





PERGURUAN TINGGI : UNIVERSITAS DIAN NUSWANTORO
 FAKULTAS : ILMU KOMPUTER
 PROGRAM STUDI : SARJANA TEKNIK INFORMATIKA

RENCANA PEMBELAJARAN SEMESTER (RPS)

Mata Kuliah	Kode	Rumpun Mata Kuliah	SKS	Semester	Tanggal Penyusunan
Dasar Pemrograman	A11.54402	Wajib Program Studi	4	4	24 September 2019
Otorisasi	Dosen Pengembang RPS		Koordinator RMK		Ketua Program Studi
	 Ajib Susanto, M.Kom.		 Ajib Susanto, M.Kom		Heru Agus Santoso, Ph.D.
Capaian Pembelajaran (CP)	Capaian Pembelajaran Program Studi				
	S9	Menunjukkan sikap bertanggungjawab atas pekerjaan di bidang keahliannya secara mandiri			
	P1	Menguasai konsep teoritis bidang pengetahuan Ilmu Komputer /Informatika secara umum dan konsep teoritis bagian khusus dalam bidang pengetahuan tersebut secara mendalam, serta mampu memformulasikan penyelesaian masalah berorientasi obyek.			
	KU2	Mampu menunjukkan kinerja mandiri, bermutu, dan terukur.			
	KK5	Menguasai konsep-konsep bahasa pemrograman berorientasi obyek, serta mampu melakukan analisa, merancang dan mengimplementasikan pemrograman berorientasi obyek untuk suatu masalah.			
	KK7	Menentukan dan menerapkan pendekatan berorientasi obyek yang sesuai dengan problem yang dihadapi, memecahkan masalah menjadi sebuah desain konsep pemrograman berorientasi obyek dan kemudian mengimplementasikannya menjadi sebuah program komputer.			
	Capaian Pembelajaran Mata Kuliah				
	M1	<ul style="list-style-type: none">Memahami latar belakang konsep pemrograman berorientasi obyek dan perbandingannya dengan pemrograman proceduralMemahami konsep dasar pemrograman berorientasi obyekMemperkenalkan dasar-dasar dan aturan pemrograman di Java sebagai basis pemrograman OOP			
	M2	<ul style="list-style-type: none">Memahami ObjectMemahami Class			
	M3	<ul style="list-style-type: none">Memahami interaksi antar object			
	M4 – M5	<ul style="list-style-type: none">Memahami CollectionMemahami arraysMemahami iteratorMemahami loop			

	M6	<ul style="list-style-type: none"> • Memahami Package • Memahami Library
	M7	<ul style="list-style-type: none"> • Memahami konsep Inheritance/ pewarisan
	M8	<ul style="list-style-type: none"> • Memahami polymorphism
	M9	<ul style="list-style-type: none"> • Memahami jenis error dan sifatnya • Mampu melakukan proses penanganan error
	M10	<ul style="list-style-type: none"> • Memahami konsep abstract • Memahami konsep interface
	M12	<ul style="list-style-type: none"> • Memahami GUI
	M13 - M14	<ul style="list-style-type: none"> • Memahami interface component • Memahami OOP dengan database
	M15	<ul style="list-style-type: none"> • Memahami Konsep Testing • Memahami dokumentasi
Deskripsi Singkat Mata Kuliah	<p>Dalam perkuliahan ini mengajarkan konsep pemrograman berorientasi obyek disertai contoh penerapan konsep pemrograman berorientasi obyek. Pada kuliah ini mahasiswa diharapkan dapat merancang dan mengimplementasikan permasalahan ke dalam konsep pemrograman berorientasi obyek dengan diberikan tugas-tugas pemrograman dan tugas proyek</p>	
Materi Pembelajaran/ Pokok Bahasan	<ol style="list-style-type: none"> 1. Pengenalan dan Dasar Pemrograman PBO; penjelasan Definisi PBO, Teknologi Java, Perbandingan pemrograman prosedural dan pemrograman berorientasi obyek, Struktur Java, Identifier, Kata Kunci, Identifier, Type Data, operator, inialisasi dan nilai default. 2. Object & Class; Membuat Class, Menginstantiasi Object, Attribute, Methods, Constructor, Parameter, assignment and conditional statement. 3. Interaksi antar Object; Manipulasi attribute dan method melalui class lain, Hak akses, Static class, Overloading 4. Koleksi Object; Collection, Arrays, Iterator, Loop 5. Package & Library; Package, Library (Math, Date, Time, dll). 6. Inheritance/ Pewarisan; Konsep inheritance/pewarisan, Deklarasi inheritance, Super class. 7. Polymorphism; Konsep polymorphism, Overloading methods, Overriding methods. 8. Exception Handling; Exception Categories, Common exception, Try, catch, throw, throws, Membuat Eksepsi sendiri. 9. Abstract & Interface; Abstract, Implement abstract, Interface. 10. GUI (Graphic User Interface); GUI Component, Event Handling, GUI Layout. 11. GUI & Database; Interface Component, Koneksi Database, Aplikasi OOP dengan Database. 12. Testing & Dokumentasi; Debugging, Testing, Unit Testing, Automation test, dokumentasi 	
Pustaka	Utama :	
	<ol style="list-style-type: none"> 1. David J. Barnes & Michael Kölling, Object First With Java, Fifth edition, Prentice Hall / Pearson Education, 2012. 2. Luliana Cosmina, Java for Absolute Beginners- Learn to Program the Fundamentals the Java 9+ Way, APRESS, 2018. 3. Roger Codenhead, Java in 21 Days, Sams Teach Yourself (Covering Java 8), 7th Edition, Pearson Education Inc, 2016. 4. Oracle, The Java™ Tutorial, http://docs.oracle.com/javase/tutorial/java/nutsandbolts/, 1995-2019. 	

	5. Oracle, The Java Tutorial Bundle, http://www.oracle.com/technetwork/java/javase/downloads/java-se-7-tutorial-2012-02-28-1536013.html , 2017.	
	Pendukung :	
	1. Mala Gupta, OCA Java SE 8 Programmer I Certification Guide, Manning Publication Co, 2017 2. Kathy Sierra, Bert Bates, Oca Java SE 8 Programmer I Exam Guide, McGraw-Hill Education, 2017	
Media Pembelajaran	Perangkat Lunak :	Perangkat Keras :
	Java SE 8, NetBean, Notepad++, Vcode	Proyektor, 1 set komputer yang memadai
Tim Teaching	Tim Pengampu Mata Kuliah Pemrograman Berorientasi Obyek	
Mata Kuliah Syarat	-	

Mggu	Sub CP MK (sebagai kemampuan akhir yang diharapkan)	Indikator	Kriteria & Bentuk Penilaian	Metode Pembelajaran [Estimasi Waktu] Pengalaman Belajar	Materi pembelajaran	Bobot Penilaian
(1)	(2)	(3)	(4)	(5)	(6)	(7)
1	<p>Setelah mengikuti kuliah ini mahasiswa dapat:</p> <ul style="list-style-type: none"> Memahami latar belakang konsep pemrograman berorientasi obyek dan perbandingannya dengan pemrograman procedural Memahami konsep dasar pemrograman berorientasi obyek Memperkenalkan dasar-dasar dan aturan pemrograman di Java sebagai basis pemrograman OOP 	<p>Teori Mahasiswa mampu:</p> <ol style="list-style-type: none"> Menyebutkan cakupan materi dan pokok bahasan Memahami pokok bahasan yang diajarkan <p>Praktek Mahasiswa dapat:</p> <ol style="list-style-type: none"> Mengenal tools yaitu bahasa Java yang akan digunakan sebagai salah satu tools yang digunakan sebagai pengekskusi coding. Menuliskan code sederhana dan mengeksekusinya. Mengetahui beberapa error message yang sering ditemui. 	<p>Teori Kriteria : Ketepatan dan penguasaan Bentuk non test : Tulisan makalah</p> <p>Praktek Kriteria : Ketepatan dan penguasaan Bentuk non test :</p> <ol style="list-style-type: none"> Pemahaman dan pengenalan tools. Pemahaman Type data, atribut Pemahaman komputasi di Java 	<p>Teori</p> <ul style="list-style-type: none"> Kuliah & Diskusi [TM:2x50'] Tugas-1: Studi kasus [BT+BM: (1+1)x(2x50')] <p>Praktek</p> <ul style="list-style-type: none"> Kuliah & Diskusi [TM:2x50'] Tugas-1: Instalasi Tools yaitu Bahasa Java. Pengecekan error message pada penulisan code sederhana. [BT+BM: (1+1)x(2x50')] 	<p>Teori</p> <ol style="list-style-type: none"> Penjelasan secara umum tentang mata kuliah/kontrak belajar Penjelasan tentang silabus Penjelasan tentang cara penilaian Sekilas tentang sejarah pemrograman Java Menjelaskan PBO Menjelaskan Teknologi Java (J2SE, J2ME, J2EE, Java Card) Menjelaskan konsep pemrograman procedural dan OOP Menjelaskan Struktur Java, Identifier, Kata Kunci, Identifier, Type Data, operator, inisialisasi dan nilai default <p>Praktek</p> <ol style="list-style-type: none"> Pengetahuan dasar pemrograman berorientasi obyek. Pengenalan tools bahasa Java. Penulisan code sederhana dalam bahasa Java. Pemahaman tentang error message dalam penulisan code program bahasa Java. 	
2	<p>Setelah mengikuti kuliah ini mahasiswa dapat:</p> <ul style="list-style-type: none"> Memahami object Memahami Class. 	<p>Teori Mahasiswa mampu :</p> <ol style="list-style-type: none"> Memahami Object Memahami Class. <p>Praktek</p>	<p>Teori Kriteria : Ketepatan dan penguasaan Bentuk non test :</p>	<p>Teori</p> <ul style="list-style-type: none"> Kuliah & Diskusi [TM:2x50'] Tugas-2: Studi kasus [BT+BM: (1+1)x(2x50')] 	<p>Teori</p> <ol style="list-style-type: none"> Pengenalan Type Data Variabel Printf Scanf 	

Mggu	Sub CP MK (sebagai kemampuan akhir yang diharapkan)	Indikator	Kriteria & Bentuk Penilaian	Metode Pembelajaran [Estimasi Waktu] Pengalaman Belajar	Materi pembelajaran	Bobot Penilaian
(1)	(2)	(3)	(4)	(5)	(6)	(7)
	<ul style="list-style-type: none"> Membuat kode sederhana dengan konsep class & Object dengan benar 	<p>Mahasiswa dapat:</p> <ol style="list-style-type: none"> Menjelaskan class & object yang digunakan dalam pembuatan kode program Menggunakan type-type data primitif, variable dan method dalam menyelesaikan masalah pada pembuatan program perhitungan sederhana dan pembuatan object-nya Membedakan penggunaan type-type data primitif, atribut dan method untuk studi kasus yang sederhana. 	<p>Tulisan makalah</p> <p>Praktek Kriteria : Ketepatan dan penguasaan Bentuk non test :</p> <ol style="list-style-type: none"> Pengenalan class dan object. Laporan hasil praktek 	<p>Praktek</p> <ul style="list-style-type: none"> Kuliah & Diskusi [TM:2x50'] Tugas-2: Studi kasus penerapan class dan object dalam code program serta pemilihan type data yang tepat dalam penggunaan atribut & method. [BT+BM: (1+1)x(2x50')] 	<p>e. Komentar</p> <p>Praktek</p> <ol style="list-style-type: none"> Mengenalkan pendefenisi class & object dalam bentuk kode program. Mengenalkan atribut & method. Cara penginputan atribut dan memanggil method serta cara mengoutputkannya. 	
3	<p>Setelah mengikuti kuliah ini mahasiswa dapat :</p> <ul style="list-style-type: none"> memahami interaksi antar object Memahami konsep hak akses atribut & method Memahami persoalan-persoalan yang diberikan, menganalisa dan mampu memberikan problem solving (pemecahan masalah). 	<p>Teori</p> <ol style="list-style-type: none"> Menjelaskan cara manipulasi attribute dan method melalui class lain bagaimana mengatur hak akses Menjelaskan static class Menjelaskan overloading Contoh program dan pembahasan <p>Praktek Mahasiswa dapat:</p> <ol style="list-style-type: none"> Menyelesaikan perhitungan matematik menggunakan operator- 	<p>Teori Kriteria : Ketepatan dan penguasaan Bentuk non test : Tulisan makalah</p> <p>Praktek Kriteria : Ketepatan dan penguasaan Bentuk non test :</p> <ol style="list-style-type: none"> Pengenalan method dengan 	<p>Teori</p> <ul style="list-style-type: none"> Kuliah & Diskusi [TM:2x50'] Tugas-3: Studi kasus [BT+BM: (1+1)x(2x50')] <p>Praktek</p> <ul style="list-style-type: none"> Kuliah & Diskusi [TM:2x50'] Tugas-3: Studi kasus method dengan parameter, overloading dalam code program serta pemecahan kasus sederhana. 	<p>Teori</p> <ol style="list-style-type: none"> Manipulasi attribute dan method melalui class lain Hak akses Static class Overloading <p>Praktek</p> <ol style="list-style-type: none"> Pengenalan method dengan paramater & cara memanggilnya. Pengenalan overloading Pengenalan interaksi antar object Pengenalan keyword static. 	

Mggu	Sub CP MK (sebagai kemampuan akhir yang diharapkan)	Indikator	Kriteria & Bentuk Penilaian	Metode Pembelajaran [Estimasi Waktu] Pengalaman Belajar	Materi pembelajaran	Bobot Penilaian
(1)	(2)	(3)	(4)	(5)	(6)	(7)
		operator dan method dengan paramater. 2. Menyelesaikan permasalahan dalam bentuk studi kasus dan menterjemahkannya dalam bentuk kode program.	parameter, hak akses atribut & method b. Pengenalan static class, static atribut, static method c. Pengenalan overloading d. Laporan hasil praktek	[BT+BM: (1+1)x(2x50')]		
4-5	Setelah mengikuti kuliah ini mahasiswa dapat: <ul style="list-style-type: none"> Memahami Collection Memahami arrays Memahami iterator Memahami loop 	<p>Teori Mahasiswa mampu:</p> <ol style="list-style-type: none"> Menjelaskan tentang koleksi object Memecahkan masalah dengan menggunakan koleksi object. <p>Praktek Mahasiswa dapat memecahkan masalah dengan koleksi object dan menterjemahkannya dalam bentuk kode program.</p>	<p>Teori Kriteria : Ketepatan dan penguasaan Bentuk non test : Tulisan makalah</p> <p>Praktek Kriteria : Ketepatan dan penguasaan Bentuk non test :</p> <ol style="list-style-type: none"> Pengenalan koleksi object. Laporan hasil praktek 	<p>Teori</p> <ul style="list-style-type: none"> Kuliah & Diskusi [TM:2x50'] Tugas-4: Studi kasus [BT+BM: (1+1)x(2x50')] Kuliah & Diskusi [TM:2x50'] Tugas-5: Studi kasus [BT+BM: (1+1)x(2x50')] <p>Praktek</p> <ul style="list-style-type: none"> Kuliah & Diskusi [TM:2x50'] Tugas-4: Studi kasus tentang pembuatan code program dengan menggunakan berbagai koleksi object. [BT+BM: (1+1)x(2x50')] 	<p>Teori</p> <ol style="list-style-type: none"> Pengertian collection Pengertian Arrays Pengertian Iterator Pengertian Loop <p>Praktek</p> <ol style="list-style-type: none"> Pengenalan collection Pengenalan Arrays Pengenalan Iterator Pengenalan percabangan dan Looping Penulisan kode program dengan koleksi object 	
6	Setelah mengikuti kuliah ini mahasiswa dapat :	<p>Teori</p> <ol style="list-style-type: none"> Menjelaskan tentang konsep pacakage 	<p>Teori</p>	<p>Teori</p> <ul style="list-style-type: none"> Kuliah & Diskusi [TM:2x50'] 	<p>Teori</p> <ol style="list-style-type: none"> Package Library 	

Minggu	Sub CP MK (sebagai kemampuan akhir yang diharapkan)	Indikator	Kriteria & Bentuk Penilaian	Metode Pembelajaran [Estimasi Waktu] Pengalaman Belajar	Materi pembelajaran	Bobot Penilaian
(1)	(2)	(3)	(4)	(5)	(6)	(7)
	<ul style="list-style-type: none"> Menjelaskan konsep Package & Library Memahami pembuatan package dalam pemecahan masalah. Memahami penggunaan library dalam menyelesaikan masalah yang diberikan 	2. Menjelaskan tentang library (Date, Time, Math, dll) 3. Contoh program dan pembahasan Praktek Mahasiswa dapat: 1. Memecahkan masalah analisa kasus dan menterjemahkannya dalam bentuk kode program 2. Analisa kasus yang diselesaikan dengan menggunakan beberapa package dan diselesaikan dengan membuat paket JAR.	Kriteria : Ketepatan dan penguasaan Bentuk non test : Tulisan makalah Praktek Kriteria : Ketepatan dan penguasaan Bentuk non test : a. Pengenalan package & library. b. Laporan hasil praktek	<ul style="list-style-type: none"> Tugas-6: Studi kasus [BT+BM: (1+1)x(2x50')] Praktek <ul style="list-style-type: none"> Kuliah & Diskusi [TM:2x(2x50')] Tugas-6: Studi kasus analisa menggunakan package dan library. [BT+BM: 2x(1+1)x(2x50')] 	c. File JAR Praktek <ol style="list-style-type: none"> Pengenalan package. Pengenalan library. Pembuatan file JAR. 	
7	Setelah mengikuti kuliah ini mahasiswa dapat: <ul style="list-style-type: none"> Memahami konsep inheritance atau pewarisan. Memahami super class dan sub class 	Teori <ol style="list-style-type: none"> Menjelaskan tentang konsep inheritance Menjelaskan super class Contoh program dan pembahasan Praktek <ol style="list-style-type: none"> Mahasiswa dapat memecahkan masalah dengan konsep inheritance dan menterjemahkannya dalam bentuk kode program. 	Teori Kriteria : Ketepatan dan penguasaan Bentuk non test : Tulisan makalah Praktek Kriteria : Ketepatan dan penguasaan Bentuk non test : a. Pengenalan inheritance. b. Laporan hasil praktek	Teori <ul style="list-style-type: none"> Kuliah & Diskusi [TM:2x50'] Tugas-7: Studi kasus [BT+BM: (1+1)x(2x50')] Praktek <ul style="list-style-type: none"> Kuliah & Diskusi [TM:2x50'] Tugas-6: Studi kasus analisa menggunakan konsep inheritance. [BT+BM: (1+1)x(2x50')] 	Teori <ol style="list-style-type: none"> Inheritance/pewarisan Super class dan sub class Praktek <ol style="list-style-type: none"> Pengertian inheritance, super class dan sub class. Latihan penggunaan inheritance. 	

Minggu	Sub CP MK (sebagai kemampuan akhir yang diharapkan)	Indikator	Kriteria & Bentuk Penilaian	Metode Pembelajaran [Estimasi Waktu] Pengalaman Belajar	Materi pembelajaran	Bobot Penilaian
(1)	(2)	(3)	(4)	(5)	(6)	(7)
		2. Analisa kasus yang diselesaikan dengan menggunakan konsep inheritance.				
8	UJIAN TENGAH SEMESTER					
9	Setelah mengikuti kuliah ini mahasiswa dapat: <ul style="list-style-type: none"> memahami dan menjelaskan konsep polymorphism Memahami overloading dan overriding 	Teori <ol style="list-style-type: none"> Menjelaskan pengertian polymorphism Memecahkan masalah yang diberikan dengan menggunakan konsep polymorphism, overloading dan overriding. Praktek <p>Mahasiswa dapat memecahkan masalah yang diberikan dengan menggunakan konsep polymorphism, overloading dan overriding.</p>	Teori Kriteria : Ketepatan dan penguasaan Bentuk non test : Tulisan makalah Kriteria : Ketepatan dan penguasaan Bentuk non test : Tulisan makalah Praktek Kriteria : Ketepatan dan penguasaan Bentuk non test : <ol style="list-style-type: none"> Pengenalan konsep polymorphism, overloading dan overriding. Laporan hasil praktek 	Teori <ul style="list-style-type: none"> Kuliah & Diskusi [TM:2x50'] Tugas-9: Studi kasus [BT+BM: (1+1)x(2x50')] Praktek <ul style="list-style-type: none"> Kuliah & Diskusi [TM:2x(2x50')] Tugas-9: Studi kasus pembuatan code program dengan menggunakan konsep polymorphism, overloading dan overriding [BT+BM: 2x(1+1)x(2x50')] 	Teori <ol style="list-style-type: none"> Polymorphism Overloading Overriding Praktek <ol style="list-style-type: none"> Pengertian konsep polymorphism, overloading dan overriding Overloading. Overriding 	
10	Setelah mengikuti kuliah ini mahasiswa dapat:	Teori <ol style="list-style-type: none"> Mahasiswa dapat Mengetahui jenis error dan sifatnya 	Teori Kriteria : Ketepatan dan penguasaan	Teori <ul style="list-style-type: none"> Kuliah & Diskusi [TM:2x50'] Tugas-10: Studi kasus 	Teori <ol style="list-style-type: none"> Kategori eksepsi Eksepsi secara umum 	

Minggu	Sub CP MK (sebagai kemampuan akhir yang diharapkan)	Indikator	Kriteria & Bentuk Penilaian	Metode Pembelajaran [Estimasi Waktu] Pengalaman Belajar	Materi pembelajaran	Bobot Penilaian
(1)	(2)	(3)	(4)	(5)	(6)	(7)
	<ul style="list-style-type: none"> memahami dan menjelaskan konsep Eksepsi Memahami berbagai penggunaan penanganan error. 	2. Mampu melakukan proses penanganan error Praktek Mahasiswa dapat memecahkan masalah yang diberikan dengan menggunakan berbagai jenis eksepsi dan error dengan menterjemahkannya dalam bentuk kode program	Bentuk non test : Tulisan makalah Praktek Kriteria : Ketepatan dan penguasaan Bentuk non test : a. Pengenalan eksepsi dan error. b. Laporan hasil praktek	[BT+BM: (1+1)x(2x50')] Praktek <ul style="list-style-type: none"> Kuliah & Diskusi [TM:2x50'] Tugas-10: Studi kasus pembuatan code program dengan menggunakan konsep eksepsi dan error handling [BT+BM: (1+1)x(2x50')] 	c. Menjelaskan try, catch, throw, throws d. Menjelaskan cara membuat eksepsi sendiri Praktek <ul style="list-style-type: none"> Eksepsi secara umum Eksepsi try, catch, throw, throws. Eksepsi buatan sendiri 	
11	Setelah mengikuti kuliah ini mahasiswa dapat: <ul style="list-style-type: none"> Menjelaskan Abstract Menjelaskan Interface. 	Teori <ol style="list-style-type: none"> Menjelaskan pengertian dan konsep abstract Menjelaskan pengertian dan konsep interface Menjelaskan perbedaan abstract dan interface Memecahkan masalah yang diberikan dengan menggunakan abstract dan interface. Praktek <ol style="list-style-type: none"> Memecahkan masalah kasus abstract, interface dan menterjemahkannya dalam bentuk kode program. Kasus yang diselesaikan dengan menggunakan abstract dan interface. 	Teori Kriteria : Ketepatan dan penguasaan Bentuk non test : Tulisan makalah Kriteria : Ketepatan dan penguasaan Bentuk non test : Tulisan makalah Praktek Kriteria : Ketepatan dan penguasaan Bentuk non test : c. Pengenalan abstract dan interface. d. Laporan hasil praktek	Teori <ul style="list-style-type: none"> Kuliah & Diskusi [TM:2x50'] Tugas-11: Studi kasus [BT+BM: (1+1)x(2x50')] Praktek <ul style="list-style-type: none"> Kuliah & Diskusi [TM:2x(2x50')] Tugas-11: Studi kasus pembuatan code program dengan menggunakan abstract dan interface. [BT+BM: 2x(1+1)x(2x50')] 	Teori <ol style="list-style-type: none"> abstract interface Praktek <ol style="list-style-type: none"> Pengertian abstract dan interface. Latihan penggunaan abstract class. Latihan penggunaan interface. Latihan gabungan penggunaan abstarct dan interface 	

Minggu	Sub CP MK (sebagai kemampuan akhir yang diharapkan)	Indikator	Kriteria & Bentuk Penilaian	Metode Pembelajaran [Estimasi Waktu] Pengalaman Belajar	Materi pembelajaran	Bobot Penilaian
(1)	(2)	(3)	(4)	(5)	(6)	(7)
12	Setelah mengikuti kuliah ini mahasiswa dapat: <ul style="list-style-type: none"> memahami dan menjelaskan konsep GUI. Memahami berbagai penggunaan GUI. 	Teori <ol style="list-style-type: none"> Memahami GUI. Mampu menggunakan GUI untuk berbagai formulir. Praktek <ol style="list-style-type: none"> Memecahkan masalah berbagai formulir dan menterjemahkannya dalam bentuk kode program. Kasus yang diselesaikan dengan menggunakan GUI. 	Teori Kriteria : Ketepatan dan penguasaan Bentuk non test : Tulisan makalah Praktek Kriteria : Ketepatan dan penguasaan Bentuk non test : <ol style="list-style-type: none"> Pengenalan GUI. Laporan hasil praktek 	Teori <ul style="list-style-type: none"> Kuliah & Diskusi [TM:2x50'] Tugas-12: Studi kasus [BT+BM: (1+1)x(2x50')] Praktek <ul style="list-style-type: none"> Kuliah & Diskusi [TM:2x50'] Tugas-12: Studi kasus pembuatan code program dengan menggunakan GUI. [BT+BM: (1+1)x(2x50')] 	Teori <ol style="list-style-type: none"> Menjelaskan GUI Component Menjelaskan Event Handling Menjelaskan GUI Layout Contoh program dan pembahasan Praktek <ol style="list-style-type: none"> Pengenalan GUI. Membuat Form dengan GUI. Penanganan event handling 	
13-14	Setelah mengikuti kuliah ini mahasiswa dapat: <ul style="list-style-type: none"> memahami dan menjelaskan interface kompone . Memahami OOP dengan database 	Teori <ol style="list-style-type: none"> Mampu mengkombinasikan GUI dan database Dapat menggunakan GUI dan database dalam satu program untuk menyelesaikan masalah Praktek <ol style="list-style-type: none"> Memecahkan masalah dengan GUI dan database serta menterjemahkannya dalam bentuk kode program. Kasus yang diselesaikan dengan menggunakan 	Teori Kriteria : Ketepatan dan penguasaan Bentuk non test : Tulisan makalah Praktek Kriteria : Ketepatan dan penguasaan Bentuk non test : <ol style="list-style-type: none"> Pengenalan GUI dan database. Laporan hasil praktek 	Teori <ul style="list-style-type: none"> Kuliah & Diskusi [TM:2x50'] Tugas-13 : Studi kasus [BT+BM: (1+1)x(2x50')] Kuliah & Diskusi [TM:2x50'] Tugas-14 : Studi kasus [BT+BM: (1+1)x(2x50')] Praktek <ul style="list-style-type: none"> Kuliah & Diskusi [TM:2x50'] Tugas-13&14: Studi kasus pembuatan code program dengan menggunakan 	Teori <ol style="list-style-type: none"> GUI Component. Event Handling GUI Layout Koneksi dan akses database Laporan Praktek <ol style="list-style-type: none"> Membuat koneksi ke database. Membuat form master data. Membuat form transaksi Membuat laporan 	

Minggu	Sub CP MK (sebagai kemampuan akhir yang diharapkan)	Indikator	Kriteria & Bentuk Penilaian	Metode Pembelajaran [Estimasi Waktu] Pengalaman Belajar	Materi pembelajaran	Bobot Penilaian
(1)	(2)	(3)	(4)	(5)	(6)	(7)
		kombinasi GUI dan database.		kombinasi GUI dan database. [BT+BM: (1+1)x(2x50')]		
15	Setelah mengikuti kuliah ini mahasiswa dapat: 1. Memahami dan menjelaskan debugging dan testing 2. Memahami unit testing 3. Memahami dan menjelaskan dokumentasi	Teori <ol style="list-style-type: none"> 1. Mampu menjelaskan debugging dan testing 2. Mampu menjelaskan unit testing 3. Dapat membuat dokumentasi program Praktek <ol style="list-style-type: none"> 1. Melakukan pengujian unit testing. 2. Membuat dokumentasi program. 	Teori Kriteria : Ketepatan dan penguasaan Bentuk non test : Tulisan makalah Praktek Kriteria : Ketepatan dan penguasaan Bentuk non test : c. Pengenalan unit testing dan dokumentasi. d. Laporan hasil praktek	Teori <ul style="list-style-type: none"> • Kuliah & Diskusi [TM:2x50'] • Tugas-15 : Studi kasus [BT+BM: (1+1)x(2x50')] Praktek <ul style="list-style-type: none"> • Kuliah & Diskusi [TM:2x50'] • Tugas-15: Studi kasus melakukan testing dan membuat dokumentasi. [BT+BM: (1+1)x(2x50')] 	Teori <ol style="list-style-type: none"> a. Debuging. b. Testing c. Unit testing d. Automation test Praktek <ol style="list-style-type: none"> a. Melakukan debugging dan testing program. b. Melakukan unit testing. c. Membuat dokumentasi program 	
16	UJIAN AKHIR SEMESTER					

Catatan :

[1]. TM : tatap Muka


[2]. [TM:2x50'] : Kuliah tatap muka 1 kali (minggu) x 2 sks x 50 menit=100 menit

[3]. [BT+BM:(1+1)x(2x50')] : Belajar terstruktur 1 kali (minggu) dan belajar mandiri 1 kali (minggu) x 2 sks x 50 menit = 200 menit (3 jam 20 menit)

[4]. Penulisan daftar pustaka disarankan menggunakan salah satu standar/style penulisan pustaka internasional, dalam contoh ini menggunakan style APA

[5]. RPS : Rencana Pembelajaran Semester, RMK : Rumpun Mata Kuliah, Prodi : Program Studi

RENCANA TUGAS MAHASISWA 1

	PERGURUAN TINGGI FAKULTAS PROGRAM STUDI	: UNIVERSITAS DIAN NUSWANTORO : ILMU KOMPUTER : TEKNIK INFORMATIKA – S1
RENCANA TUGAS MAHASISWA		
MATA KULIAH	: Pemrograman Berorientasi Obyek	
KODE	: A11.54402	
SKS	: 4	
SEMESTER	: 4	
DOSEN	: Tim Pengampu Mata Kuliah Pemrograman Berorientasi Obyek	
BENTUK TUGAS	: Tugas Mandiri	
JUDUL TUGAS	: Tugas-1: Instalasi	
SUB CAPAIAN PEMBELAJARAN MATA KULIAH		
Mahasiswa mengetahui tools atau aplikasi yang digunakan untuk membuat program		
DESKRIPSI TUGAS		
Mencari tools yang digunakan untuk menulis code Java seperti compiler dan editor, lalu mahasiswa wajib menginstallasi tools tersebut pada computer pribadi dan membuat kode sederhana untuk menampilkan data diri		
METODE Pengerjaan Tugas		
<ol style="list-style-type: none"> 1. Mencari Tools yang digunakan untuk menulis code Java lalu mendownload dan menginstall di komputer pribadi 2. Mahasiswa membuat kode sederhana untuk menampilkan data pribadi seperti nama, alamat, ttl, dll. 3. Menyusun laporan hasil percobaan, berupa step by step instalasi, menulis program, compile dan running program 		
BENTUK DAN FORMAT LUARAN		
a. Objek garapan : instalasi tools untuk menulis Bahasa Java dan menulis kode sederhana untuk menampilkan data pribadi dan program untuk menghitung rumus bidang b. Bentuk Luaran Laporan makalah ditulis dengan MS Word dengan sistematika dan format sesuai dengan standar panduan penulisan laporan di Udinus, dikumpulkan dengan format ekstensi (*.pdf), dengan sistematika nama file: (Kelompok kuliah-Tugas 1- Nim Nama)		
INDIKATOR, KRITERIA DAN BOBOT PENILAIAN		
a. Format Penulisan [bobot 20%] <ol style="list-style-type: none"> 1. Penulisan Makalah dengan sistematika dan format yang telah ditetapkan, kemutakhiran <i>literature</i> (5 tahun terakhir), kejelasan dan ketajaman meringkas, konsistensi dan kerapian dalam sajian tulisan. 2. Kesesuaian sistematika nama file yang dikirim melalui email 		
b. Isi Makalah [50%] <ol style="list-style-type: none"> 1. Ketepatan sistematika pelaporan makalah sesuai dengan standar panduan penulisan laporan; 2. Ketepatan tata tulis makalah sesuai dengan ejaan bahasa Indonesia yang benar dan baik dalam penyajian tabel, gambar; 3. Kerapian sajian makalah yang dikumpulkan; 4. Kelengkapan penggunaan fitur-fitur yang ada dalam MS Word dalam penulisan dan sajian makalah 5. Penulisan rujukan dan penulisan sitasi yang baik 		
c. Ketepatan Waktu Pengumpulan [30%] Kesesuaian waktu pengiriman tugas dengan jadwal yang sudah disepakati		
JADWAL PELAKSANAAN		
Pengumpulan tugas : Minggu ke 2		
LAIN-LAIN		

Kirimkan ke dosen pengampu masing-masing

DAFTAR PUSTAKA

1. David J. Barnes & Michael Kölling, Object First With Java, Fifth edition, Prentice Hall / Pearson Education, 2012.
2. Luliana Cosmina, Java for Absolute Beginners- Learn to Program the Fundamentals the Java 9+ Way, APress, 2018.
3. Roger Codenhead, Java in 21 Days, Sams Teach Yourself (Covering Java 8), 7th Edition, Pearson Education Inc, 2016.
4. Oracle, The Java™ Tutorial, <http://docs.oracle.com/javase/tutorial/java/nutsandbolts/>, 1995-2019.
5. Oracle, The Java Tutorial Bundle, <http://www.oracle.com/technetwork/java/javase/downloads/java-se-7-tutorial-2012-02-28-1536013.html>, 2017.