



**YAYASAN DEWI NURDIN HAMZAH  
UNIVERSITAS NURDIN HAMZAH**




**FAKULTAS ILMU KOMPUTER (FILKOM)**

**PROGRAM STUDI SARJANA SISTEM INFORMASI**

Jl.Kol.Abunjani Sipin – Jambi. Telp (0741) 668723 http://www.unh.ac.id –  
sistem.informasi@unh.ac.id

Kode

**RENCANA PEMBELAJARAN SEMESTER**

Mata Kuliah (MK)	Kode	Bahan Kajian (BK)	Bobot ( 4 / SKS)		Semester	Tanggal Penyusunan
			Praktek	Teori		
Pemrograman Berorientasi Objek	SIKB6409	2-8	Praktek	-	6 (Enam)	
Pengesahan	Dosen Pengembang RPS		Koordinator BK		Ketua Program Studi Sistem Infromasi	
	 Junaidi Surya, M.Kom NIDN : 1010107601		 Junaidi Surya, M.Kom NIDN : 1010107601		 Junaidi Surya, M.Kom NIDN : 1010107601	
Capaian Pembelajaran	<b>CPL – Program Studi yang dibeban pada Mata Kuliah (MK)</b>					
	<b>Sikap</b>	S9	Menunjukkan sikap bertanggung jawab dalam pengembangan perangkat lunak.			
		S11	Memiliki integritas dan kejujuran dalam penulisan kode (anti-plagiarisme).			
	<b>Pengetahuan</b>	P3	Menguasai konsep pemrograman berorientasi objek (enkapsulasi, inheritance, polimorfisme, abstraksi).			
	<b>Keterampilan Umum</b>	KU.1	Mampu berpikir logis, sistematis, dan terstruktur dalam pemecahan masalah perangkat lunak..			
	<b>Keterampilan Khusus</b>	KK.1	Mampu merancang dan mengimplementasikan aplikasi sederhana berbasis OOP dengan Python).			
		KK.4	Mampu mengintegrasikan konsep OOP ke dalam pengembangan aplikasi modular dan reusable			
<b>Penunjang</b>	PJ-01	Adaptif Terhadap Teknologi Terbaru				
	<b>Capaian Pembelajaran Mata Kuliah (CPMK)</b>					

CPMK	1	Mampu memahami konsep dasar OOP dan menerapkannya dalam pembuatan kelas dan objek sederhana (SKKNI J.620100.019.02).)						
	2	Mampu menerapkan prinsip enkapsulasi dan access modifier dalam pengembangan kelas yang aman..						
	3	Mampu menerapkan inheritance dan polimorfisme dalam membangun hierarki kelas yang reusable.						
	4	Mampu mengimplementasikan abstract class dan method untuk mendesain sistem yang modular						
	5	Mampu membangun aplikasi sederhana berbasis OOP yang menerapkan seluruh konsep OOP.).						
	6	Mampu melakukan presentasi dan dokumentasi proyek aplikasi OOP yang telah dikembangkan.						
<b>Kemampuan akhir tiap tahapan belajar (Sub-CPMK)</b>								
Sub-CPMK	1	Kontrak Kuliah & Konsep Dasar OOP						
	2	Class & Object dalam pemrograman OOP						
	3	Encapsulation & Access Modifier						
	4	Inheritance, Class Object dan Superclass						
	5	Polymorphism & Method Overriding dan Abstract Class & Method						
	6	Aplikasi OOP Sederhana						
	7	State Management dalam OOP						
	8	Integrasi dengan Modularitas						
<b>Korelasi CPMK terhadap CPL</b>								
<b>Capaian Pembelajaran Mata Kuliah (CPMK)</b>		<b>CPL yang Didukung</b>						
Kode CPMK	Deskripsi CPMK	S9	S11	P3	KU1	KK1	KK4	PJ01
CPMK 1,	Mampu memahami konsep dasar OOP dan menerapkannya dalam pembuatan kelas dan objek sederhana	✓	✓	✓	✓	✓	✓	✓
CPMK 2	Mampu menerapkan prinsip enkapsulasi dan access modifier dalam pengembangan kelas yang aman	✓	✓	✓	✓	✓	✓	✓
CPMK 3	Mampu menerapkan inheritance dan polimorfisme dalam membangun hierarki kelas yang reusable.	✓	✓	✓	✓	✓	✓	✓
CPMK 4	Mampu mengimplementasikan abstract class dan method untuk mendesain sistem yang modular	✓	✓	✓	✓	✓	✓	✓

	CPMK 5	Mampu membangun aplikasi sederhana berbasis OOP yang menerapkan seluruh konsep OOP.	✓	✓	✓	✓	✓	✓	✓
	CPMK 6	Mampu melakukan presentasi dan dokumentasi proyek aplikasi OOP yang telah dikembangkan	✓	✓	✓	✓	✓	✓	✓
Deskripsi Singkat MK	Mata kuliah Pemrograman Berorientasi Objek (4 SKS) Mata kuliah ini membekali mahasiswa kemampuan merancang dan mengembangkan perangkat lunak menggunakan paradigma <b>Object Oriented Programming (OOP)</b> . Mahasiswa mempelajari konsep <b>class, object, encapsulation, inheritance, polymorphism, abstraction</b> , serta implementasi aplikasi berbasis objek menggunakan Python. Mata kuliah menekankan pendekatan <b>project-based learning</b> hingga menghasilkan <b>Aplikasi berbasis OOP</b>								
Bahan Kajian : Materi Pelajaran	<ol style="list-style-type: none"> <li>1. Kontrak Kuliah &amp; Konsep Dasar OOP</li> <li>2. Class &amp; Object dalam pemrograman OOP</li> <li>3. Encapsulation &amp; Access Modifier</li> <li>4. Inheritance, Class Object dan Superclass</li> <li>5. Polymorphism &amp; Method Overriding dan Abstract Class &amp; Method</li> <li>6. Aplikasi OOP Sederhana</li> <li>7. State Management dalam OOP</li> <li>8. Integrasi dengan Modularitas</li> </ol>								
Pustaka	Buku Utama :								
	<ol style="list-style-type: none"> <li>1. Junaidi Surya (2024), "Pemrograman Berorientasi Objek Python" Penerbit Sonpedia-Jambi Indonesia</li> <li>2. Abdul Kadir (2021), "Dasar Pemrograman Python 3" Penerbit Andi – Yogyakarta</li> <li>3. Nana Suarna (2025), "Belajar Algoritma dan Pemrograman Dasar Menggunakan Python", Penerbit Yrama Widya</li> </ol>								
	Pendukung :								
	<ol style="list-style-type: none"> <li>1. Modul Praktikum Perkuliahan Pemrograman Dasar</li> <li>2. <a href="https://pynative.com/python/">https://pynative.com/python/</a></li> <li>3. <a href="https://www.w3schools.com/python/default.asp">https://www.w3schools.com/python/default.asp</a></li> <li>4. <a href="https://www.youtube.com/watch?v=iA8lLwmtKQM&amp;list=PLZS-MHyEIro7cgStrKAMhgnOT66z2qKz1">https://www.youtube.com/watch?v=iA8lLwmtKQM&amp;list=PLZS-MHyEIro7cgStrKAMhgnOT66z2qKz1</a></li> </ol>								
Dosen Pengampu	1. Junaidi Surya, M.Kom, 2. Laylin Puad, M.Kom, Fattachul Huda Aminuddin, S.Kom, M.PdT								
Mata Kuliah Prasyarat	Pemrograman Berbasis Platform Nilai > D								

Rencana Kegiatan Pembelajaran Mingguan							
Minggu Ke -	Kemampuan akhir tiap tahapan (Sub CPMK)	Indikator	Kriteria & Teknik	Bentuk Pembelajaran; Metode Pembelajaran; Penugasan Mahasiswa (Estimasi Waktu)		Materi Pembelajaran (Pustaka)	Bobot Penilaian (%)
				Luring	Daring		
1	Kontrak kuliah & konsep dasar OOP	Pengantar OOP, perbedaan prosedural vs OOP, kelebihan & kekurangan OOP	Kuliah, diskusi, instalasi Python & PyCharm/VSCode	<b>Kuliah:</b> 1×45 menit • Pengenalan mata kuliah • Demo Streamlit pertama <b>Praktikum:</b> 2×45 menit • Instalasi tools • Hello World app <b>Diskusi:</b> 1×50 menit • Aturan akademik	-	<b>Pustaka:</b> • Modul 1: Pengantar • <a href="https://docs.streamlit.io/">https://docs.streamlit.io/</a> • <a href="https://python.org">Python.org</a> dokumentasi on	
2	Class & Object	Sintaks kelas, atribut, metode, constructor ( <code>__init__</code> )	Praktikum: membuat kelas Mahasiswa, Buku, dll.	<b>Kuliah:</b> 1×45 menit • Variabel & tipe data • String operations <b>Praktikum:</b> 2×45 menit • Input/output di Streamlit • <code>st.write()</code> & <code>st.text()</code> <b>Latihan:</b> 1×50 menit • Mini project: Biodata app		<b>Pustaka:</b> • Buku 1: Bab 2-3 • <a href="https://w3schools.com/python">w3schools.com/python</a> • PyNative exercises	
3	Encapsulation & Access Modifier	Public, private, protected, getter/setter, name mangling	Praktikum: aplikasi dompet digital dengan atribut private	<b>Kuliah:</b> 1×45 menit • Operator Python • Percabangan if-elif-else <b>Praktikum:</b> 2×45 menit • Conditional rendering Streamlit • Match-case untuk navigation • Build calculator app			
4	Inheritance & Superclass	Single, multilevel, hierarchical inheritance, <code>super()</code>	Praktikum: sistem pewarisan Kendaraan → Mobil → MobilSport	<b>Kuliah:</b> 1×45 menit • List, tuple, dictionary • List comprehension <b>Lab:</b> 2×50 menit • <code>st.selectbox()</code> dengan list • <code>st.dataframe()</code> display <b>Review:</b> 1×50 menit • Data structure best practices			
5	Polymorphism & Method Overriding	Overriding, overloading (konseptual), polimorfisme dengan inheritance	Praktikum: sistem pembayaran dengan metode berbeda	• Function definition • Lambda functions <b>Praktikum:</b> 2×45 menit • File upload/download • CSV/JSON processing <b>Project:</b> 1×50 menit • Modular app design			

6-7	Abstract Class & Method dan Integrasi Konsep	abc.ABC, @abstractmethod, kontrak antar kelas	Praktikum: sistem bentuk geometri dengan abstract class	<b>Kuliah: 1×45</b> menit • Streamlit layout system • Columns, containers <b>Praktikum: 5×45</b> menit			
8	<b>UTS (Ujian Tengah Semester)</b>						
9-10	Aplikasi OOP Sederhana	Perancangan sistem manajemen karyawan / toko	Praktikum: analisis kasus, desain kelas	<b>Kuliah: 1×45</b> menit • Multi-page architecture • Navigation patterns <b>Praktikum: 5×45</b> menit • Page organization • Shared state across pages <b>Architecture: 1×50</b> menit • App structure design			
11-12	State Management dalam OOP	Session state, persistent object, serialisasi data sederhana	Praktikum: simpan/load objek ke file JSON	<b>Kuliah: 1×45</b> menit • State management concepts • Session state deep dive <b>Praktikum : 2×45</b> menit • Form state persistence <b>Optimization: 1×50</b> menit • State performance tips		<b>Pustaka:</b> • Streamlit Session State • State management patterns • Performance guidelines	
13-14	Integrasi dengan Modularitas	Pembagian modul, package, __init__.py, import antar modul	Integrasi dengan Modularitas	<b>Kuliah: 1×45</b> menit • Database concepts • SQL with Python <b>Lab: 2×50</b> menit • SQLite integration • CRUD operations in UI <b>Design: 1×50</b> menit • Database schema design		<b>Pustaka:</b> • SQLite documentation • SQLAlchemy guide • Database security best practices	
15	Persiapan Proyek Akhir	Penyusunan proposal, diagram kelas, pembagian fitur	Workshop proyek, konsultasi desain				
16	<b>UJIAN AKHIR SEMESTER (UAS) &amp; PRESENTASI</b>	Integrasi Matakuliah ke Project final.					

### Strategi Penilaian (Assessment)

1. **Kehadiran & Sikap: 10%**
2. **Tugas Mandiri/Terstruktur/Kuis: 20%**
3. **UTS/Praktikum : 30%**
4. **UAS (Final Project): 40%**

