

# **RENCANA PEMBELAJARAN SEMESTER SOFTWARE ENGINEERING**



**Disusun Oleh :**

**SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER ROSMA**

**PROGRAM STUDI SISTEM INFORMASI**

**2020**

**DIAGRAM ANALISIS HASIL PEMBELAJARAN MATAKULIAH SOFTWARE  
ENGINEERING**



**SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER ROSMA  
PROGRAM STUDI SISTEM INFORMASI**

**RENCANA PEMBELAJARAN SEMESTER**

<b>MATA KULIAH</b>	<b>KODE</b>	<b>Rumpun Mata Kuliah</b>	<b>BOBOT(sks)</b>	<b>SEMESTER</b>	<b>Tgl Penyusunan</b>
Software Engineering		Sistem Informasi	4		
<b>OTORISASI</b>	<b>Dosen Pengembang RPS</b>	<b>Koordinator RMK</b>	<b>Kepala Program Studi</b>		
			Lila Sertiyani, S.T, M.Kom		
<b>Capaian Pembelajaran (CP)</b>	<b>CPL-PRODI</b>				
	P1	Menguasai konsep teoritis bidang pengetahuan Sistem Informasi secara umum dan konsep teoritis bagian khusus dalam bidang pengetahuan tersebut secara mendalam, serta mampu memformulasikan penyelesaian masalah prosedural			
	P2	Menguasai konsep teoritis yang mengkaji, menerapkan dan mengembangkan serta mampu memformulasikan dan mampu mengambil keputusan yang tepat dalam penyelesaian masalah			
	P3	Mempunyai pengetahuan dalam penyusunan algoritma pemrograman yang efektif dan efisien serta dapat merancang, membangun dan mengelola aplikasi sistem informasi secara tepat dan akurat untuk pendukung pengambilan keputusan			
	P4	Memiliki pengetahuan sesuai dengan capaian pembelajaran program studi S1 sistem Informasi			
	KU1	Mampu menerapkan pemikiran logis, kritis, sistematis dan inovatif dalam konteks pengembangan atau implementasi ilmu pengetahuan dan teknologi yang memperhatikan dan menerapkan nilai humaniora yang sesuai dengan bidang keahliannya			
	KU11	Memiliki kemampuan untuk menjadi tenaga profesional untuk pengelolaan basis data, rekayasa perangkat lunak, jaringan komputer, komputer grafis, dan aplikasi multimedia serta memiliki kemampuan menulis laporan penelitian dengan baik serta mengelola proyek sistem informasi, mempresentasikan karya tersebut			
	KU12	Memiliki kecakapan hidup sesuai capaian pembelajaran program studi Sistem Informasi			
	KK1	Mampu mendesain arsitektur SI/TI			
	KK4	Mampu menjadi profesional TI yang bertindak sebagai mediator antara klien dan tim teknis			

	KK5	Mampu mengintegrasikan business requirement kedalam teknologi dan memastikan kelancaran operasional bisnis
	KK6	Mampu menggunakan pengetahuan bisnis dan teknis untuk menganalisis proses bisnis, sistem komputer dan infrastruktur
	KK7	Mampu mengembangkan strategi yang efektif dan efisien untuk memenuhi kebutuhan sehari – hari organisasi atau perusahaan
	KK9	Mampu bekerja untuk memecahkan masalah yang berpusat pada IT dalam suatu organisasi.
	KK15	Memiliki ketrampilan pemecahan masalah dan manajemen proyek serta mampu membangun tim dan ketrampilan analitis.
	<b>CP-MK</b>	
	M1	Introduction to Software Engineering
	M2	System Dependability and Security
	M3	Advanced Software Engineering
	M4	Software Management
<b>Deskripsi Mata Kuliah</b>	Mata kuliah ini mengajarkan kepada mahasiswa tentang Software Engineering	
<b>Materi Pembelajaran / Pokok Bahasan</b>	<b>SUB-CP-MK</b>	
		<ol style="list-style-type: none"> <li>1. Introduction Software Engineering and Software Process[M1]</li> <li>2. Agile Software Development[M1]</li> <li>3. Requirement engineering[M1]</li> <li>4. System Modeling[M1]</li> <li>5. Architectural Design[M1]</li> <li>6. Design and Implementation[M1]</li> <li>7. Software Testing [M1]</li> <li>8. Software Evolution[M1]</li> <li>9. Dependable system[M2]</li> <li>10. Reliability Engineering[M2]</li> <li>11. Safety Engineering[M2]</li> <li>12. Security Engineering[M2]</li> <li>13. Resilience Engineering[M2]</li> <li>14. Software Reuse[M3]</li> <li>15. Component-based software engineering[M3]</li> <li>16. Distributed software engineering[M3]</li> <li>17. Service-Oriented Software Engineering[M3]</li> <li>18. System engineering [M3]</li> </ol>

	19. System of system[M3] 20. Real-time software engineering[M3] 21. Project Management[M4] 22. Project Planning [M4] 23. Quality Management[M4] 24. Configuration Management[M4]					
<b>Pustaka</b>	<b>Utama :</b>					
	Sommerville, Ian(2016). <a href="#">Software Engineering</a> Tenth Edition . United of America : Perason					
	<b>Pendukung :</b>					
<b>Media Pembelajaran</b>	<b>Perangkat Lunak :</b>			<b>Perangkat Keras :</b>		
	Microsoft Word			Proyektor , Papan Tulis ,Spidol, Penghapus		
<b>Tema Teaching</b>						
<b>Matakuliah Syarat</b>						
<b>Mg Ke-</b>	<b>Sub-CP-MK (sbg kemampuan akhir yang diharapkan)</b>	<b>Indikator</b>	<b>Kriteria &amp; Bentuk Penilaian</b>	<b>Metode Pembelajaran [Estimasi Waktu]</b>	<b>Materi Pembelajaran [Pustaka]</b>	<b>Bobot Penilaian[%]</b>
1	Introduction and Software Process[C2, A2, P1]	Ketepatan dalam menjelaskan : <ul style="list-style-type: none"> <li>Introduction and Software Process</li> </ul>	<b>Kriteria :</b> Ketepatan dan penguasaan <b>Bentuk non-test :</b> Tanya jawab	<ul style="list-style-type: none"> <li>Kuliah dan diskusi [TM 3 x 50"]</li> </ul>	<ul style="list-style-type: none"> <li>Professional software development</li> <li>Software engineering ethics</li> <li>Software process models</li> <li>Process activities</li> <li>Coping with change</li> <li>Process improvement</li> </ul>	5%
2	<ul style="list-style-type: none"> <li>Agile Software Development</li> <li>Requirements Engineering[C2, A2, P1]</li> </ul>	Ketepatan dalam menjelaskan : <ul style="list-style-type: none"> <li>Agile Software Development</li> <li>Requirements Engineering</li> </ul>	<b>Kriteria :</b> Ketepatan dan penguasaan <b>Bentuk non-test :</b> Tanya jawab	<ul style="list-style-type: none"> <li>Kuliah dan diskusi [TM 3 x 50"]</li> </ul>	<ul style="list-style-type: none"> <li>Agile methods</li> <li>Agile development techniques</li> <li>Agile project management</li> <li>Scaling agile methods</li> </ul>	

					<ul style="list-style-type: none"> <li>▪ Functional and non-functional requirements</li> <li>▪ Requirement elicitation</li> <li>▪ Requirement specification</li> <li>▪ Requirement validation</li> <li>▪ Requirement change</li> </ul>	
3	<ul style="list-style-type: none"> <li>▪ System Modeling</li> <li>▪ Architectural Design [C2, A2, P1]</li> </ul>	Ketepatan dalam menjelaskan : <ul style="list-style-type: none"> <li>▪ System Modeling</li> <li>▪ Architectural Design</li> </ul>	<b>Kriteria :</b> Ketepatan dan penguasaan <b>Bentuk non-test :</b> Tanya jawab	<ul style="list-style-type: none"> <li>▪ Kuliah dan diskusi [TM 3 x 50"]</li> </ul>	<ul style="list-style-type: none"> <li>▪ Context models</li> <li>▪ Interaction models</li> <li>▪ Structural model</li> <li>▪ Behavioral models</li> <li>▪ Model-driven architecture</li> <li>▪ Architectural design decision</li> <li>▪ Architectural views</li> <li>▪ Architectural patterns</li> <li>▪ Application architectures</li> </ul>	5%
4	Design and Implementation Software Testing [C2, A2, P1]	Ketepatan dalam menjelaskan : <ul style="list-style-type: none"> <li>▪ Design and Implementation</li> <li>▪ Software Testing</li> </ul>	<b>Kriteria :</b> Ketepatan dan penguasaan <b>Bentuk non-test :</b> Tanya jawab	<ul style="list-style-type: none"> <li>▪ Kuliah dan diskusi [TM 3 x 50"]</li> </ul>	<ul style="list-style-type: none"> <li>▪ Object – Oriented Design Using the UML</li> <li>▪ Design patterns</li> <li>▪ Implementation issue</li> <li>▪ Open-source development</li> <li>▪ Development testing</li> <li>▪ Test-driven development</li> <li>▪ Release testing</li> <li>▪ User testing</li> </ul>	5%

5	Software Evolution [C2, A2, P1]	Ketepatan dalam menjelaskan : ▪ Software Evolution	<b>Kriteria :</b> Ketepatan dan penguasaan <b>Bentuk non-test</b> : Tanya jawab	▪ Kuliah dan diskusi [TM 3 x 50”]	▪ Evolution Process ▪ Legacy System ▪ Software Maintenance ▪ Review materi	10%
6	▪ Dependable system ▪ Reliability Engineering [C2, A2, P1]	Ketepatan dalam menjelaskan : ▪ Dependable system ▪ Reliability Engineering	<b>Kriteria :</b> Ketepatan dan penguasaan <b>Bentuk non-test</b> : Tanya jawab	▪ Kuliah dan diskusi [TM 3 x 50”]	▪ Dependability properties ▪ Sociotechnical system ▪ Redudancy and diversity ▪ Dependable process ▪ Formal methods and dependability ▪ Availability and reliability ▪ Reliability requirements ▪ Fault-tolerant architectures ▪ Programming for reliability ▪ Reliability measurement	10%
7	Safety Engineering [C2, A2, P1]	Ketepatan dalam menjelaskan : Safety Engineering	<b>Kriteria :</b> Ketepatan dan penguasaan <b>Bentuk non-test</b> : Tanya jawab	▪ Kuliah dan diskusi [TM 3 x 50”]	▪ Safety – critical system ▪ Safety requirement ▪ Safety engineering process ▪ Safety cases ▪ Review Materi	5%
<b>UJIAN TENGAH SEMESTER</b>						

9	<ul style="list-style-type: none"> <li>▪ Security engineering</li> <li>▪ Resilience Engineering [C2, A2, P1]</li> </ul>	<p>Ketepatan dan penguasaan dalam :</p> <ul style="list-style-type: none"> <li>▪ Security engineering</li> <li>▪ Resilience Engineering</li> </ul>	<p><b>Kriteria :</b> Ketepatan dan penguasaan <b>Bentuk Non Test :</b> Tanya jawab</p>	<ul style="list-style-type: none"> <li>▪ Kuliah dan diskusi [TM 3 x 50”]</li> </ul>	<ul style="list-style-type: none"> <li>▪ Security and dependability</li> <li>▪ Security and organization</li> <li>▪ Security and requirement</li> <li>▪ Secure system design</li> <li>▪ Security testing and assurance</li> <li>▪ Cybersecurity</li> <li>▪ Sociotechnical resilience</li> <li>▪ Resilient system design</li> </ul>	5%
10	<ul style="list-style-type: none"> <li>▪ Software reuse</li> <li>▪ Component-based software engineering [C2, A2, P1]</li> </ul>	<p>Ketepatan dan penguasaan dalam :</p> <ul style="list-style-type: none"> <li>▪ Software reuse</li> <li>▪ Component-based software engineering</li> </ul>	<p><b>Kriteria :</b> Ketepatan dan penguasaan <b>Bentuk Non Test :</b> Tanya jawab</p>	<ul style="list-style-type: none"> <li>▪ Kuliah dan diskusi [TM 3 x 50”]</li> </ul>	<ul style="list-style-type: none"> <li>▪ The reuse lanscape</li> <li>▪ Application frameworks</li> <li>▪ Software product lines</li> <li>▪ Application system reuse</li> <li>▪ Components and component models</li> <li>▪ CBSE processes</li> <li>▪ Component composition</li> </ul>	15%
11	<ul style="list-style-type: none"> <li>▪ Distributed software engineering</li> <li>▪ Service-Oriented Software Engineering [C2, A2, P1]</li> </ul>	<p>Ketepatan dan penguasaan dalam memahami :</p> <ul style="list-style-type: none"> <li>▪ Distributed software engineering</li> <li>▪ Service-Oriented Software Engineering</li> </ul>	<p><b>Kriteria :</b> Ketepatan dan penguasaan <b>Bentuk Non Test :</b> Tanya jawab</p>	<ul style="list-style-type: none"> <li>▪ Kuliah dan diskusi [TM 3 x 50”]</li> </ul>	<ul style="list-style-type: none"> <li>▪ Distributed system</li> <li>▪ Client-server computing</li> <li>▪ Architectural patterns for distributed system</li> <li>▪ Software as a service</li> <li>▪ Service-oriented architecture</li> <li>▪ RESTful Service</li> <li>▪ Service Engineering</li> </ul>	5%



					<ul style="list-style-type: none"> <li>▪ Service Composition</li> </ul>	
12	<ul style="list-style-type: none"> <li>▪ System Engineering</li> <li>▪ System of System [C2, A2, P1]</li> </ul>	<p>Ketepatan dan penguasaan dalam memahami :</p> <ul style="list-style-type: none"> <li>▪ System Engineering</li> <li>▪ System of System</li> </ul>	<p><b>Kriteria :</b> Ketepatan dan penguasaan <b>Bentuk Non Test :</b> Tanya jawab</p>	<ul style="list-style-type: none"> <li>▪ Kuliah dan diskusi [TM 3 x 50”]</li> </ul>	<ul style="list-style-type: none"> <li>▪ Sociotechnical system</li> <li>▪ Conceptual design</li> <li>▪ System procurement</li> <li>▪ System development</li> <li>▪ System operation and evolution</li> <li>▪ System complexity</li> <li>▪ System of system classification</li> <li>▪ Reductionism and complex system</li> <li>▪ System of system engineering</li> <li>▪ System of system architecture</li> </ul>	5%
13	<p>Real-time software engineering [C2, A2, P1]</p>	<p>Ketepatan dan penguasaan dalam memahami : Real-time software engineering</p>	<p><b>Kriteria :</b> Ketepatan dan penguasaan <b>Bentuk Non Test :</b> Tanya jawab</p>	<ul style="list-style-type: none"> <li>▪ Kuliah dan diskusi [TM 3 x 50”]</li> </ul>	<ul style="list-style-type: none"> <li>▪ Embedded system design</li> <li>▪ Architectural patterns for real – time software</li> <li>▪ Timing analysis</li> <li>▪ Real-time operating system</li> <li>▪ Review materi</li> </ul>	5%
14	<ul style="list-style-type: none"> <li>▪ Project Management</li> <li>▪ Project planning [C2, A2, P1]</li> </ul>	<p>Ketepatan dan penguasaan dalam menjelaskan :</p> <ul style="list-style-type: none"> <li>▪ Project Management</li> <li>▪ Project planning</li> </ul>	<p><b>Kriteria :</b> Ketepatan dan penguasaan <b>Bentuk Non Test :</b> Tanya jawab</p>	<ul style="list-style-type: none"> <li>▪ Kuliah dan diskusi [TM 3 x 50”]</li> </ul>	<ul style="list-style-type: none"> <li>▪ Risk management</li> <li>▪ Managing people</li> <li>▪ Teamwork</li> <li>▪ Software pricing</li> <li>▪ Plan-driven development</li> <li>▪ Project scheduling</li> <li>▪ Agile planning</li> <li>▪ Estimation techniques</li> </ul>	5%

					<ul style="list-style-type: none"> <li>▪ COCOMO Cost Modeling</li> </ul>	
15	<ul style="list-style-type: none"> <li>▪ Quality Management</li> <li>▪ Configuration Management [C2, A2, P1]</li> </ul>	<p>Ketepatan dan penguasaan dalam menjelaskan :</p> <ul style="list-style-type: none"> <li>▪ Quality Management</li> <li>▪ Configuration Management</li> </ul>	<p><b>Kriteria :</b> Ketepatan dan penguasaan</p> <p><b>Bentuk Non Test :</b> Tanya jawab</p>	<ul style="list-style-type: none"> <li>▪ Kuliah dan diskusi [TM 3 x 50"]</li> </ul>	<ul style="list-style-type: none"> <li>▪ Software quality</li> <li>▪ Software standards</li> <li>▪ Reviews and inspections</li> <li>▪ Quality management and agile development</li> <li>▪ Software measurement</li> <li>▪ Version management</li> <li>▪ System building</li> <li>▪ Change management</li> <li>▪ Release management</li> </ul>	5%
<b>UJIAN AKHIR SEMESTER</b>						

Catatan :

- 1) TM : Tatap muka , BT : Belajar Terstruktur , BM : Belajar Mandiri
- 2) [TM 2x50"] dibaca : 2 sks x 50 menit = 150 menit
- 3) [BT+BM: (1x50")]dibaca : belajar terstuktut dan mandiri 1x 50 menit
- 4) Mahasiswa memahami Introduction and Software Process [C2,A2,P2] : menunjukkan bahwa sub-CPMK ini mengandung kemampuan ranah taksonomi kognitif level 2(memahami), ranah taksonomi afektif level 2 (menanggapi) dan ranah taksonomi psikomotorik level 1(meniru)



**SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN  
KOMPUTER ROSMA**  
**PROGRAM STUDI SISTEM INFORMASI**

**RENCANA TUGAS MAHASISWA**

<b>MATA KULIAH</b>	<b>Software Engineering</b>				
<b>KODE</b>		<b>SKS</b>	<b>4</b>	<b>SEMESTER</b>	
<b>DOSEN PENGAMPU</b>					
<b>BENTUK TUGAS</b>					
<b>JUDUL TUGAS</b>					
<b>SUB CAPAIN PEMBELAJARAN MATA KULIAH</b>					
<b>DISKRIPSI TUGAS</b>					
<b>METODE Pengerjaan Tugas</b>	1.				
<b>BENTUK DAN FORMAT LUARAN</b>	a. Obyek garapan : b. Bentuk Luaran :				
<b>INDIKATOR , KRITERIA DAN BOBOT PENILAIAN</b>	a. Hasil analisis (%) b. Hasil dokumentasi (%)				
<b>JADWAL PELAKSANAAN</b>					
<b>LAIN - LAIN</b>	Tugas wajib				
<b>DAFTAR RUJUKAN</b>					



**SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN  
KOMPUTER ROSMA  
PROGRAM STUDI SISTEM INFORMASI**

**RENCANA UJIAN TENGAH SEMESTER MAHASISWA**

**MATA KULIAH** Software Engineering

**KODE** SKS 4 SEMESTER

**DOSEN  
PENGAMPU**

**BENTUK UJIAN TENGAH SEMESTER**

**JUDUL UJIAN TENGAH SEMESTER**

**DISKRIPSI UJIAN TENGAH SEMESTER**

**METODE UJIAN TENGAH SEMESTER**

**BENTUK DAN FORMAT LUARAN**

- a. Obyek garapan :
- b. Bentuk Luaran :

**INDIKATOR , KRITERIA DAN BOBOT PENILAIAN**

Hasil analisis (100%)

**JADWAL PELAKSANAAN**

**LAIN - LAIN**

**DAFTAR RUJUKAN**



**SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN  
KOMPUTER ROSMA  
PROGRAM STUDI SISTEM INFORMASI**

**RENCANA UJIAN AKHIR SEMESTER MAHASISWA**

**MATA KULIAH** Software Engineering

**KODE** **SKS** **4** **SEMESTER**

**DOSEN  
PENGAMPU**

**BENTUK UJIAN AKHIR SEMESTER**

**JUDUL UJIAN AKHIR SEMESTER**

**DISKRIPSI UJIAN AKHIR SEMESTER**

**METODE UJIAN AKHIR SEMESTER**

1.

**BENTUK DAN FORMAT LUARAN**

a. Obyek garapan :

b. Bentuk Luaran :

**INDIKATOR , KRITERIA DAN BOBOT PENILAIAN**

**JADWAL PELAKSANAAN**

**LAIN - LAIN**

**DAFTAR RUJUKAN**