UNIVERSITAS HASANUDDIN
FAKULTAS TEKNIK
PROGRAM STUDI TEKNIK INFORMATIKA - S1
RENCANA PEMBELAJARAN SEMESTER
(*SEMESTER LECTURE PLAN*)

Mata Kuliah (MK)	KODE	Rumpun MK	BOBOT (sks)	SEMESTER	Tgl Penyusunan
Pemrograman Berorientasi Obyek	23D12120403		3	3	

OTORITAS	Dosen Pengembang RPS	Koordinator MK	Ketua PRODI
	Dr. Eng. Ir. Muhammad Niswar, S.T., M.InfoTech., Elly Warni, ST.,MT., Novy Nur R.A Mokobombang, ST., Ms.TM., Ph.D., A. Ais Prayogi, ST., M.Eng, Tyanita Puti Marindah, S.T., M.Inf.	Dr. Eng. Ir. Muhammad Niswar, S.T., M.InfoTech.	Prof. Dr. Ir. Indrabayu, S.T., M.T., M.Bus.Sys., IPM., ASEAN.Eng.

Capaian Pembelajaran Mata Kuliah	**CPL-PRODI yang dibebankan pada MK**	
	CPL-1:	Memiliki pengetahuan dasar Ilmu Komputer, meliputi algoritma, pemrograman, rekayasa perangkat lunak, serta Matematika dan Statistika dan mampu mengaplikasikannya dalam proses penyelesaian permasalahan komputasi.
	CPL-4:	Mampu mendesain, mengimplementasikan, dan mengevaluasi solusi berbasis komputasi dengan menerapkan teori Ilmu Komputer dan dasar-dasar pengembangan perangkat lunak, mencakup analisis kebutuhan, pemodelan, implementasi, dan evaluasi sistem serta komponennya.
	CPL-6:	Mampu bekerja secara efektif dalam tim, baik sebagai pemimpin maupun anggota, dan menjalankan peran dalam mengelola tugas sendiri serta kinerja kolaboratif dalam kegiatan yang terkait dengan disiplin program
	CPL ⇒ Capaian Pembelajaran Mata Kuliah (CPMK)	
	Setelah menyelesaikan mata kuliah ini, diharapkan:	
	CPL-1 (IK 1.1, IK 1.3):	**CPMK-1.1 :** Mahasiswa mampu menerapkan dan menganalisis konsep dasar pemrograman berorientasi objek sebagai landasan abstraksi dan penyelesaian permasalahan komputasi.
	CPL-4 (IK 4.1, IK 4.3):	**CPMK-4.1 :** Mahasiswa mampu menganalisis dan mengevaluasi desain perangkat lunak berorientasi objek menggunakan prinsip dan metode pengembangan perangkat lunak.
	CPL-6 (IK 6.1):	**CPMK-6.1 :** Mahasiswa mampu bekerja secara efektif dalam tim pengembangan perangkat lunak berorientasi objek dengan menjalankan peran sebagai anggota maupun pemimpin, serta bertanggung jawab atas pengembangan komponen *front-end* dan/atau *back-end* dalam proyek tim.
	CPMK ⇒ Sub-CPMK	
	CPMK-1.1	**SUB-CPMK-1.1.1:** Mahasiswa **mampu menerapkan dan menganalisis** konsep dasar pemrograman berorientasi objek yang meliputi *class*, *object*, *encapsulation*, *inheritance*, dan *polymorphism* dalam penyusunan solusi permasalahan komputasi terstruktur secara tepat. (IK 1.1)

		SUB-CPMK-1.1.2: Mahasiswa mampu memodelkan sistem berorientasi objek menggunakan *Unified Modeling Language (UML)* (IK 1.3)
	CPMK-4.1	**SUB-CPMK-4.1.1:** Mahasiswa mampu merancang struktur perangkat lunak dengan menerapkan prinsip *SOLID dan common class* untuk menghasilkan desain yang modular dan berkelanjutan. (IK 4.1) **SUB-CPMK-4.1.2:** Mahasiswa mampu memverifikasi kualitas perangkat lunak melalui *testing, debugging, unit testing,* dan *automation*. (IK 4.3)
	CPMK-6.1	**SUB-CPMK-6.1.1:** Mahasiswa mampu berkolaborasi dalam tim untuk mengintegrasikan komponen *front-end* dan *back-end* serta mengevaluasi kinerja kolaboratif dalam proyek pemrograman berorientasi objek. (IK 6.1)

Korelasi antara CPL/CPMK terhadap Sub-CPMK

CPL yang dibebankan pada MK	CPMK	SUB CPMK	Bentuk Asesmen*						Bobot	Nilai	Skor Mahasiswa
			Formative	Sumative							
				Quiz	Partisipasi	Mid Tes	Tugas	Hasil Project			
CPL-1	CPMK-1.1	SUB-CPMK-1.1.1		10	10	0	0	0	20		
		SUB-CPMK-1.1.2		0	10	20	0	0	30		
CPL-4	CPMK-4.1	SUB-CPMK-4.1.1		0	0	0	10	0	10		
		SUB-CPMK-4.1.2		0	0	0	10	0	10		
CPL-6	CPMK-6.1	SUB-CPMK-6.1.1		0	0	0	0	30	30		
				10	20	20	20	30	100		

Deskripsi Matakuliah	Mata kuliah ini mempelajari konsep-konsep dasar pada object oriented programming (OOP), seperti class – object, encapsulation, inheritance, polymorphism, interface, abstract class. Mahasiswa akan mempelajari bagaimana membangun program aplikasi dengan konsep pemrograman berorientasi objek, Secara spesifik konsep-konsep tersebut akan dipelajari dengan bantuan bahasa pemrograman Java. Metode Pembelajaran bauran, teori dan praktik.
Materi Pembelajaran /Pokok Bahasan	1. Pengenalan pemrograman berorientasi obyek 2. Dasar pemrograman java 3. Konsep pemrograman berorientasi obyek (class, obyek, encapsulation, inheritance, polymorphism) 4. Unified Modeling Language (UML) 5. Common class 6. Testing, Debugging, Unit Testing, Automation 7. Studi kasus
Referensi	**Referensi Utama**
	1. Introduction to Object Oriented Programming with Java (C. Thomas Wu) 2. A Comprehensive Introduction to Object Oriented Programming with Java (C. Thomas Wu)
	Referensi Tambahan

	Object First with Java (David Barnes, Michael Kolling)
Tim Pengajaran	Dr. Eng. Ir. Muhammad Niswar, S.T., M.InfoTech., Elly Warni, ST.,MT., Novy Nur R.A Mokobombang, ST., Ms.TM., Ph.D., A. Ais Prayogi, ST., M.Eng, Tyanita Puti Marindah, S.T., M.Inf.
Mata kuliah syarat	Dasar Pemrograman Komputer, Matematika Diskrit, Algoritma dan Struktur Data

Pertemuan Ke-	**Sub CPMK (Kemampuan akhir tiap tahapan belajar)**	**Penilaian (*Assesment*)**		**Bentuk dan Metode Pembelajaran [estimasi waktu] (*Learning Method*)**		**Materi Pembelajaran (*Content*)**	**Bobot Penilaian (%)**
		Indikator (*Indicator*)	**Teknik & Kriteria (*Techniques & Criteria*)**	**Luring (*Offline System*)**	**Daring (*Online System*)**		
1	**2**	**3**	**4**	**5**	**6**	**7**	**8**
1	CPL 1 : Mahasiswa memperoleh konsep dasar pemrograman berorientasi obyek yaitu class, object, encapsulation, inheritance, dan polymorphism (Sub -CPMK- 1.1.1)	**Formative:** - **Sumative:** Memperoleh konsep dasar OOP	**Kriteria Formative:** **Kriteria Sumative:** **Teknik Penilaian:** None	**Kuliah:** Metode lainnya 3x50 menit		1. OOP paradigm dan procedural paradigm 1. Konsep OOP 2. OOP benefit 3. OOP component 1.	0
2-3	CPL 1 : Mahasiswa memperoleh konsep dasar pemrograman berorientasi obyek yaitu class, object, encapsulation, inheritance, dan polymorphism (Sub-CPMK 1.1.1)	**Formative:** - **Sumative:** Memperoleh dasar pemrograman java dan konsep kelas, object, Method, Property, Access Modifier, Constructor, dan mampu membuat kelas (sederhana) dan object dalam bahasa Java	**Kriteria Formative:** **Kriteria Sumative:** Quiz (10) **Teknik Penilaian:** None	**Kuliah:** Metode lainnya 6x50 menit		1. Struktur dasar, Input output, Tipe data, string 1. Operator, Percabangan, Perulangan, Array 1. Konsep kelas,obyek, Method, Property, Access Modifier, Constructor	10

4-5	CPL 1 : Mahasiswa memperoleh konsep dasar pemrograman berorientasi obyek yaitu class, object, encapsulation, inheritance, dan polymorphism (Sub-CPMK 1.1.1)	**Formative:** - **Sumative:** Mendiskusikan konsep inheritance, polimorfism, overloading, dan encapsulation	**Kriteria Formative:** **Kriteria Sumative:** Partisipasi (10) **Teknik Penilaian:** None	**Kuliah:** Metode lainnya 6x50 menit		1. Inheritance, Polimorfisme 1. Overloading, Encapsulation	10
6-7	CPL 1 : Mahasiswa mampu memodelkan sistem berorientasi objek menggunakan *Unified Modeling Language (UML)* (Sub-CPMK 1.1.2)	**Formative:** - **Sumative:** Memperoleh dan menerapkan konsep unified modeling language (UML)	**Kriteria Formative:** **Kriteria Sumative:** Partisipasi (10) **Teknik Penilaian:** None	**Kuliah:** Diskusi kelompok (Small Group Discussion),Studi Kasus (Case Study) 6x50 menit		Konsep dan implementasi unified modeling language (UML)	10
8	CPL 1 : Mahasiswa memperoleh konsep dasar pemrograman berorientasi obyek yaitu class, object, encapsulation, inheritance, dan polymorphism (Sub-CPMK 1.1.1)	**Formative:** - **Sumative:** -	**Kriteria Formative:** **Kriteria Sumative:** Mid Tes (20) **Teknik Penilaian:** None				20
9-10	CPL 4 : Mahasiswa mampu merancang struktur perangkat lunak dengan menerapkan prinsip *SOLID* untuk menghasilkan desain yang modular dan berkelanjutan. (Sub-CPMK 4.1.1)	**Formative:** - **Sumative:** Menerapkan konsep SOLID Principles	**Kriteria Formative:** **Kriteria Sumative:** Tugas (10) dinilai dengan rubrik 23D121 **Teknik Penilaian:** None	**Kuliah:** Metode lainnya 3x50 menit **Kuliah:** Studi Kasus (Case Study),Pembelajaran kolaboratif (Collaborative Learning) 2x3x50 menit		SOLID Principles	10
11-12	CPL 4 : Mahasiswa mampu menerapkan konsep unified modeling language (UML), SOLID Principles, common class,	**Formative:** - **Sumative:** Mengetahuikonsep	**Kriteria Formative:** **Kriteria Sumative:** - **Teknik Penilaian:**	**Kuliah:** Metode lainnya 6x50 menit		Konsep common class (Generics,Exception, input/output stream, Datetime, Casting)	0

	testing, debugging, unit testing, automation (Sub-CPMK 4.1.1)	common class (Generics,Exception, input/output stream, Datetime, Casting)	None				
13	CPL 4 : Mahasiswa mampu memverifikasi kualitas perangkat lunak melalui *testing, debugging, unit testing,* dan *automation* (Sub-CPMK 4.1.2)	**Formative:** - **Sumative:** Mengetahuikonsep testing, debugging, unit testing, automation	**Kriteria Formative:** **Kriteria Sumative:** Tugas (10) dinilai dengan rubrik 23D121 **Teknik Penilaian:** None	**Kuliah:** Metode lainnya 3x50 menit		Testing, debugging, unit testing, automation	10
14-16	CPL 6 : Mahasiswa mampu bekerjasama dalam pendelegasian tugas dalam menyelesaikan studi kasus nyata pemrograman berorientasi objek (Sub-CPMK 6.1.1)	**Formative:** **-** **Sumative:** Membangun program aplikasi	**Kriteria Formative:** **Kriteria Sumative:** Hasil Project (30) **Teknik Penilaian:** None	**Kuliah:** Diskusi kelompok (Small Group Discussion),Studi Kasus (Case Study) 9x50 menit		-	30
							100