
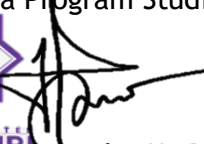





UNIVERSITAS NAHDHATUL ULAMA SUNAN GIRI
FAKULTAS SAINS DAN TEKNOLOGI
PROGRAM STUDI S1 SISTEM KOMPUTER

RENCANA PEMBELAJARAN SEMESTER

MATA KULIAH	KODE MATA KULIAH	BOBOT (SKS)	SEMESTER	TANGGAL PENGESAHAN
Pemograman Berbasis GUI	MKB 306	3	Ganjil 2021/2022	21 September 2021
Otoritas / Pengesahan	Dosen Pengembang RPS  Ucta Paradema, M.Kom.	Ketua Program Studi  Rahmat Irsyada, M. Pd.	Unit Kendali Mutu Prodi  Roihatur Rohmah, M. Si.	

1. Program Studi	S1 Sistem Komputer
2. Matakuliah	Pemograman Berbasis GUI
3. Kode Matakuliah	MKB 306
4. Bobot SKS	3
5. Semester	Ganjil 2021.2022
6. Matakuliah Prasyarat	-
7. Dosen Pengampu	Ucta Paradema, M.Kom
8. Capaian Pembelajaran Matakuliah (CPMK)	Setelah mengikuti mata kuliah pemograman berbasis GUI mahasiswa akan: konsep-konsep dasar padaobject oriented programming (OOP), seperti class – object, encapsulation, inheritance, polymorphism, interface, abstract class.

9. Bahan Kajian	<ol style="list-style-type: none"> 1. OOP history 2. Encapsulation 3. Class relationship (class diagram) 4. Inheritance & polymorphism 5. Interface & abstract classes 6. Static & final method & variables 7. GUI & SWING
------------------------	---

ACARA PEMBELAJARAN

Minggu Ke	Kemampuan Akhir Yang Diharapkan	Bahan Kajian (Materi Ajar)	Metode Pembelajaran	Alokasi Waktu	Kriteria (Indikator) Capaian	Penilaian		Referensi
						Instrumen	Bobot	
<i>10</i>	<i>11</i>	<i>12</i>	<i>13</i>	<i>14</i>	<i>15</i>	<i>16</i>	<i>17</i>	<i>18</i>
1	Mahasiswa menguasai pemahaman tentang CP mata kuliah dan cara hpencapaiannya melalui proses pembelajaran dengan bahan kajiannya selama satu semester	RPS, SAP, kontrak perkuliahan dan Instrumen asesment	<ul style="list-style-type: none"> • Pemaparan di kelas dan diskusi kelompok. • Belajar mandiri untuk konstruksi pengetahuan tentang CP mata kuliah dan cara pencapaiannya • Penugasan terstruktur: Membuat narasi esay tentang CP mata kuliah dan peran mahasiswa dalam pembelajaran 	2 x 50 menit 2 x 60 menit 2 x 60 menit	Mahasiswa menjelaskan dengan baik tentang CP, proses pembelajaran dan perannya dalam pembelajaran untuk mencapai CP.	Rubrik Holistk	5%	

2	<p>Mahasiswa mengetahui sejarah OOP.</p> <p>Mahasiswa memahami perlunya OOP untuk menyelesaikan permasalahan, yang kurang tepat jika procedural oriented diterapkan.</p> <p>Mahasiswa mengetahui bahasa pemrograman Java yang akan digunakan untuk mempelajari OOP</p>	<ul style="list-style-type: none"> • OOP history • compare procedural and oop by giving a simple case study • Introduction to java 	<ul style="list-style-type: none"> • presentasi 	<p>2 x 50 menit</p> <p>2 x 60 menit</p> <p>2 x 60 menit</p>	<p>Mahasiswa menjelaskan dengan baik tentang</p> <ul style="list-style-type: none"> • OOP history • compare procedural and oop by giving a simple case study • Introduction to java 	Rubrik holistik	10%	
3	Mahasiswa memahami	<ul style="list-style-type: none"> • The difference between class 	<ul style="list-style-type: none"> • presentasi dan diskusi • praktik 	2 x 50	Mahasiswa	Rubrik		
Minggu Ke	Kemampuan Akhir Yang Diharapkan	Bahan Kajian (Materi Ajar)	Metode Pembelajaran	Alokasi Waktu	Kriteria (Indikator) Capaian	Penilaian		Referensi
						Instrumen	Bobot	
10	11	12	13	14	15	16	17	18
	<p>perbedaan antara kelas dan object</p> <p>Mahasiswa mampu membuat kelas (sederhana) dan object dalam bahasa Java.</p>	<p>and object, by giving some examples</p> <ul style="list-style-type: none"> • making a simple first object (Fokus: first programming with java) 		<p>menit</p> <p>2 x 60 menit</p> <p>2 x 60 menit</p>	<p>mempresentasikan:</p> <ul style="list-style-type: none"> • The difference between class and object, by giving some examples • making a simple first object (Fokus: first programming with java) 	presentasi		
						Rubrik deskriptif		

4	<p>Mahasiswa memahami penggunaan serta dapat membuat program sederhana menggunakan tipe primitif, array, keyword, reference variable, deklarasi dan assignment objek</p> <p>Mahasiswa memahami dan mengimplementasikan dalam program tentang siklus objek (penggunaan constructor dan garbage collector)</p>	<ul style="list-style-type: none"> Primitive and reference: Type, keywords, reference variable, object declaration & assignment, objects in garbage collector, arrays Object cycle: life and death Constructor & Garbage 	<ul style="list-style-type: none"> presentasi dan diskusi praktik 	<p>2 x 50 menit</p> <p>2 x 60 menit</p> <p>2 x 60 menit</p>	<p>Mahasiswa menjelaskan:</p> <ul style="list-style-type: none"> Primitive and reference: Type, keywords, reference variable, object declaration & assignment, objects in garbage collector, arrays Object cycle: life and death Constructor & Garbage 	<p>Rubrik presentasi</p> <p>Rubrik deskriptif</p>	15%	
5	<p>Mahasiswa memahami makna dan perlunya</p>	<p>Encapsulation • Why do we need</p>	<ul style="list-style-type: none"> Pemaparan dan diskusi 	<p>2 x 50 menit</p> <p>2 x 60</p>	<p>Mahasiswa mengkonstruksikan pengetahuannya</p>			
Minggu Ke	Kemampuan Akhir Yang Diharapkan	Bahan Kajian (Materi Ajar)	Metode Pembelajaran	Alokasi Waktu	Kriteria (Indikator) Capaian	Penilaian		Referensi
						Instrumen	Bobot	
<i>10</i>	<i>11</i>	<i>12</i>	<i>13</i>	<i>14</i>	<i>15</i>	<i>16</i>	<i>17</i>	<i>18</i>

	enkapsulasi. Mahasiswa memahami dan dapat mengimplementasikan enkapsulasi dalam suatu program sederhana.	<ul style="list-style-type: none"> • encapsulation? • Object state • Instance variable vs local variables • Methods argument & return type • Pass by value • Getters and setters • Sample class diagram representing encapsulation 		menit 2 x 60 menit	dengan baik	Rubrik holistik		
6	Mahasiswa memahami diagram kelas beserta makna simbol relasi yang terdapat pada diagram kelas.	<p>Class relationship (class diagram)</p> <ul style="list-style-type: none"> • association, aggregation, composition • Example: simple class diagram & code in java related 	<ul style="list-style-type: none"> • Presentasi dan diskusi • praktikum 	2 x 50 menit 2 x 60 menit 2 x 60 menit	Mahasiswa menjelaskan : <ul style="list-style-type: none"> • association, aggregation, composition • Example: simple class diagram & code in java related 	Rubrik holistik	5%	
7	Mahasiswa memahami diagram kelas beserta makna simbol relasi yang terdapat pada diagram kelas.	<p>Class relationship (class diagram)</p> <ul style="list-style-type: none"> • association, aggregation, composition <p>Example: simple class diagram &</p>	<ul style="list-style-type: none"> • Presentasi dan diskusi • praktikum 	2 x 50 menit 2 x 60 menit 2 x 60	Mahasiswa menjelaskan : <ul style="list-style-type: none"> • association, aggregation, composition • Example: simple class diagram & code in java related 	Rubrik holistik	20%	
Minggu Ke	Kemampuan Akhir Yang Diharapkan	Bahan Kajian (Materi Ajar)	Metode Pembelajaran	Alokasi Waktu	Kriteria (Indikator) Capaian	Penilaian		Referensi
						Instrumen	Bobot	

10	11	12	13	14	15	16	17	18
		code in java related		menit				
UJIAN TENGAH SEMESTER (UTS)								
8	Mahasiswa memahami makna dan perlunya inheritance dan polymorphism.	Inheritance & polymorphism <ul style="list-style-type: none"> Why do we need inheritance & polymorphism? How inheritance promotes software reusability Designing for inheritance Superclass and subclass relationship & access modifier Method Overriding 	<ul style="list-style-type: none"> Presentasi dan diskusi praktikum 	2 x 50 menit 2 x 60 menit 2 x 60 menit	Mahasiswa mampu menjelaskan: <ul style="list-style-type: none"> Why do we need inheritance & polymorphism? How inheritance promotes software reusability Designing for inheritance Superclass and subclass relationship & access modifier Method Overriding 	Rubrik Holistik	5%	
9	Mahasiswa memahami dan dapat mengimplementasikan inheritance dan polymorphism dalam suatu program sederhana.	<ul style="list-style-type: none"> Polymorphism: in object declaration & assignment, in method argument & return type Method overloading Constructor overloading 	<ul style="list-style-type: none"> Presentasi dan diskusi praktikum 	2 x 50 menit 2 x 60 menit 2 x 60 menit	Mahasiswa mampu menjelaskan: <ul style="list-style-type: none"> Polymorphism: in object declaration & assignment, in method argument & return type Method overloading Constructor overloading 	Rubrik Holistik	5%	
Minggu Ke	Kemampuan Akhir Yang Diharapkan	Bahan Kajian (Materi Ajar)	Metode Pembelajaran	Alokasi Waktu	Kriteria (Indikator) Capaian	Penilaian	Referensi	

						Instrumen	Bobot	
<i>10</i>	<i>11</i>	<i>12</i>	<i>13</i>	<i>14</i>	<i>15</i>	<i>16</i>	<i>17</i>	<i>18</i>
		(revisited) & superclass constructor <ul style="list-style-type: none"> • Calls to super(), this () • Sample class diagram 			(revisited) & superclass constructor <ul style="list-style-type: none"> • Calls to super(), this () • Sample class diagram 			
10	Mahasiswa memahami perlunya dan mengimplementasikan interface dan abstract classes dalam suatu program sederhana.	Interface & abstract classes <ul style="list-style-type: none"> • Why do we need abstract classes? Abstract vs concrete class • Abstract classes and methods • Overloading and overriding (revisited) 	<ul style="list-style-type: none"> • Presentasi dan diskusi 	2 x 50 menit 2 x 60 menit 2 x 60 menit	Mahasiswa mampu memahami Interface & abstract classes <ul style="list-style-type: none"> • Why do we need abstract classes? Abstract vs concrete class • Abstract classes and methods • Overloading and overriding (revisited) 	Rubrik holistik Rubric deskriptif	5%	
11	Mahasiswa memahami perlunya dan mengimplementasikan interface dan abstract classes dalam suatu program sederhana.	<ul style="list-style-type: none"> • Casting • Why do we need interface? • Interface: making & using • Sample class diagram representing interface & abstract class 	<ul style="list-style-type: none"> • Presentasi dan diskusi 	2 x 50 menit 2 x 60 menit 2 x 60 menit	Mahasiswa mampu menjelaskan <ul style="list-style-type: none"> • Casting • Why do we need interface? • Interface: making & using • Sample class diagram representing interface & abstract 	Rubrik holistik Rubric deskriptif	5%	

Minggu Ke	Kemampuan Akhir Yang Diharapkan	Bahan Kajian (Materi Ajar)	Metode Pembelajaran	Alokasi Waktu	Kriteria (Indikator) Capaian	Penilaian		Referensi
						Instrumen	Bobot	
10	11	12	13	14	15	16	17	18
					class			
12	Mahasiswa memahami perlunya dan mengimplementasikan static & final variabel dan method dalam suatu program sederhana.	Static & final method & variables <ul style="list-style-type: none"> • Static method • Static variable • Constants (static final variable) • Final method & class • Wrapper classes & autoboxing • Static imports • Example: Math class • Instance variable vs static variable 	<ul style="list-style-type: none"> • Presentasi • Diskusi • praktikum 	2 x 50 menit 2 x 60 menit 2 x 60 menit	Mahasiswa mampu menjelaskan Static & final method & variables <ul style="list-style-type: none"> • Static method • Static variable • Constants (static final variable) • Final method & class • Wrapper classes & autoboxing • Static imports • Example: Math class Instance variable vs static variable	Rubrik holistik Rubric deskriptif	5%	
13	Mahasiswa mampu membuat aplikasi berbasis GUI/Swing	GUI & SWING <ul style="list-style-type: none"> • User event • Listener interface • Action event • Inner class • Swing component 	<ul style="list-style-type: none"> • Presentasi • Diskusi praktikum 	2 x 50 menit 2 x 60 menit 2 x 60 menit	Mahasiswa mampu mengimplementasikan GUI & SWING <ul style="list-style-type: none"> • User event • Listener interface • Action event • Inner class Swing component	Rubrik holistik Rubric deskriptif	5%	

14	Mahasiswa mampu membuat aplikasi berbasis GUI/Swing	GUI & SWING • User event • Listener	• Presentasi • Diskusi • praktikum	2 x 50 menit	Mahasiswa mampu mengimplementasikan GUI & SWING	Rubrik holistik	5%	
Minggu Ke	Kemampuan Akhir Yang Diharapkan	Bahan Kajian (Materi Ajar)	Metode Pembelajaran	Alokasi Waktu	Kriteria (Indikator) Capaian	Penilaian		Referensi
						Instrumen	Bobot	
<i>10</i>	<i>11</i>	<i>12</i>	<i>13</i>	<i>14</i>	<i>15</i>	<i>16</i>	<i>17</i>	<i>18</i>
		• interface • Action event • Inner class • Swing component		2 x 60 menit 2 x 60 menit	• User event • Listener • interface • Action event • • Inner class Swing component	Rubric deskriptif		
UJIAN AKHIR SEMESTER (UAS)								

REFERENSI

1. Head First Java, 2nd edition, 2008, Bert Bates and Kathy Sierra, O'Reilly
2. Java™ How to Program, 9th, 2012, Prentice Hall
3. Head First Object Oriented Design and Analysis, 1st edition, 2006, Brett D. McLaughlin, Gary Pollice, David West, O'Reilly Media

