|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | **UNIVERSITAS KRISTEN INDONESIA TORAJA**  **FAKULTAS TEKNIK**  **PROGRAM STUDI TEKNIK INFORMATIKA** | | | | | | | | | | | | **Kode Dokumen** | | |
| **RENCANA PEMBELAJARAN SEMESTER** | | | | | | | | | | | | | | | | |
| **MATA KULIAH (MK)** | | | **KODE** | | | **Rumpun MK** | | | **BOBOT (sks)** | | | **SEMESTER** | | **Tgl Penyusunan** | | |
|  | | |  | | |  | | | T= | | P= |  | |  | | |
| **Pemrograman Berorientasi Objek** | | | **Pengembang RPS** | | | **Koordinator RMK** | | | **GKM-F** | | | **Ketua PRODI** | | | | |
| **Melki Garonga’, S.Kom., M.Kom.** | | | Ttd | | | Ttd | | | **Aryo Michael, S.Kom., M.Kom.** | | | | |
| **Capaian Pembelajaran (CP)** | | **CPL-PRODI yang dibebankan pada MK** | | | |  | | | | | | | | | | |
| CPL04 | | Mampu berpikir kritis, logis, dan analitis dalam mengidentifikasi serta menyelesaikan permasalahan di bidang keahliannya dengan pendekatan ilmiah. | | | | | | | | | | | | |
| CPL07 | | Memahami prinsip-prinsip dasar di bidang teknologi informasi, komputasi, dan matematika yang menjadi fondasi pengembangan sistem serta penerapan teknologi digital. | | | | | | | | | | | | |
| CPL08 | | Menguasai teori-teori pengembangan perangkat lunak, sistem operasi, jaringan, dan keamanan data, serta metodologi manajemen proyek yang mendukung pembuatan solusi teknologi yang handal dan terintegrasi. | | | | | | | | | | | | |
| CPL14 | | Terampil dalam mengaplikasikan teknologi terbaru melalui proyek praktikum dan studi kasus, yang melibatkan simulasi, pemecahan masalah riil, serta inovasi berbasis teknologi digital di bidang IoT, AI, maupun Enterprise System. | | | | | | | | | | | | |
| **Capaian Pembelajaran Mata Kuliah (CPMK)** | | | | |  | | | | | | | | | |
| CPMK041 | | Mahasiswa mampu menganalisis permasalahan dan merancang solusi berbasis teknologi informasi secara logis dan sistematis sesuai dengan prinsip keilmuan di bidang informatika. | | | | | | | | | | | | |
| CPMK071 | | Mahasiswa mampu menjelaskan prinsip dasar teknologi informasi, pemrograman, dan arsitektur komputer sebagai landasan dalam merancang dan mengembangkan sistem berbasis digital. | | | | | | | | | | | | |
| CPMK081 | | Mahasiswa mampu menguasai prinsip-prinsip pemrograman dan arsitektur komputer serta menerapkan metode komputasi dalam merancang solusi perangkat lunak yang efisien dan terintegrasi. | | | | | | | | | | | | |
| CPMK141 | | Mahasiswa mampu menerapkan teknologi dan konsep terkini dalam pengembangan perangkat lunak dan sistem berbasis digital melalui praktik langsung dan studi kasus, khususnya pada pengembangan aplikasi dan sistem IoT. | | | | | | | | | | | | |
| **Kemampuan akhir tiap tahapan belajar (Sub-CPMK)** | | | | |  | | | | | | | | | |
| Sub-CPMK1 | | Mahasiswa dapat menjelaskan konsep dasar pemrograman berorientasi objek serta perbedaannya dengan pemrograman prosedural. | | | | | | | | | | | | |
| Sub-CPMK2 | | Mahasiswa mampu mengidentifikasi dan mendefinisikan class, object, attribute, dan method dalam pemrograman berorientasi objek. | | | | | | | | | | | | |
| Sub-CPMK3 | | Mahasiswa dapat menerapkan prinsip enkapsulasi untuk membatasi akses terhadap data. | | | | | | | | | | | | |
| Sub-CPMK4 | | Mahasiswa mampu menggunakan konsep pewarisan (inheritance) untuk memperluas fungsionalitas class. | | | | | | | | | | | | |
| Sub-CPMK5 | | Mahasiswa dapat mengimplementasikan konsep polymorphism dalam bentuk overloading dan overriding. | | | | | | | | | | | | |
| Sub-CPMK6 | | Mahasiswa mampu memanfaatkan konstruktor dan destruktor dalam pengelolaan objek. | | | | | | | | | | | | |
| Sub-CPMK7 | | Mahasiswa dapat menggunakan konsep interface dan abstract class dalam desain sistem. | | | | | | | | | | | | |
| Sub-CPMK8 | | Mahasiswa mampu menerapkan prinsip komposisi dan agregasi dalam membangun relasi antar objek. | | | | | | | | | | | | |
| Sub-CPMK9 | | Mahasiswa dapat menggunakan exception handling untuk mengelola kesalahan dalam program. | | | | | | | | | | | | |
| Sub-CPMK10 | | Mahasiswa mampu membuat dan mengelola koleksi data menggunakan struktur data berbasis objek seperti ArrayList, LinkedList, atau Map. | | | | | | | | | | | | |
| Sub-CPMK11 | | Mahasiswa dapat memanfaatkan file I/O untuk menyimpan dan membaca data dalam program berorientasi objek. | | | | | | | | | | | | |
| Sub-CPMK12 | | Mahasiswa mampu mendesain dan mengimplementasikan program berorientasi objek sesuai dengan studi kasus. | | | | | | | | | | | | |
| Sub-CPMK13 | | Mahasiswa dapat melakukan testing dan debugging terhadap program berorientasi objek. | | | | | | | | | | | | |
| Sub-CPMK14 | | Mahasiswa mampu mendokumentasikan dan mempresentasikan hasil pengembangan perangkat lunak berorientasi objek. | | | | | | | | | | | | |
| **Matriks CPL terhadap Sub-CPMK** | | | | |  | | | | | | | | | |
|  | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | |  | **Sub-CPMK1** | **Sub-CPMK2** | **Sub-CPMK3** | **Sub-CPMK4** | **Sub-CPMK5** | **Sub-CPMK6** | **Sub-CPMK7** | **Sub-CPMK8** | **Sub-CPMK9** | **Sub-CPMK10** | **Sub-CPMK11** | **Sub-CPMK12** | **Sub-CPMK13** | **Sub-CPMK14** | | **CPMK041** |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | **CPMK071** |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | **CPMK081** |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | **CPMK141** |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | | | | | | | | | | | | | | |
| **Deskripsi Singkat Mata Kuliah** | | Mata kuliah Pemrograman Berorientasi Objek membahas konsep dan penerapan paradigma pemrograman berorientasi objek (OOP) seperti enkapsulasi, pewarisan, dan polymorphism. Mahasiswa akan mempelajari teknik perancangan dan implementasi program menggunakan class dan objek, serta membangun aplikasi skala kecil hingga menengah yang modular, terstruktur, dan dapat dikembangkan. | | | | | | | | | | | | | | |
| **Bahan Kajian: Materi Pembelajaran** | | 1. Konsep dasar pemrograman berorientasi objek dan perbedaannya dengan pemrograman prosedural 2. Definisi class, object, attribute, dan method dalam OOP 3. Prinsip enkapsulasi dan akses kontrol (private, public, protected) 4. Penerapan inheritance dan konsep hirarki class 5. Konsep polymorphism: method overloading dan method overriding 6. Penggunaan konstruktor dan destruktor dalam pengelolaan objek 7. Perbedaan dan penerapan abstract class dan interface 8. Komposisi dan agregasi dalam hubungan antar objek 9. Penanganan error dan exception (try-catch-finally) 10. Penggunaan struktur data berbasis objek: array, ArrayList, LinkedList, Map 11. Pemrograman file I/O: membaca dan menulis data ke file 12. Desain dan implementasi aplikasi berbasis objek menggunakan studi kasus 13. Proses testing dan debugging program berorientasi objek 14. Dokumentasi dan presentasi hasil pengembangan aplikasi OOP | | | | | | | | | | | | | | |
| **Pustaka** | | **Utama:** |  | | | | | | | | | | | | | |
| 1. Rizky Budi Haryanto, Pemrograman Berorientasi Objek: Teori dan Implementasi dengan Java., Penerbit, 2021. 2. Agus Suryadi, M.Kom, dkk., Buku Ajar Pemrograman Berorientasi Objek., Penerbit Pustaka Galeri Mandiri, 2020. 3. Nazuarsyah, ST., MT, Dhea Aulia, Cut Nurul A’la, Inda Fatya, Buku Ajar Dasar-Dasar Pemrograman Berorientasi Objek., Bandar Publishing, 2020. 4. Ilham Fadhil, Pemrograman Berorientasi Objek dengan Java: Panduan Lengkap untuk Pemula., Penerbit Andi Offset, 2020. 5. Dewi Rachmawati, Modul Pembelajaran Pemrograman Berorientasi Objek (PBO)., Penerbit Universitas XYZ, 2020. 6. Rudi Hartono, Pemrograman Berorientasi Objek dalam C++: Teori dan Praktik., Penerbit Informatika Bandung, 2019. 7. Diana, Dasar-Dasar Pemrograman Berorientasi Objek: Konsep dan Implementasi., Penerbit Salemba Empat, 2019. | | | | | | | | | | | | | | |
| **Pendukung:** |  | | | | | | | | | | | | | |
|  | | | | | | | | | | | | | | |
| **Dosen Pengampu** | | Melki Garonga’, S.Kom., M.Kom. | | | | | | | | | | | | | | |
| **Mata kuliah syarat** | |  | | | | | | | | | | | | | | |
| **Mg Ke-** | **Kemampuan akhir tiap tahapan belajar**  **(Sub-CPMK)** | | **Penilaian** | | | | | **Bantuk Pembelajaran,**  **Metode Pembelajaran,**  **Penugasan Mahasiswa,**  **[ Estimasi Waktu]** | | | | | **Materi Pembelajaran**  **[Pustaka]** | | **Bobot Penilaian (%)** |
| **Indikator** | | **Kriteria dan Teknik** | | | **Luring (*offline*)** | | **Daring (*online*)** | | |
| **(1)** | **(2)** | | **(3)** | | **(4)** | | | **(5)** | | **(6)** | | | **(7)** | | **(8)** |
| 1 | Mahasiswa dapat menjelaskan konsep dasar pemrograman berorientasi objek serta perbedaannya dengan pemrograman prosedural. | | 1. Menjelaskan prinsip dasar OOP. 2. Menguraikan perbedaan OOP dan prosedural programming. 3. Memberikan contoh penerapan OOP dalam bahasa pemrograman tertentu. | | **Kriteria:**   1. Rubrik penilaian   (terlampir)   1. Mahasiswa   memberikan  respon terhadap  materi kuliah,  setiap respon  bernilai 5  **Bentuk Penilaian :**  Aktifitas Partisipasif | | | **Pendekatan:**  Saintifik Model:  Pembelajaran Berbasis masalah  **Metode:**  Diskusi, Presentasi  (3x50) | |  | | | **Materi:**  Konsep dasar OOP, paradigma OOP vs prosedural, kelebihan dan kelemahan masing-masing.  **Pustaka:**  *Agus Suryadi, M.Kom, dkk., Buku Ajar Pemrograman Berorientasi Objek., Penerbit Pustaka Galeri Mandiri, 2020.* | | 5% |
| 2 | Mahasiswa mampu mengidentifikasi dan mendefinisikan class, object, attribute, dan method dalam pemrograman berorientasi objek. | | 1. Menjelaskan pengertian class dan object. 2. Mendefinisikan attribute dan method dalam sebuah class. | | **Kriteria:**   1. Rubrik penilaian (terlampir) 2. Mahasiswa   memberikan  respon terhadap  materi kuliah,  setiap respon  bernilai 5  **Bentuk Penilaian :**  Aktifitas Partisipasif | | | **Pendekatan:**  Saintifik Model:  Pembelajaran Berbasis masalah  **Metode:**  Diskusi, Presentasi  (3x50) | |  | | | **Materi:**  Struktur class dan object, pembuatan attribute dan method, contoh program class sederhana.  **Pustaka:**  *Agus Suryadi, M.Kom, dkk., Buku Ajar Pemrograman Berorientasi Objek., Penerbit Pustaka Galeri Mandiri, 2020.* | | 2% |
| 3 | Mahasiswa dapat menerapkan prinsip enkapsulasi untuk membatasi akses terhadap data. | | 1. Menggunakan modifier akses (private, public, protected). 2. Mengimplementasikan getter dan setter. 3. Menjelaskan manfaat enkapsulasi dalam pengamanan data. | | **Kriteria:**   1. Rubrik penilaian (terlampir) 2. Mahasiswa   memberikan  respon terhadap  materi kuliah,  setiap respon  bernilai 5  **Bentuk Penilaian :**  Aktifitas Partisipasif | | | **Pendekatan:**  Saintifik Model:  Pembelajaran Berbasis masalah  **Metode:**  Diskusi, Presentasi  (3x50) | |  | | | **Materi:**  Konsep enkapsulasi, implementasi kontrol akses, penggunaan getter dan setter.  **Pustaka:**  *Agus Suryadi, M.Kom, dkk., Buku Ajar Pemrograman Berorientasi Objek., Penerbit Pustaka Galeri Mandiri, 2020.* | | 3% |
| 4 | Mahasiswa mampu menggunakan konsep pewarisan (inheritance) untuk memperluas fungsionalitas class. | | 1. Menjelaskan prinsip inheritance. 2. Menerapkan inheritance dalam program. | | **Kriteria:**   1. Rubrik penilaian (terlampir) 2. Mahasiswa   memberikan  respon terhadap  materi kuliah,  setiap respon  bernilai 5  **Bentuk Penilaian :**  Aktifitas Partisipasif | | | **Pendekatan:**  Saintifik Model:  Pembelajaran Berbasis masalah  **Metode:**  Diskusi, Presentasi  (3x50) | |  | | | **Materi:**  Pewarisan class, pewarisan atribut dan method, penggunaan extends (Java/PHP/C++).  **Pustaka:**  *Agus Suryadi, M.Kom, dkk., Buku Ajar Pemrograman Berorientasi Objek., Penerbit Pustaka Galeri Mandiri, 2020.* | | 3% |
| 5 | Mahasiswa dapat mengimplementasikan konsep polymorphism dalam bentuk overloading dan overriding. | | 1. Menjelaskan konsep polymorphism. 2. Mengimplementasikan overloading dan overriding. 3. Membedakan method overloading dan overriding. | | **Kriteria:**   1. Rubrik penilaian (terlampir) 2. Mahasiswa   memberikan  respon terhadap  materi kuliah,  setiap respon  bernilai 5  **Bentuk Penilaian :**  Aktifitas Partisipasif | | | **Pendekatan:**  Saintifik Model:  Pembelajaran Berbasis masalah  **Metode:**  Diskusi, Presentasi  (3x50) | |  | | | **Materi:**  Konsep polymorphism, method overloading dan overriding, contoh program.  **Pustaka:**  *Agus Suryadi, M.Kom, dkk., Buku Ajar Pemrograman Berorientasi Objek., Penerbit Pustaka Galeri Mandiri, 2020.* | | 3% |
| 6 | Mahasiswa mampu memanfaatkan konstruktor dan destruktor dalam pengelolaan objek. | | 1. Menggunakan konstruktor untuk inisialisasi objek. 2. Mengimplementasikan destruktor untuk pembersihan objek. | | **Kriteria:**   1. Rubrik penilaian (terlampir) 2. Mahasiswa   memberikan  respon terhadap  materi kuliah, setiap respon  bernilai 5  **Bentuk Penilaian :**  Aktifitas Partisipasif | | | **Pendekatan:**  Saintifik Model:  Pembelajaran Berbasis masalah  **Metode:**  Diskusi, Presentasi  (3x50) | |  | | | **Materi:**  Konstruktor default dan parameter, destruktor dan siklus hidup objek.  **Pustaka:**  *Agus Suryadi, M.Kom, dkk., Buku Ajar Pemrograman Berorientasi Objek., Penerbit Pustaka Galeri Mandiri, 2020.* | | 3% |
| 7 | Mahasiswa dapat menggunakan konsep interface dan abstract class dalam desain sistem. | | 1. Menjelaskan perbedaan abstract class dan interface. 2. Mengimplementasikan interface dalam program. 3. Mendesain class turunan dari abstract class. | | **Kriteria:**   1. Rubrik penilaian (terlampir) 2. Mahasiswa   memberikan  respon terhadap  materi kuliah, setiap respon  bernilai 5  **Bentuk Penilaian :**  Aktifitas Partisipasif | | | **Pendekatan:**  Saintifik Model:  Pembelajaran Berbasis masalah  **Metode:**  Diskusi, Presentasi  (3x50) | |  | | | **Materi:**  Konsep abstract class dan interface, penerapan dalam desain modular.  **Pustaka:**  *Agus Suryadi, M.Kom, dkk., Buku Ajar Pemrograman Berorientasi Objek., Penerbit Pustaka Galeri Mandiri, 2020.* | | 3% |
| 8 | **Evaluasi Tengah Semester / UjianTengan Semester** | | | | | | | | | | | | | |  |
| 9 | Mahasiswa mampu menerapkan prinsip komposisi dan agregasi dalam membangun relasi antar objek. | | 1. Menjelaskan konsep komposisi dan agregasi. 2. Membuat relasi antar class menggunakan komposisi atau agregasi. | | **Kriteria:**   1. Rubrik penilaian (terlampir) 2. Mahasiswa memberikan   respon terhadap  materi kuliah, setiap respon  bernilai 5  **Bentuk Penilaian :**  Aktifitas Partisipasif | | | **Pendekatan:**  Saintifik Model:  Pembelajaran Berbasis masalah  **Metode:**  Diskusi, Presentasi  (3x50) | |  | | | **Materi:**  Hubungan antar objek, class dalam class, diagram UML untuk komposisi/agregasi.  **Pustaka:**  *Agus Suryadi, M.Kom, dkk., Buku Ajar Pemrograman Berorientasi Objek., Penerbit Pustaka Galeri Mandiri, 2020.* | | 3% |
| 10 | Mahasiswa dapat menggunakan exception handling untuk mengelola kesalahan dalam program. | | 1. Menjelaskan jenis-jenis exception. 2. Mengimplementasikan try-catch-finally. 3. Menulis exception kustom (custom exception). | | **Kriteria:**   1. Rubrik penilaian (terlampir) 2. Mahasiswa memberikan respon terhadap materi kuliah, setiap respon bernilai 5   **Bentuk Penilaian :**  Aktifitas Partisipasif | | | **Pendekatan:**  Saintifik Model:  Pembelajaran Berbasis masalah  **Metode:**  Diskusi, Presentasi  (3x50) | |  | | | **Materi:**  Penanganan kesalahan (runtime), struktur try-catch, throw dan throws.  **Pustaka:**  *Agus Suryadi, M.Kom, dkk., Buku Ajar Pemrograman Berorientasi Objek., Penerbit Pustaka Galeri Mandiri, 2020.* | | 3% |
| 11 | Mahasiswa mampu membuat dan mengelola koleksi data menggunakan struktur data berbasis objek seperti ArrayList, LinkedList, atau Map. | | 1. Menggunakan struktur data collection dalam program. 2. Membedakan jenis koleksi data dan penggunaannya. | | **Kriteria:**   1. Rubrik penilaian (terlampir) 2. Mahasiswa memberikan respon terhadap materi kuliah, setiap respon bernilai 5   **Bentuk Penilaian :**  Aktifitas Partisipasif | | | **Pendekatan:**  Saintifik Model:  Pembelajaran Berbasis masalah  **Metode:**  Diskusi, Presentasi  (3x50) | |  | | | **Materi:**  ArrayList, LinkedList, HashMap, dan implementasi penggunaannya.  **Pustaka:**  *Agus Suryadi, M.Kom, dkk., Buku Ajar Pemrograman Berorientasi Objek., Penerbit Pustaka Galeri Mandiri, 2020.* | | 3% |
| 12 | Mahasiswa dapat memanfaatkan file I/O untuk menyimpan dan membaca data dalam program berorientasi objek. | | 1. Mengimplementasikan pembacaan dan penulisan file. 2. Menyimpan data objek dalam file. | | **Kriteria:**   1. Rubrik penilaian (terlampir) 2. Mahasiswa memberikan respon terhadap materi kuliah, setiap respon bernilai 5   **Bentuk Penilaian :**  Aktifitas Partisipasif | | | **Pendekatan:**  Saintifik Model:  Pembelajaran Berbasis masalah  **Metode:**  Diskusi, Presentasi  (3x50) | |  | | | **Materi:**  Pemrograman file I/O (input/output), stream, pembacaan dan penulisan file teks.  **Pustaka:**  *Agus Suryadi, M.Kom, dkk., Buku Ajar Pemrograman Berorientasi Objek., Penerbit Pustaka Galeri Mandiri, 2020.* | | 5% |
| 13 | Mahasiswa mampu mendesain dan mengimplementasikan program berorientasi objek sesuai dengan studi kasus. | | 1. Menyusun class diagram berdasarkan studi kasus. 2. Mengimplementasikan program secara modular. 3. Menyusun struktur program sesuai desain. 4. Melakukan pengujian awal fungsionalitas program. | | **Kriteria:**   1. Rubrik penilaian (terlampir) 2. Mahasiswa memberikan respon terhadap materi kuliah, setiap respon bernilai 5   **Bentuk Penilaian :**  Aktifitas Partisipasif | | | **Pendekatan:**  Saintifik Model:  Pembelajaran Berbasis masalah  **Metode:**  Diskusi, Presentasi  (3x50) | |  | | | **Materi:**  Analisis kebutuhan, class diagram, implementasi sistem berbasis objek.  **Pustaka:**  *Agus Suryadi, M.Kom, dkk., Buku Ajar Pemrograman Berorientasi Objek., Penerbit Pustaka Galeri Mandiri, 2020.* | | 5% |
| 14 | Mahasiswa dapat melakukan testing dan debugging terhadap program berorientasi objek. | | 1. Menemukan dan memperbaiki kesalahan sintaks dan logika. 2. Menggunakan tools debugging atau print log. 3. Melakukan uji fungsional pada method. | | **Kriteria:**   1. Rubrik penilaian (terlampir) 2. Mahasiswa memberikan respon terhadap materi kuliah, setiap respon bernilai 5   **Bentuk Penilaian :**  Aktifitas Partisipasif | | | **Pendekatan:**  Saintifik Model:  Pembelajaran Berbasis masalah  **Metode:**  Diskusi, Presentasi  (3x50) | |  | | | **Materi:**  Testing program, teknik debugging, tools bantu seperti IDE debugger.  **Pustaka:**  *Agus Suryadi, M.Kom, dkk., Buku Ajar Pemrograman Berorientasi Objek., Penerbit Pustaka Galeri Mandiri, 2020.* | | 5% |
| 15 | Mahasiswa mampu mendokumentasikan dan mempresentasikan hasil pengembangan perangkat lunak berorientasi objek. | | 1. Menyusun dokumentasi teknis. 2. Menyajikan hasil pengembangan dalam bentuk laporan. 3. Memaparkan hasil proyek secara lisan atau visual. | | **Kriteria:**   1. Rubrik penilaian (terlampir) 2. Mahasiswa memberikan respon terhadap materi kuliah, setiap respon bernilai 5   **Bentuk Penilaian :**  Aktifitas Partisipasif | | | **Pendekatan:**  Saintifik Model:  Pembelajaran Berbasis masalah  **Metode:**  Diskusi, Presentasi  (3x50) | |  | | | **Materi:**  Dokumentasi program (technical writing), format laporan proyek, teknik presentasi.  **Pustaka:**  *Agus Suryadi, M.Kom, dkk., Buku Ajar Pemrograman Berorientasi Objek., Penerbit Pustaka Galeri Mandiri, 2020.* | | 5% |
| 16 | **Evaluasi Akhir Semester / Ujian Akhir Semester** | | | | | | | | | | | | | |  |

|  |  |  |
| --- | --- | --- |
| **BOBOT** | **RENTANG NILAI** | **HURU F** |
| 4.00 | >86 | A |
| 3.75 | 80-85 | A- |
| 3.50 | 74-79 | B+ |
| 3.00 | 68-73 | B |
| 2.75 | 62-67 | B- |
| 2,50 | 56-61 | C+ |
| 2.00 | 50-55 | C |
| 1.00 | 44-49 | D |
| 0.00 | <43 | E |

|  |  |
| --- | --- |
| **ASPEK PENILAIAN** | **PERSEN- TASE** |
| UAS (Penilaian Proyek) | 30 % |
| UTS | 30 % |
| Tugas (Tg) membuat cerita dan simulasi cerita | 20 % |
| (Partisipasi Aktif (PA)) | 20 % |

Rumus Nilai Akhir Mata kuliah:

**NA = (20 X RP, RPA) + (20 X RTG) + (20 X RUTS) + (40 X RUAS)**

**EVALUASI**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **BENTUK TES** | **JENIS TES** | **KISI-KISI INSTRUMEN PENILAIAN** | **INSTRUMEN PENILAIAN** | **RUBRIK PENILAIAN** |
| Tes/ Non Tes/ Lembar Observasi Kinerja | Lisan/ Tertulis/ Praktik Kinerja/ Observasi | Terlampir | Terlampir | Terlampir |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**KISI-KISI INSTRUMEN PENILAIAN**

| **NO** | **KEMAMPUAN AKHIR YANG DIHARAPKAN**  **(SUB-CPMK)** | **BENTUK INSTRUMEN**  **(PILIHAN GANDA/ URAIAN/ OBSERVASI/ PRAKTIK)** | **ASPEK** | | | **NOMOR BUTIR SOAL** |
| --- | --- | --- | --- | --- | --- | --- |
| **KOGNITIF**  **(C1-C6)** | **AFEKTIF**  **(A1-A5)** | **PSIMOTORIK**  **(P1-P5)** |  |
| 1. | SUB-CPMK 1 |  |  |  |  |  |
| 2. |  |  |  |  |  |  |
| 3. |  |  |  |  |  |  |
| 4. |  |  |  |  |  |  |
| 5. |  |  |  |  |  |  |
| 6. |  |  |  |  |  |  |
| 7. |  |  |  |  |  |  |
| 8. |  |  |  |  |  |  |
| 9. |  |  |  |  |  |  |
| 10. |  |  |  |  |  |  |
| 11. |  |  |  |  |  |  |
| 12. |  |  |  |  |  |  |
| 13. |  |  |  |  |  |  |

RUBRIK SKALA PERSEPSI

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Aspek/Dimensi yang Dinilai** | **Sangat Kurang** | **Kurang** | **Cukup** | **Baik** | **Sangat Baik** |
| **<20** | **(21-40)** | **(41-60)** | **(61-80)** | **>80** |
| Kemampuan Komunikasi |  |  |  |  |  |
| Penguasaan Materti |  |  |  |  |  |
| Kemampuan Menghadapi Pertanyaan |  |  |  |  |  |
| Penggunaan Alat Peraga Persentasi |  |  |  |  |  |
| Ketepatan Menyelesaikan Masalah |  |  |  |  |  |

**INSTRUMEN PENILAIAN**

*Lampirkan*

**RUBRIK PENILAIAN**

*Lampirkan*

**CATATAN DAN KETERANGAN:**

**Evaluasi dan Penilaian Mata Kuliah**

1. **Ujian Tengah Semester (UTS)**

Materi yang akan diujikan meliputi materi perkuliahan pada pertemuan pertama sampai pertemuan ke tujuh/delapan dengan memberikan beberapa soal/tugas kepada mahasiswa.

1. **Ujian Akhir Semester (UAS)**

Materi yang akan diujikan meliputi materi perkuliahan pada pertemuan pertama sampai terakhir, yang dilaksanakan sesuai dengan kalender akademik.

1. ***Performance* (Tugas dan Partisipasi Aktif)**

Nilai performance merupakan penilaian yang diambilkan dari aktivitas kelas meliputi: penyelesaian tugas terstruktur maupun mandiri dengan baik dan tepat waktu, presensi, keaktifan berpartisipasi dalam diskusi, etika dalam perkuliahan dan diskusi, menghargai teman, dan sebagainya yang dianggap perlu sebagai penunjang.