

	Universitas Negeri Surabaya Fakultas Vokasi Program Studi D4 Manajemen Informatika					Kode Dokumen																																																																																							
	RENCANA PEMBELAJARAN SEMESTER																																																																																												
MATA KULIAH (MK)	KODE	Rumpun MK	BOBOT (sks)			SEMESTER	Tgl Penyusunan																																																																																						
Prak. Pemrograman Berorientasi Objek	5730102211		T=2	P=0	ECTS=3.18	3	12 Agustus 2024																																																																																						
OTORISASI	Pengembang RPS		Koordinator RMK			Koordinator Program Studi																																																																																							
	I Gde Agung Sri Sidhimantra S.Kom, M.Kom		I Gde Agung Sri Sidhimantra S.Kom, M.Kom			DODIK ARWIN DERMAWAN																																																																																							
Model Pembelajaran	Project Based Learning																																																																																												
Capaian Pembelajaran (CP)	CPL-PRODI yang dibebankan pada MK																																																																																												
	CPL-7	Dapat mengkaji dan memanfaatkan ilmu pengetahuan dan teknologi dalam memberikan solusi serta bertanggung jawab terhadap hasil kerja kelompok maupun mandiri yang bermutu dan terukur dengan sifat dan konteks yang sesuai dengan bidang keahlian terapananya.																																																																																											
	CPL-9	Mampu memilih sumberdaya dan memanfaatkan perangkat atau teknologi modern untuk merancang dan mewujudkan bidang rekayasa perangkat lunak sesuai dengan kebutuhan fungsional dan non fungsional secara spesifik dengan pertimbangan yang tepat terhadap masalah kultural, sosial, dan lingkungan dengan mengacu kepada metode dan standar industri.																																																																																											
	CPL-11	Mampu meningkatkan kinerja atau mutu suatu proses dalam perangkat lunak melalui pengujian, pengukuran obyek kerja, analisis dan interpretasi data sesuai prosedur dan standar.																																																																																											
	Capaian Pembelajaran Mata Kuliah (CPMK)																																																																																												
	CPMK - 1	Mahasiswa dapat memahami konsep dasar pemrograman berorientasi objek seperti class, objects, abstraction, encapsulation, inheritance dan polymorphism dalam pembuatan program																																																																																											
	CPMK - 2	Mahasiswa dapat merancang sebuah aplikasi dan melakukan debugging pengujian program dengan menggunakan pemrograman berorientasi objek																																																																																											
	CPMK - 3	Mahasiswa dapat merancang dan menangani user interface dalam sebuah aplikasi																																																																																											
	Matrik CPL - CPMK																																																																																												
		<table border="1"> <tr> <th>CPMK</th> <th>CPL-7</th> <th>CPL-9</th> <th>CPL-11</th> </tr> <tr> <td>CPMK-1</td> <td>✓</td> <td></td> <td></td> </tr> <tr> <td>CPMK-2</td> <td></td> <td>✓</td> <td></td> </tr> <tr> <td>CPMK-3</td> <td></td> <td></td> <td>✓</td> </tr> </table>						CPMK	CPL-7	CPL-9	CPL-11	CPMK-1	✓			CPMK-2		✓		CPMK-3			✓																																																																						
	CPMK	CPL-7	CPL-9	CPL-11																																																																																									
	CPMK-1	✓																																																																																											
	CPMK-2		✓																																																																																										
	CPMK-3			✓																																																																																									
	Matrik CPMK pada Kemampuan akhir tiap tahapan belajar (Sub-CPMK)																																																																																												
	<table border="1"> <tr> <th rowspan="2">CPMK</th> <th colspan="16">Minggu Ke</th> </tr> <tr> <th>1</th> <th>2</th> <th>3</th> <th>4</th> <th>5</th> <th>6</th> <th>7</th> <th>8</th> <th>9</th> <th>10</th> <th>11</th> <th>12</th> <th>13</th> <th>14</th> <th>15</th> <th>16</th> </tr> <tr> <td>CPMK-1</td> <td>✓</td> <td>✓</td> <td>✓</td> <td>✓</td> <td>✓</td> <td>✓</td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td>CPMK-2</td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td>✓</td> <td>✓</td> <td></td> <td></td> <td></td> <td>✓</td> <td>✓</td> <td>✓</td> <td>✓</td> <td></td> <td></td> </tr> <tr> <td>CPMK-3</td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td>✓</td> <td>✓</td> <td>✓</td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> </table>						CPMK	Minggu Ke																1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	CPMK-1	✓	✓	✓	✓	✓	✓												CPMK-2							✓	✓				✓	✓	✓	✓			CPMK-3									✓	✓	✓						
CPMK	Minggu Ke																																																																																												
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16																																																																													
CPMK-1	✓	✓	✓	✓	✓	✓																																																																																							
CPMK-2							✓	✓				✓	✓	✓	✓																																																																														
CPMK-3									✓	✓	✓																																																																																		
Deskripsi Singkat MK	Mata kuliah ini mengajarkan konsep dan teknik pemrograman berorientasi objek. Dalam mengajarkan konsep PBO tersebut, bahasa pemrograman Python akan digunakan karena Python menggunakan konsep objek dan class dalam pembuatan program. Materi ajar meliputi pengantar konsep OOP, Python Visual Studio Code, objek, class, method, constructor, I/O, pewarisan sifat (inheritance), enkapsulasi (encapsulation), polymorphism (polymorphism), overloading, UML, file, Swing, applet dan exception.																																																																																												
Pustaka	Utama :																																																																																												
		1. Harold, E.R. 2014. Python Network Programming, 4th edition. O'Reilly 2. "Think Python: How to Think Like a Computer Scientist" Allen B. Downey, 2nd Edition (2015)																																																																																											
	Pendukung :																																																																																												

		1. Jaworski, J. 1998. Python 2 Unleashed. Sams Publishing 2. Modul Teori dan Student Activity 3. Holmes, B.J., Joice D.T. 2001. Object-Oriented Programming With Python, second edition. 4. Bakker, J. 2005. Beginning Python Objects From Concepts to Code, second edition, Apress. 5. Classes: Dokumentasi resmi mengenai class, object, dan abstraction. 6. Inheritance: Dokumentasi lengkap tentang inheritance, polymorphism, dan konsep terkait. 7. Errors and Exceptions: Dokumentasi resmi untuk debugging dan exception handling.					
Dosen Pengampu		Faris Abdi El Hakim, S.Kom., M.Tr.Kom. I Gde Agung Sri Sidhimantra, S.Kom., M.Kom. Rosita, S.Pd., M.Pd. Binti Kholifah, S.Kom., M.Tr.Kom. Dimas Novian Aditia Syahputra, S.Tr.T., M.Tr.T. M Adamu Islam Mashuri, S.Tr.T., M.Tr.Kom. Moch Deny Pratama, S.Tr.Kom., M.Kom.					
Mg Ke-	Kemampuan akhir tiap tahapan belajar (Sub-CPMK)	Penilaian		Bentuk Pembelajaran, Metode Pembelajaran, Penugasan Mahasiswa, [Estimasi Waktu]		Materi Pembelajaran [Pustaka]	Bobot Penilaian (%)
		Indikator	Kriteria & Bentuk	Luring (offline)	Daring (online)		
(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
1	Pengenalan OOP dan penerapan konsep class dan objek dalam pemrograman python	1. Memahami konsep dasar Pemrograman Berorientasi Objek (OOP) 2.1.2. Mengetahui sintaks dasar Python dalam konteks OOP 3. Mampu membuat kelas dan objek sederhana dalam Python.	Kriteria: Keaktifan mahasiswa dalam bertanya Bentuk Penilaian : Penilaian Praktikum, Praktik / Unjuk Kerja	Kuliah, Tugas Praktikum : Membuat kelas dan objek Laptop, membuat pola dasar dengan turtle menggunakan konsep class dan objek [PB:1x(2x50")]	Asynchronous : Laporan kegiatan praktikum [PT,KM: (1,1)x(2x60")]	Materi: Kelas dan Objek Pustaka: <i>Classes: Dokumentasi resmi mengenai class, object, dan abstraction.</i>	5%
2	Menerapkan konsep konstruktor dan enkapsulasi dalam pemrograman python	1. Memahami konsep konstruktor dalam OOP 2. Memahami dan menerapkan prinsip enkapsulasi dalam Python 3. Mampu mengimplementasikan akses modifier (public, private) untuk mengatur aksesibilitas atribut dan metode 4. Mahasiswa dapat membuat objek bergerak sederhana di Pygame dengan prinsip OOP	Kriteria: 1. Keaktifan mahasiswa dalam bertanya 2. Kemampuan menyelesaikan tugas Bentuk Penilaian : Penilaian Praktikum	Kuliah, Tugas Praktikum: Membuat program Smartphone menggunakan konsep konstruktor dan enkapsulasi [PB:1x(2x50")]	Asynchronous : Laporan kegiatan praktikum [PT,KM: (1,1)x(2x60")]	Materi: Enkapsulasi Pustaka: <i>Bakker, J. 2005. Beginning Python Objects From Concepts to Code, second edition, Apress.</i> Materi: Abstraksi Pustaka: <i>Classes: Dokumentasi resmi mengenai class, object, dan abstraction.</i>	5%
3	Menerapkan konsep metode dan atribut kelas dalam pemrograman python	1. Memahami konsep metode dan atribut kelas (class methods dan class attributes) 2. Membedakan antara metode/atribut instance dan metode/atribut kelas 3. Mengimplementasikan metode dan atribut kelas dalam program Python	Kriteria: 1. Keaktifan mahasiswa dalam bertanya 2. Kemampuan menyelesaikan tugas Bentuk Penilaian : Penilaian Praktikum	Kuliah, Tugas Praktikum: Membuat program Laptop menggunakan konsep metode dan atribut [PB:1x(2x50")]	Asynchronous : Laporan kegiatan praktikum [PT,KM: (1,1)x(2x60")]	Materi: Metode dan atribut Pustaka: <i>Bakker, J. 2005. Beginning Python Objects From Concepts to Code, second edition, Apress.</i>	5%
4	Menerapkan konsep Inheritance dalam pemrograman python	1. Memahami konsep pewarisan (inheritance) dalam Pemrograman Berorientasi Objek (OOP) 2. Menerapkan pewarisan dalam pembuatan kelas di Python 3. Memahami peran superclass dan subclass serta cara mengakses anggota dari superclass	Kriteria: 1. Keaktifan mahasiswa dalam bertanya 2. Kemampuan menyelesaikan tugas Bentuk Penilaian : Penilaian Praktikum	Kuliah, Tugas Praktikum: Membuat program perangkat elektronik menggunakan konsep inheritance [PB:1x(2x50")]	Asynchronous : Laporan kegiatan praktikum [PT,KM: (1,1)x(2x60")]	Materi: Inheritance Pustaka: <i>Inheritance: Dokumentasi lengkap tentang inheritance, polymorphism, dan konsep terkait.</i>	5%

5	Menerapkan konsep Inheritance dalam pemrograman python	1. Memahami konsep dasar polimorfisme dalam Pemrograman Berorientasi Objek (OOP) 2. Menerapkan polimorfisme dalam metode dan kelas yang berbeda 3. Mengimplementasikan polimorfisme dengan teknik overriding dan penggunaan metode umum pada kelas turunan	Kriteria: 1. Keaktifan mahasiswa dalam bertanya 2. Kemampuan menyelesaikan tugas Bentuk Penilaian : Penilaian Praktikum	Kuliah, Tugas Praktikum: Membuat program hewan dengan konsep polimorfisme [PB:1x(2x50")]	Asynchronous : Laporan kegiatan praktikum [PT,KM: (1,1)x(2x60")]	Materi: Inheritance Pustaka: <i>Inheritance: Dokumentasi lengkap tentang inheritance, polymorphism, dan konsep terkait.</i>	5%
6	Menerapkan konsep abstraksi dalam pemrograman python	1. Memahami konsep dasar abstraksi dalam Pemrograman Berorientasi Objek (OOP) 2. Menerapkan abstraksi dalam pembuatan kelas abstrak menggunakan modul abc pada Python 3. Menerapkan metode abstrak dalam kelas turunan	Kriteria: 1. Keaktifan mahasiswa dalam bertanya 2. Kemampuan menyelesaikan tugas Bentuk Penilaian : Penilaian Praktikum	Kuliah, Tugas Praktikum: Membuat program kendaraan menggunakan konsep abstraksi [PB:1x(2x50")]	Asynchronous : Laporan kegiatan praktikum [PT,KM: (1,1)x(2x60")]	Materi: Abstraksi Pustaka: <i>Classes: Dokumentasi resmi mengenai class, object, dan abstraction.</i>	5%
7	Menerapkan exception handling dalam pemrograman python	1. Memahami konsep error dan exception dalam Python 2. Menggunakan blok try-except untuk menangani exception 3. Menangani berbagai jenis exception dan menggunakan blok finally untuk membersihkan sumber daya 4. Membuat exception sendiri (custom exception) dalam Python	Kriteria: 1. Keaktifan mahasiswa dalam bertanya 2. Kemampuan menyelesaikan tugas Bentuk Penilaian : Penilaian Praktikum	Kuliah, Tugas Praktikum: Membuat program kalkulator dengan menerapkan exception handling [PB:1x(2x50")]	Asynchronous : Laporan kegiatan praktikum [PT,KM: (1,1)x(2x60")]	Materi: Exception handling Pustaka: <i>Errors and Exceptions: Dokumentasi resmi untuk debugging dan exception handling.</i>	5%
8	Evaluasi Tengah Semester / Ujian Tengah Semester	Membuat exception sendiri (custom exception) dalam Python	Kriteria: Kemampuan menyelesaikan tugas Bentuk Penilaian : Tes	Evaluasi Tengah Semester / Ujian Tengah Semester : Quiz [PB:1x(2x50")]	- [PT,KM: (1,1)x(2x60")]	Materi: Konsep OOP Pustaka: <i>Holmes, B.J., Joice D.T. 2001. Object-Oriented Programming With Python, second edition.</i>	0%
9	Menerapkan konsep OOP dalam pembuatan game sederhana python	1. Memahami konsep dasar OOP (class, object, inheritance, encapsulation) dalam pengembangan game dengan PyGame 2. Mengaplikasikan konsep OOP dalam pembuatan game sederhana menggunakan PyGame 3. Membuat program game berbasis PyGame dengan menggunakan class, objek, dan prinsip-prinsip OOP	Kriteria: 1. Keaktifan mahasiswa dalam bertanya 2. Kemampuan menyelesaikan tugas Bentuk Penilaian : Penilaian Praktikum	Kuliah, Tugas Praktikum : Membuat game pygame sederhana dengan menerapkan PBO [PB:1x(4x50")]	Asynchronous : Laporan kegiatan praktikum [PT,KM: (1,1)x(4x60")]	Materi: kelas, objek, abstraksi Pustaka: <i>Classes: Dokumentasi resmi mengenai class, object, dan abstraction.</i>	5%

10	Menerapkan konsep OOP dalam pembuatan game sederhana python	1. Memahami konsep dasar OOP (class, object, inheritance, encapsulation) dalam pengembangan game dengan PyGame 2. Mengaplikasikan konsep OOP dalam pembuatan game sederhana menggunakan PyGame 3. Membuat program game berbasis PyGame dengan menggunakan class, objek, dan prinsip-prinsip OOP	Kriteria: 1. Keaktifan mahasiswa dalam bertanya 2. Kemampuan menyelesaikan tugas Bentuk Penilaian : Penilaian Praktikum	Kuiah, Tugas Praktikum : Membuat game pygame sederhana dengan menerapkan PBO [PB:1x(4x50")]	Asynchronous : Laporan kegiatan praktikum [PT,KM: (1,1)x(4x60")]	Materi: kelas, objek, abstraksi Pustaka: <i>Classes: Dokumentasi resmi mengenai class, object, dan abstraction.</i>	5%
11	Menerapkan konsep OOP dalam pembuatan game sederhana python	1. Memahami konsep dasar OOP (class, object, inheritance, encapsulation) dalam pengembangan game dengan PyGame 2. Mengaplikasikan konsep OOP dalam pembuatan game sederhana menggunakan PyGame 3. Membuat program game berbasis PyGame dengan menggunakan class, objek, dan prinsip-prinsip OOP	Kriteria: 1. Keaktifan mahasiswa dalam bertanya 2. Kemampuan menyelesaikan tugas Bentuk Penilaian : Penilaian Praktikum	Kuiah, Tugas Praktikum : Membuat game pygame sederhana dengan menerapkan PBO [PB:1x(4x50")]	Asynchronous : Laporan kegiatan praktikum [PT,KM: (1,1)x(4x60")]	Materi: kelas, objek, abstraksi Pustaka: <i>Classes: Dokumentasi resmi mengenai class, object, dan abstraction.</i>	5%
12	Menerapkan konsep OOP dalam pembuatan proyek akhir	1. Mahasiswa mampu menentukan ide proyek berbasis PBO 2. Mahasiswa mampu membuat perancangan awal (blueprint) dari sistem menggunakan UML 3. Mahasiswa mampu mengimplementasikan class dan objek berdasarkan perancangan yang sudah dibuat 4. Mahasiswa mampu menerapkan prinsip OOP dalam kode mereka 5. Mahasiswa mampu menerapkan inheritance dan polimorfisme dalam proyek 6. Mahasiswa mampu menerapkan inheritance dan polimorfisme dalam proyek 7. Mahasiswa mampu menyelesaikan proyek berbasis PBO 8. Mahasiswa mampu mendemonstrasikan hasil proyek kepada kelas	Kriteria: 1. Keaktifan mahasiswa diskusi kelompok 2. Kemampuan menyelesaikan tugas Bentuk Penilaian : Penilaian Hasil Project / Penilaian Produk	Kuiah, Tugas Praktikum : Membuat proyek kelompok dengan menerapkan PBO [PB:1x(4x50")]	Asynchronous : Laporan kegiatan praktikum [PT,KM: (1,1)x(4x60")]	Materi: Inheritance Polimorfisme Pustaka: <i>Inheritance: Dokumentasi lengkap tentang inheritance, polymorphism, dan konsep terkait.</i>	10%

13	Menerapkan konsep OOP dalam pembuatan proyek akhir	1.Mahasiswa mampu menentukan ide proyek berbasis PBO 2.Mahasiswa mampu membuat perancangan awal (blueprint) dari sistem menggunakan UML 3.Mahasiswa mampu mengimplementasikan class dan objek berdasarkan perancangan yang sudah dibuat 4.Mahasiswa mampu menerapkan prinsip OOP dalam kode mereka 5.Mahasiswa mampu menerapkan inheritance dan polimorfisme dalam proyek 6.Mahasiswa mampu menerapkan inheritance dan polimorfisme dalam proyek 7.Mahasiswa mampu menyelesaikan proyek berbasis PBO 8.Mahasiswa mampu mendemonstrasikan hasil proyek kepada kelas	Kriteria: 1.Keaktifan mahasiswa diskusi kelompok 2.Kemampuan menyelesaikan tugas Bentuk Penilaian : Penilaian Hasil Project / Penilaian Produk	Kuiah, Tugas Praktikum : Membuat proyek kelompok dengan menerapkan PBO [PB:1x(4x50")]	Asynchronous : Laporan kegiatan praktikum [PT,KM: (1,1)x(4x60")]	Materi: Inheritance Polimorfisme Pustaka: <i>Inheritance: Dokumentasi lengkap tentang inheritance, polymorphism, dan konsep terkait.</i>	10%
14	Menerapkan konsep OOP dalam pembuatan proyek akhir	1.Mahasiswa mampu menentukan ide proyek berbasis PBO 2.Mahasiswa mampu membuat perancangan awal (blueprint) dari sistem menggunakan UML 3.Mahasiswa mampu mengimplementasikan class dan objek berdasarkan perancangan yang sudah dibuat 4.Mahasiswa mampu menerapkan prinsip OOP dalam kode mereka 5.Mahasiswa mampu menerapkan inheritance dan polimorfisme dalam proyek 6.Mahasiswa mampu menerapkan inheritance dan polimorfisme dalam proyek 7.Mahasiswa mampu menyelesaikan proyek berbasis PBO 8.Mahasiswa mampu mendemonstrasikan hasil proyek kepada kelas	Kriteria: 1.Keaktifan mahasiswa diskusi kelompok 2.Kemampuan menyelesaikan tugas Bentuk Penilaian : Penilaian Hasil Project / Penilaian Produk	Kuiah, Tugas Praktikum : Membuat proyek kelompok dengan menerapkan PBO [PB:1x(4x50")]	Asynchronous : Laporan kegiatan praktikum [PT,KM: (1,1)x(4x60")]	Materi: Inheritance Polimorfisme Pustaka: <i>Inheritance: Dokumentasi lengkap tentang inheritance, polymorphism, dan konsep terkait.</i>	10%

15	Menerapkan konsep OOP dalam pembuatan proyek akhir	1. Mahasiswa mampu menentukan ide proyek berbasis PBO 2. Mahasiswa mampu membuat perancangan awal (blueprint) dari sistem menggunakan UML 3. Mahasiswa mampu mengimplementasikan class dan objek berdasarkan perancangan yang sudah dibuat 4. Mahasiswa mampu menerapkan prinsip OOP dalam kode mereka 5. Mahasiswa mampu menerapkan inheritance dan polimorfisme dalam proyek 6. Mahasiswa mampu menerapkan inheritance dan polimorfisme dalam proyek 7. Mahasiswa mampu menyelesaikan proyek berbasis PBO 8. Mahasiswa mampu mendemonstrasikan hasil proyek kepada kelas	Kriteria: 1. Keaktifan mahasiswa diskusi kelompok 2. Kemampuan menyelesaikan tugas Bentuk Penilaian : Penilaian Hasil Project / Penilaian Produk	Kuliah, Tugas Praktikum : Membuat proyek kelompok dengan menerapkan PBO [PB:1x(4x50")]	Asynchronous : Laporan kegiatan praktikum [PT,KM: (1,1)x(4x60")]	Materi: Inheritance Polimorfisme Pustaka: <i>Inheritance: Dokumentasi lengkap tentang inheritance, polymorphism, dan konsep terkait.</i>	20%
16	Menerapkan konsep OOP dalam pembuatan proyek akhir	Mahasiswa mampu mendemonstrasikan hasil proyek kepada kelas	Kriteria: 1. Keaktifan mahasiswa diskusi kelompok 2. Kemampuan menyelesaikan tugas Bentuk Penilaian : Penilaian Hasil Project / Penilaian Produk	Evaluasi Akhir Semester / Ujian Akhir Semester : Demo Proyek [PB:1x(4x50")]	Asynchronous : Laporan Hasil Proyek [PT,KM: (1,1)x(4x60")]	Materi: Konsep OOP Pustaka: <i>Holmes, B.J., Joice D. T. 2001. Object-Oriented Programming With Python, second edition.</i>	0%

Rekap Persentase Evaluasi : Project Based Learning

No	Evaluasi	Persentase
1.	Penilaian Hasil Project / Penilaian Produk	50%
2.	Penilaian Praktikum	47.5%
3.	Praktik / Unjuk Kerja	2.5%
		100%

Catatan

- Capaian Pembelajaran Lulusan Prodi (CPL - Prodi)** adalah kemampuan yang dimiliki oleh setiap lulusan prodi yang merupakan internalisasi dari sikap, penguasaan pengetahuan dan ketrampilan sesuai dengan jenjang prodinya yang diperoleh melalui proses pembelajaran.
- CPL yang dibebankan pada mata kuliah** adalah beberapa capaian pembelajaran lulusan program studi (CPL-Prodi) yang digunakan untuk pembentukan/pengembangan sebuah mata kuliah yang terdiri dari aspek sikap, ketrampilan umum, ketrampilan khusus dan pengetahuan.
- CP Mata kuliah (CPMK)** adalah kemampuan yang dijabarkan secara spesifik dari CPL yang dibebankan pada mata kuliah, dan bersifat spesifik terhadap bahan kajian atau materi pembelajaran mata kuliah tersebut.
- Sub-CPMK Mata kuliah (Sub-CPMK)** adalah kemampuan yang dijabarkan secara spesifik dari CPMK yang dapat diukur atau diamati dan merupakan kemampuan akhir yang direncanakan pada tiap tahap pembelajaran, dan bersifat spesifik terhadap materi pembelajaran mata kuliah tersebut.
- Indikator penilaian** kemampuan dalam proses maupun hasil belajar mahasiswa adalah pernyataan spesifik dan terukur yang mengidentifikasi kemampuan atau kinerja hasil belajar mahasiswa yang disertai bukti-bukti.
- Kreteria Penilaian** adalah patokan yang digunakan sebagai ukuran atau tolok ukur ketercapaian pembelajaran dalam penilaian berdasarkan indikator-indikator yang telah ditetapkan. Kreteria penilaian merupakan pedoman bagi penilai agar penilaian konsisten dan tidak bias. Kreteria dapat berupa kuantitatif ataupun kualitatif.
- Bentuk penilaian:** tes dan non-tes.
- Bentuk pembelajaran:** Kuliah, Responsi, Tutorial, Seminar atau yang setara, Praktikum, Praktik Studio, Praktik Bengkel, Praktik Lapangan, Penelitian, Pengabdian Kepada Masyarakat dan/atau bentuk pembelajaran lain yang setara.
- Metode Pembelajaran:** Small Group Discussion, Role-Play & Simulation, Discovery Learning, Self-Directed Learning, Cooperative Learning, Collaborative Learning, Contextual Learning, Project Based Learning, dan metode lainnya yg setara.
- Materi Pembelajaran** adalah rincian atau uraian dari bahan kajian yg dapat disajikan dalam bentuk beberapa pokok dan sub-pokok bahasan.
- Bobot penilaian** adalah prosentasi penilaian terhadap setiap pencapaian sub-CPMK yang besarnya proposional dengan tingkat kesulitan pencapaian sub-CPMK tsb., dan totalnya 100%.
- TM=Tatap Muka, PT=Penugasan terstruktur, BM=Belajar mandiri.

RPS ini telah divalidasi pada tanggal 10 November 2025

Koordinator Program Studi D4
Manajemen Informatika



DODIK ARWIN DERMAWAN
NIDN 0008017807

UPM Program Studi D4
Manajemen Informatika



NIDN 0010089901

File PDF ini digenerate pada tanggal 6 Desember 2025 Jam 13:38 menggunakan aplikasi RPS-OBE SiDia Unesa

