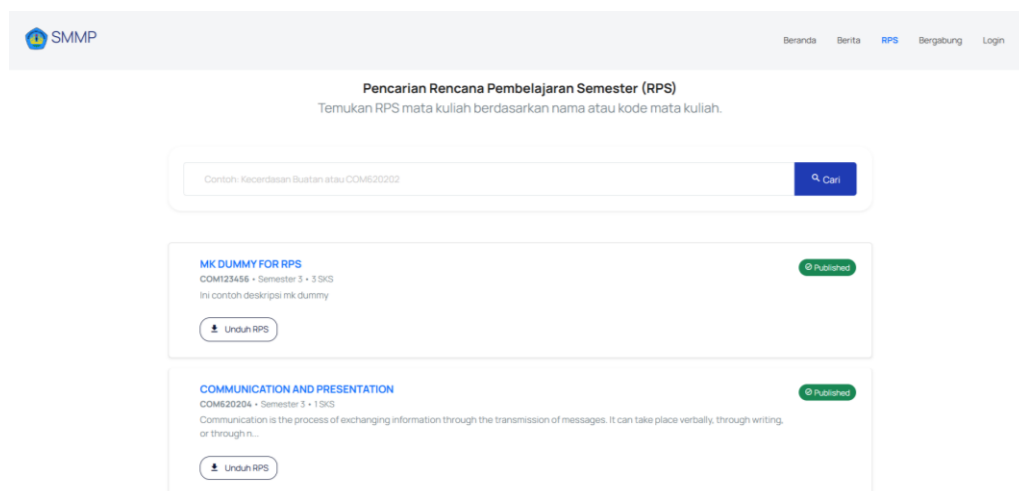
memungkinkan umpan balik *real-time* dari pengguna untuk iterasi desain yang lebih cepat, fleksibel, dan tervalidasi. Berikut adalah contoh tampilan antarmuka untuk fitur-fitur utama sistem.

1. Melihat Daftar RPS Publik (UC-001/ACT-001)

Tampilan pada Gambar 3.16 merupakan rancangan antarmuka untuk halaman daftar Rencana Pembelajaran Semester (RPS) publik. Halaman ini menyediakan fitur pencarian di bagian atas untuk menemukan RPS berdasarkan nama atau kode mata kuliah. Hasilnya kemudian ditampilkan dalam format daftar, di mana setiap RPS memuat informasi ringkas.

2. Mengunduh RPS (UC-002/ACT-002)

Seperti yang ditunjukkan pada daftar RPS di Gambar 3.16, setiap item dilengkapi dengan tombol "Unduh RPS". Tombol inilah yang digunakan oleh pengguna untuk menjalankan fungsi mengunduh dokumen RPS secara langsung.



Gambar 3.16 Rancangan Tampilan Melihat Daftar RPS Publik

3. Mengelola RPS (UC-003/ACT-003)

Tampilan pada Gambar 3.17, Gambar 3.18, dan Gambar 3.19 merupakan rancangan antarmuka untuk mengelola RPS. Proses pengelolaan Rencana Pembelajaran Semester (RPS) berawal dari halaman utama Daftar RPS pada Gambar 3.17, yang menampilkan seluruh daftar RPS dalam format tabel. Dari halaman ini, pengguna dapat melakukan dua aksi utama. Pertama, mengubah data yang sudah ada dengan menekan ikon pensil/edit yang akan mengarahkan

ke *form* edit RPS pada Gambar 3.19. Kedua, untuk membuat data RPS baru, pengguna dapat menekan tombol "Tambah RPS" yang akan membuka *form* isian seperti yang ditunjukkan pada Gambar 3.18.

Daftar RPS

[+ Tambah RPS](#)

Show entries

Search:

No	Mata Kuliah	Versi	Status	Tgl. Penyusunan	Nomor	Semester	Pengembang RPS	Action
1	LINEAR ALGEBRA	1	Rejected	28-01-2023	1	1	Dr. rer. nat. Akmal Junaidi, M.Sc	Refresh Up Down Edit Delete
2	SOFTWARE TESTING	1	Published	28-01-2023	34	3	Dr. rer. nat. Akmal Junaidi, M.Sc	Refresh
3	MACHINE LEARNING	1	Pending	28-01-2023	44	4	Dr. rer. nat. Akmal Junaidi, M.Sc	Refresh
4	NATURAL LANGUAGE PROCESSING	1	Draft	28-01-2023	54	5	Dr. rer. nat. Akmal Junaidi, M.Sc	Refresh Up Down Edit Delete

Showing 1 to 4 of 4 entries

Previous Next

Gambar 3.17 Rancangan Tampilan Mengelola RPS bagian Daftar RPS

Tambah RPS Baru ×

① Informasi Dasar

Mata Kuliah * Semester *

👤 Penanggung Jawab & Tim Dosen

Pengembang RPS * Koordinator RMK

Team Teaching

Dosen Pengampu * Dosen Anggota 1 * Dosen Anggota 2

📚 Media Pembelajaran

Software * Hardware *

📖 Pustaka & Syarat Kelulusan

Pustaka Utama *

[+ Tambah Pustaka](#)

Pustaka Pendukung

Tuliskan pustaka pendukung...

Ambang Batas Kelulusan Mahasiswa *

Ambang Batas Kelulusan Mata Kuliah *

Batal **Simpan RPS**

Gambar 3.18 Rancangan Tampilan Mengelola RPS bagian tambah RPS

Edit RPS: COMMUNICATION AND PRESENTATION ✕

Penanggung Jawab & Tim Dosen

Pengembang RPS * Koordinator RMK

Team Teaching

Dosen Pengampu * Dosen Anggota 1 * Dosen Anggota 2

Media Pembelajaran

Software * Hardware *

Pustaka & Syarat Kelulusan

Pustaka Utama *

The Communication Handbook: A Practical Guide to Effective Communication by Joanna Martin (2020)	Hapus
The Art of Communication: Achieving Interpersonal Impact and Influence by Helio Fred Garcia (2019)	Hapus
The Presentation Secrets of Steve Jobs: How to Be Insanely Great in Front of Any Audience by Carmine Gallo (2018)	Hapus
Effective Communication: A Practical Guide by Nick Harrison (2017)	Hapus
Public Speaking for Success by Dale Carnegie (2016)	Hapus
The Confident Speaker: Beat Your Nerves and Communicate at Your Best in Any Situation by Barton Schwartz (2015)	Hapus

[+ Tambah Pustaka](#)

Pustaka Pendukung

Ambang Batas Kelulusan Mahasiswa * Ambang Batas Kelulusan Mata Kuliah *

Batal **Simpan Perubahan**

Gambar 3.19 Rancangan Tampilan Mengelola RPS bagian edit RPS

RENCANA PEMBELAJARAN SEMESTER PROGRAM STUDI S1 - ILMU KOMPUTER DUMMY FAKULTAS DUMMY UNIVERSITAS DUMMY							
Identitas Mata Kuliah	NAMA MK	KODE MK	RUMPUN MATA KULLAH	BOBOT (SKS)		SEMESTER	Direvisi
	COMMUNICATION AND PRESENTATION	COM620204	Wajib	1	0	3/8	18/09/2025
Otoritas	Pengembang RPS Rizky Prabowo, M.Kom			Ketua Kelompok Keahlian		Ka PRODI Didik Kurniawan, S.Si., M.T	
Deskripsi Mata Kuliah	Communication is the process of exchanging information through the transmission of messages. It can take place verbally, through writing, or through nonverbal means such as body language or tone of voice. Communication can be one-way, in which case it is called a monologue, or two-way, in which case it is called a dialogue. Student competency in communication and presentation refers to the ability of a student to effectively communicate and present information to others in a clear, organized, and engaging manner. A student who is competent in communication and presentation is able to convey ideas, arguments, and data in a way that is easy to understand and that holds the audience's attention.						
Capaian Pembelajaran Lulusan & Capaian Pembelajaran Mata Kuliah	Capaian Pembelajaran Lulusan (CPL) PRODI						
	CPL-01	Bertakwa (A5) kepada Tuhan Yang Maha Esa, taat terhadap hukum, dan disiplin (bahan kajian) dalam kehidupan bermasyarakat dan bernegara (konteks).					
	CPL-05	Menguasai (C2) konsep teoritis dalam bidang Informatika Ilmu Komputer (bahan kajian) untuk mendesain dan mensimulasikan aplikasi teknologi multi-platform (bahan kajian) yang sesuai dengan kebutuhan industri dan masyarakat (konteks).					
	CPL-13	Having a thorough understanding and capable to apply the concepts of strategy, model, and business innovation to the enterprise system.					
	Capaian Pembelajaran Mata Kuliah (CPMK)						
	CPMK012	Mampu menjalankan kehidupan sosial masyarakat yang berdasarkan aturan dan norma hukum yang berlaku.					
	CPMK013	Mampu menerapkan disiplin dalam kehidupan bermasyarakat dan bernegara					
	CPMK135	Understanding of the fundamental concepts and principles of object-oriented programming (OOP) such as encapsulation, inheritance, and polymorphism.					
	CPMK136	Knowledge of different OOP languages such as Java, C++, and Python.					
	CPMK137	Familiarity with OOP design patterns and principles such as SOLID and DRY.					
CPMK138	Development of skills in designing, implementing, and testing object-oriented software using appropriate tools and technologies.						
CPMK139	Ability to critically evaluate and improve the maintainability, scalability, and performance of object-oriented code.						

Gambar 3.20 Rancangan Tampilan Mengelola RPS bagian *preview* RPS

4. Mengelola Aktivitas (UC-004/ACT-004)

Tampilan pada Gambar 3.21, Gambar 3.22, dan Gambar 3.23 merupakan rancangan antarmuka Mengelola Aktivitas. Pengelolaan aktivitas mingguan dilakukan seperti yang terlihat pada Gambar 3.21, di mana terdapat tabel yang menampilkan daftar aktivitas. Untuk menambahkan aktivitas baru, pengguna menekan tombol "Tambah Aktivitas" yang akan membuka *form* isian seperti pada Gambar 3.22 dan untuk mengubah data aktivitas, pengguna menekan ikon pensil/edit pada salah satu baris, yang akan menampilkan *form* edit aktivitas mingguan seperti yang ditunjukkan pada Gambar 3.23.

Detail Rencana Pembelajaran Semester
Mata Kuliah: COMMUNICATION AND PRESENTATION ← Kembali

Informasi Umum

Nomor RPS : 26

Program Studi : S1 - Ilmu Komputer Dummy

Semester : 3

Bobot SKS : 1 SKS (T1P0)

Penanggung Jawab

Pengembang RPS : Rizky Prabowo, M.Kom

Koordinator RMK : Rizky Prabowo, M.Kom

Ketua PRODI : Didik Kurniawan, S.Si., M.T

Dosen Pengampu : 1. Rizky Prabowo, M.Kom
2. Anie Rose Irawati, S.T., M.Cs

Daftar Aktivitas Mingguan + Tambah Aktivitas

Minggu	CPMK	Sub CPMK	Indikator	Materi	Asesmen & Metode	Kegiatan	Action
1	CPMK012	Mahasiswa mampu menyelesaikan pertanyaan atau tugas yang berkaitan dengan pengantar komunikasi efektif. Mahasiswa mampu menjelaskan kembali materi mengenai pengantar komunikasi efektif.	Mahasiswa mampu menyelesaikan soal atau tugas yang berkaitan dengan Pengantar Komunikasi Efektif. Mahasiswa mampu menjelaskan kembali materi mengenai Pengantar Komunikasi Efektif.	Introduction to effective communication	Asesmen: Tugas 1 Metode: Ceramah dan Diskusi	Luring: Penugasan terstruktur di kelas Daring:	

Gambar 3.21 Rancangan Tampilan Mengelola Aktivitas bagian melihat daftar aktivitas mingguan

Tambah Aktivitas Mingguan Baru ×

① Informasi Dasar

Minggu Ke- * CPMK Terkait *

- Pilih Minggu - - Pilih CPMK Terkait -

☐ Konten Pembelajaran

Deskripsi Sub CPMK * Indikator * Materi *

Pilih Sub CPMK... Tuliskan Indikator... Tuliskan Materi...

☑ Asesmen & Metode

Bentuk Asesmen * Metode Pembelajaran *

- Pilih Bentuk Asesmen - Contoh: Ceramah dan Diskusi

Kegiatan Luring Kegiatan Daring

Tuliskan kegiatan luring... Tuliskan kegiatan daring...

Batal Simpan Aktivitas

Gambar 3.22 Rancangan Tampilan Mengelola Aktivitas bagian *form* tambah aktivitas mingguan

Edit Aktivitas Minggu ke-1 ×

① Informasi Dasar

Minggu Ke- CPMK Terkait *

Minggu ke-1 CPMK012 - Mampu menjalankan kehidupan sosial masyarakat yang berdas

Minggu kegiatan tidak dapat diubah.

☐ Konten Pembelajaran

Deskripsi Sub CPMK * Indikator * Materi *

Mahasiswa mampu menyelesaikan per Mahasiswa mampu menyelesaikan so: Introduction to effective communicati

Hapus Hapus Hapus

Mahasiswa mampu menjelaskan kemb Mahasiswa mampu menjelaskan kemb +

Hapus Hapus +

☑ Asesmen & Metode

Bentuk Asesmen * Metode Pembelajaran *

- Pilih Bentuk Asesmen - Ceramah dan Diskusi

Kegiatan Luring Kegiatan Daring

Penugasan terstruktur di kelas Tuliskan kegiatan daring...

Hapus Hapus

Batal Simpan Perubahan

Gambar 3.23 Rancangan Tampilan Mengelola Aktivitas bagian *form* edit aktivitas mingguan

5. Mengelola CPL ke MK (UC-005/ACT-005)

Tampilan pada Gambar 3.24 dan Gambar 3.25 merupakan rancangan tampilan untuk mengelola CPLMK. Dalam proses menambahkan pemetaan Capaian Pembelajaran Lulusan ke Mata Kuliah (CPLMK), pengguna perlu mengisi form pada halaman Tambah CPLMK Baru Gambar 3.25 dengan memilih mata kuliah

dan CPL Prodi yang sesuai. Setelah data dikirimkan, seluruh hasil pemetaan tersebut akan ditampilkan dalam format tabel pada halaman Daftar CPLMK, seperti yang ditunjukkan pada Gambar 3.24, di mana pengguna dapat melihat dan menghapus data yang ada.

The screenshot shows a web interface titled "Daftar CPLMK". At the top left, there is a "Show" dropdown menu set to "10" and the text "entries". At the top right, there is a "Search:" input field. Below this is a table with 10 rows and 6 columns: "No", "Kode MK", "Nama MK", "Tanggal", "Kode CPL", and "Action". Each row contains data and a red "Delete" button with a trash icon. At the bottom left, it says "Showing 1 to 10 of 10 entries". At the bottom right, there are "Previous", "1", and "Next" navigation buttons.

No	Kode MK	Nama MK	Tanggal	Kode CPL	Action
1	COM123456	MK DUMMY FOR RPS	01-01-1970	CPL-03	Delete
2	COM123456	MK DUMMY FOR RPS	25-09-2025	CPL-05	Delete
3	COM620204	COMMUNICATION AND PRESENTATION	19-06-2024	CPL-01	Delete
4	COM620204	COMMUNICATION AND PRESENTATION	29-01-2023	CPL-05	Delete
5	COM620204	COMMUNICATION AND PRESENTATION	29-01-2023	CPL-13	Delete
6	COM620306	ADVANCED WEB PROGRAMMING	19-06-2024	CPL-01	Delete
7	COM620306	ADVANCED WEB PROGRAMMING	29-01-2023	CPL-02	Delete
8	COM620306	ADVANCED WEB PROGRAMMING	01-01-1970	CPL-04	Delete
9	COM620306	ADVANCED WEB PROGRAMMING	29-01-2023	CPL-07	Delete
10	COM620306	ADVANCED WEB PROGRAMMING	29-01-2023	CPL-12	Delete

Gambar 3.24 Rancangan Tampilan Mengelola CPLMK bagian melihat daftar CPLMK

Tambah CPLMK Mata Kuliah Baru

Pilih Mata Kuliah *

-

Silahkan pilih mata kuliah RPS.

Pilih CPL Prodi *

Silahkan pilih CPL Prodi.

Submit

Gambar 3.25 Rancangan Tampilan Mengelola CPLMK bagian *form* tambah CPLMK

6. Mengelola CPMK (UC-006/ACT-006)

Tampilan pada Gambar 3.26 dan Gambar 3.27 merupakan rancangan untuk mengelola Capaian Pembelajaran Mata Kuliah (CPMK). Untuk membuat CPMK baru, pengguna harus mengisi *form* pada halaman Tambah CPMK Baru seperti pada Gambar 3.27 dengan memilih CPL yang menjadi induknya serta menambahkan rincian CPMK. Setelah data dikirimkan, CPMK yang baru dibuat akan tampil pada halaman Daftar CPMK seperti pada Gambar 3.26, yang menyajikan data dalam format tabel yang dapat di *filter* berdasarkan kurikulum.

Daftar CPMK

Kurikulum

Pilih Kurikulum

Filter Reset

Show 10 entries

Search:

No	Nama CPL	Kode CPMK	Rincian CPMK
1	CPL-01	CPMK-014	Rincian MK Dummy CPMK

Gambar 3.26 Rancangan Tampilan Mengelola CPMK bagian melihat daftar CPMK

Tambah CPMK Baru

CPL *

Silahkan pilih cpl.

Judul Rincian CPMK *

Silahkan masukkan rincian CPMK.

Submit

Add Field

Gambar 3.27 Rancangan Tampilan Mengelola CPMK bagian *form* tambah CPMK

7. Mengelola Komponen (UC-007/ACT-007)

Tampilan pada Gambar 3.28 dan Gambar 3.29 merupakan rancangan tampilan untuk mengelola komponen. Untuk menambahkan komponen baru, pengguna perlu mengisikan nama komponen pada form Tambah Komponen seperti pada Gambar 3.29 lalu mengirimkannya. Setelah itu, komponen yang baru ditambahkan akan muncul dalam format tabel pada halaman Daftar Jenis seperti pada Gambar 3.28, di mana data tersebut dapat dilihat atau dihapus oleh pengguna.

Daftar Jenis

Show: entries

Search:

No	Jenis	Action
1	Hasil Praktik	Delete
2	Kualitas Presentasi	Delete
3	Ketepatan jawaban	Delete
4	Kuis ke-1	Delete
5	Kuis ke-2	Delete

Showing 1 to 5 of 5 entries

Previous Next

Gambar 3.28 Rancangan Tampilan Mengelola Komponen bagian melihat daftar komponen penilaian

Tambah Jenis

Jenis *

Submit

Gambar 3.29 Rancangan Tampilan Mengelola Komponen bagian *form* tambah komponen

8. Memvalidasi RPS (UC-008/ACT-008)

Tampilan pada Gambar 3.30 dan Gambar 3.31 merupakan rancangan untuk proses validasi RPS. Halaman utama proses ini ada pada Gambar 3.30, yang menampilkan daftar RPS yang menunggu persetujuan. Jika validator memilih untuk menolak sebuah RPS, cukup dengan menekan ikon silang (X), maka akan muncul sebuah *pop up* seperti pada Gambar 3.31, yang mewajibkan validator untuk mengisi alasan penolakan sebelum finalisasi keputusan.

Validasi RPS

Daftar RPS yang memerlukan validasi Anda untuk dapat dipublikasikan.

Show: entries

Search:

No	Kode	Mata Kuliah	Pengembang RPS	Tgl. Pengajuan	Status	Aksi Validasi
1	COM620101	LINEAR ALGEBRA	Dr. rer. nat. Akmal Junaidi, M.Sc	28 Jan 2023	Pending	
2	COM620212	SOFTWARE TESTING	Dr. rer. nat. Akmal Junaidi, M.Sc	28 Jan 2023	Pending	
3	COM620222	MACHINE LEARNING	Dr. rer. nat. Akmal Junaidi, M.Sc	28 Jan 2023	Pending	

Showing 1 to 3 of 3 entries

Previous Next

Gambar 3.30 Rancangan Tampilan Memvalidasi RPS bagian melihat daftar RPS yang perlu di validasi

Tolak RPS: LINEAR ALGEBRA

Catatan / Alasan Penolakan *

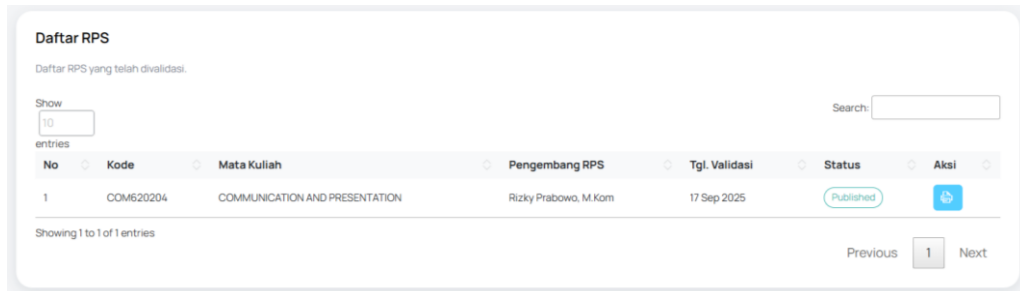
Penilaian Mingguan belum di sesuaikan dengan CPMK

Batal Kirim Penolakan

Gambar 3.31 Rancangan Tampilan Memvalidasi RPS bagian *form* tolak validasi RPS

9. Melihat Daftar RPS Tervalidasi (UC-009/ACT-009)

Tampilan pada Gambar 3.32 merupakan rancangan untuk halaman Daftar RPS yang Telah Divalidasi. Halaman ini berfungsi untuk menampilkan daftar semua RPS yang telah berhasil melewati proses validasi dan berstatus *Published*.



Daftar RPS

Daftar RPS yang telah divalidasi.

Show entries

Search:

No	Kode	Mata Kuliah	Pengembang RPS	Tgl. Validasi	Status	Aksi
1	COM620204	COMMUNICATION AND PRESENTATION	Rizky Prabowo, M.Kom	17 Sep 2025	Published	

Showing 1 to 1 of 1 entries

Previous Next

Gambar 3.32 Rancangan Tampilan Melihat Daftar RPS Tervalidasi

V. SIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan penelitian yang telah dilakukan, maka diperoleh kesimpulan sebagai berikut:

1. Fitur pengelolaan Rencana Pembelajaran Semester (RPS) berhasil dikembangkan dan diintegrasikan ke dalam Sistem Penjamin Mutu Jurusan Ilmu Komputer. Sistem ini mampu memfasilitasi seluruh alur kerja pengelolaan RPS, mulai dari penyusunan, pembaruan, penyimpanan, pemantauan, hingga proses validasi secara digital dan terpusat, sehingga mendukung pelaksanaan penjaminan mutu akademik secara lebih terstruktur.
2. Sistem telah menyediakan fitur penyajian RPS dalam format PDF yang dapat digunakan untuk keperluan administrasi dan dokumentasi. Fitur ini memungkinkan RPS dicetak dan diunduh sesuai kebutuhan, sehingga mempermudah distribusi serta arsip dokumen secara formal.

Dengan keberhasilan pengujian tersebut, sistem dinyatakan siap untuk digunakan sebagai alat bantu penjaminan mutu perkuliahan, khususnya dalam pengelolaan dokumen Rencana Pembelajaran Semester (RPS) di Jurusan Ilmu Komputer.

5.2 Saran

Berdasarkan hasil penelitian dan implementasi sistem yang telah dilakukan, terdapat saran untuk pengembangan lebih lanjut, yaitu pengembangan modul evaluasi capaian pembelajaran berbasis *Outcome-Based Education* (OBE) yang tidak hanya mengelola dokumen RPS, tetapi juga mampu melakukan perhitungan otomatis ketercapaian CPL, CPMK, serta analisis ambang batas kelulusan mata kuliah secara kuantitatif per semester.

DAFTAR PUSTAKA

- Alfarisi, I. A., Priandika, A. T., & Puspaningrum, A. S. (2023). Penerapan Framework Laravel Pada Sistem Pelayanan Kesehatan (Studi Kasus: Klinik Berkah Medical Center). *Jurnal Ilmiah Computer Science*, 2(1), 1–9. <https://doi.org/10.58602/jics.v2i1.11>
- Aliyah, Hartono, N., & Azhari Muin, A. (2024). Penggunaan User Acceptance Testing (UAT) Pada Pengujian Sistem Informasi Pengelolaan Keuangan Dan Inventaris Barang. *Switch : Jurnal Sains dan Teknologi Informasi*, 3(1), 84–100. <https://doi.org/10.62951/switch.v3i1.330>
- APTİKOM. (2024). *Panduan Kurikulum Berbasis OBE/KKNI/SKKNİ Aptikom Versi 2.0 Program Studi Sarjana Informatika/İlmu Komputer*. APTİKOM.
- Aziira, A. H., Kamil, H., & Kartika, A. D. (2023). Pembangunan Sistem Informasi Pengelolaan Rencana Pembelajaran Semester (RPS). *Jurnal Nasional Teknologi dan Sistem Informasi*, 9(2), 103–112. <https://doi.org/10.25077/teknosi.v9i2.2023.103-112>
- Azmi Ginting, N., Thamrin, I., Sadrina Zendrato, K., & Efendi Damanik, B. (2024). Perancangan Sistem Informasi Manajemen untuk Pengelolaan Kelas Stikom Tunas Bangsa Pematang Siantar Article Info. *Jurnal J-MENDIKKOM*, 1(2), 3046–5893.
- Bhatt, B., & Nandu, M. (2021). An Overview of Structural UML Diagrams. *International Research Journal of Engineering and Technology*. www.irjet.net
- Farhan Azfa, M., & Solihah, R. H. (2023). Penerapan Sistem Manajemen Mutu ISO 9001: 2015 dalam Mendukung dan Menunjang Pemasaran Produk. *Journal of Information Systems & Management*, 02(04). <https://jisma.org>

- Gupta, A. (2021). Comparative Study of Different SDLC Models. *International Journal for Research in Applied Science and Engineering Technology*, 9, 73–80. <https://doi.org/10.22214/ijraset.2021.38736>
- Hafsari, R., Arribe, E., Luthfillah Andria, M., & Miransya, V. (2024). Perancangan Sistem Informasi Penjualan Menggunakan Metode Waterfall (Studi Kasus PT. Riau Pos Intermedia). *Jurnal PROSISKO*, 11(1), 20–25.
- Hasanah, F. N., & Sri Untari, R. (2020). Buku Ajar Rekayasa Perangkat Lunak. Dalam *Buku Ajar Rekayasa Perangkat Lunak*. Umsida Press. <https://doi.org/10.21070/2020/978-623-6833-89-6>
- Heriyanto, L., & Rozi, M. F. (2023). Rancang Bangun Sistem Informasi Arsip Elektronik Menggunakan Waterfall dengan Framework Codeigniter 4. *Journal of Manufacturing and Enterprise Information System*, 1(2), 60–67. <https://doi.org/10.52330/jmeis.v1i2.188>
- Indrianto. (2023). Performance Testing On Web Information System Using Apache Jmeter and Blazemeter. *Jurnal Ilmiah Ilmu Terapan Universitas Jambi*, 7(2), 138–149. <https://doi.org/10.22437/jiituj.v7i2.28440>
- Naufal, M. D. (2023). Rancang Bangun E-RPS Prodi Teknik Informatika IIB Darmajaya Berbasis Web. *Jurnal ETNIK : Ekonomi Teknik*, 2(10), 869–879.
- Niqotaini, Z., Yulistiawan, B. S., Krisnanik, E., & Amalia, R. D. (2023). *Analisa dan Perancangan Sistem Informasi dengan Unified Modelling Language* (D. Gustian, Ed.). Indie Press.
- Noly, N. H., Usman, U., Tulus, T. J., & Iswan, I. E. (2022). Sistem Penjamin Mutu Internal dan Eksternal pada Lembaga Pendidikan Dasar. *Multiverse: Open Multidisciplinary Journal*, 1(2). <https://doi.org/https://doi.org/10.57251/multiverse.v1i2.595>
- Nur Ichsanudin, M., Yusuf Suraya, M., & Uminingsih. (2022). Pengujian Fungsional Perangkat Lunak Sistem Informasi Perpustakaan Dengan Metode Black Box Testing. *Jurnal Ilmiah Teknik dan Ilmu Komputer*, 1(2), 1–8. <https://doi.org/10.55123>

- Palinggi, O., Maesaroh, S., Permana, M. B., Huda, D. F., & Priyono, K. A. (2024). Entity-Relationship Diagram Technique in Database. *Collabits Journal*, 1(2), 102–104. <https://doi.org/10.22441/collabits.v1i2.27252>
- Permana, A. A., Agustriawan, D., Johan, M. E., Fianty, M. I., Sanjaya, S. A., Sutomo, R., Istiono, W., Pomalingo, S., Wiratama, J., Fernando, E., Faza, A., & Waworuntu, A. (2023). *Memahami Software Development Life Cycle*. Eureka Media Aksara.
- Pramudya, I. H. (2023). *Pengembangan Modul Rencana Pembelajaran Semester Pada Sistem Informasi Portal Prodi Jurusan Teknik Elektro* [Universitas Lampung]. <https://digilib.unila.ac.id/87788>
- Putri, M. P., Nadeak, E., Malahayati, M., Rahmi, N., Rini, A., Sari, D. N., Kurniati, K., Kusmiati, H., & Pratama, R. A. A. (2023). *Sistem Manajemen Basis Data Menggunakan MySQL*. CV WIDINA MEDIA UTAMA.
- Qomariah, S., & Jumiyati, J. (2025). *Digitalisasi Rencana Pembelajaran*. CV. Bo'Kampong Publishing (BKP).
- Razak, Y., Syah, D., & Hsb, A. A. (2016). Kepemimpinan, Kinerja Dosen Dalam Peningkatan Mutu Pendidikan Perguruan Tinggi. *TANZIM Jurnal Penelitian Manajemen Pendidikan*, 1(2).
- Sakdiyah, H. (2025). Hubungan antara Sistem Manajemen Mutu Pendidikan dengan Peningkatan Prestasi Akademik Siswa. *Jurnal Pengabdian Masyarakat dan Riset Pendidikan*, 4(2), 9755–9761. <https://doi.org/10.31004/jerkin.v4i2.3376>
- Sultan Adensa, A., Raihan, K., Faisal Rafi, R., Richwandi Putra, I., & Azizah, F. (2023). Pengembangan Web Dinas Perpustakaan dan Arsip Berbasis Laravel Framework Pada DPAD Kota Tangerang. *Jurnal Mahasiswa Teknik Informatika*, 7(6).
- Syafarina, G. A., & Setiawan, A. (2019). Perancangan aplikasi rencana pembelajaran semester (RPS) untuk meningkatkan pencapaian pembelajaran bagi dosen. *Technologia: Jurnal Ilmiah*, 10(4), 202–206.

Syarif A, M. R. (2024). *Perancangan Aplikasi Pengembangan Rencana Pembelajaran Semester Dengan Menggunakan Metode Prototype*. [Universitas Islam Sultan Agung].

<https://repository.perpus.unissula.ac.id/id/eprint/68018>

ZareRavasan, A., & Krčál, M. (2021). A Systematic Literature Review on 30 Years of Empirical Research on Information Systems Business Value. *Journal of Global Information Management*, 29(6).

<https://doi.org/10.4018/JGIM.288894>