

RENCANA PEMBELAJARAN SEMESTER (RPS)

IKG214

SOFTWARE PROJECT 1



Disusun oleh:

<Nama Dosen/ Tim Pengajar MK Software Project 1>

**PROGRAM STUDI S1 ILMU KOMPUTASI
FAKULTAS INFORMATIKA**

TELKOM UNIVERSITY

LEMBAR PENGESAHAN

Rencana Pembelajaran Semester (RPS) ini telah disahkan untuk mata kuliah berikut:

Kode Mata Kuliah : **IKG214**

Nama Mata Kuliah : ***SOFTWARE PROJECT 1***

Mengetahui

Ketua Program Studi S1 Ilmu Komputasi

Dr. Deni Saepudin

Bandung, 2015

Menyetujui

Ketua KK <nama KK yang menjadi pengampu
MK *Software Project 1* di ProDi S1 Ilmu
Komputasi>

<mohon diisi dengan nama ketua KK dilengkapi
dengan gelar akademik terakhir>

DAFTAR ISI

LEMBAR PENGESAHAN.....	ii
DAFTAR ISI.....	iii
A. PROFIL MATA KULIAH.....	1
IDENTITAS MATA KULIAH	1
DESKRIPSI SINGKAT MATA KULIAH	1
DAFTAR PUSTAKA	1
B. RENCANA PEMBELAJARAN SEMESTER (RPS)	2
C. RANCANGAN INTERAKSI DOSEN–MAHASISWA.....	10
D. RANCANGAN TUGAS	24
E. PENILAIAN DENGAN RUBRIK	32
F. PENENTUAN NILAI AKHIR MATA KULIAH.....	33

A. PROFIL MATA KULIAH

IDENTITAS MATA KULIAH

Nama Mata Kuliah	:	<i>Software Project 1</i>	
Kode Mata Kuliah	:	IKG2I4	
SKS	:	4 (empat)	
Jenis	:	Mata kuliah wajib	
Jam pelaksanaan	:	Tatap muka di kelas	= 4 jam per pekan
		Tutorial/ responsi	= 2 jam per pekan
Semester / Tingkat	:	4 / 2	
<i>Pre-requisite</i>	:	Pemrograman Terstruktur 1 (KUG1E3) dan Pemrograman Terstruktur 2 (IKG2A3).	
<i>Co-requisite</i>	:	-	
Bidang Kajian	:	Pemrograman berorientasi objek (<i>object oriented programming</i>), konsep sistem basis data (<i>database system concepts</i>).	

DESKRIPSI SINGKAT MATA KULIAH

Mata kuliah ini merupakan mata kuliah yang bertujuan untuk membekali mahasiswa program studi sarjana Ilmu Komputasi dengan kemampuan dan *skill* praktis untuk mengembangkan perangkat lunak. Bidang kajiannya meliputi pemrograman berorientasi objek (*object-oriented programming*) dan konsep sistem basis data (*database system concepts*).

DAFTAR PUSTAKA

1. J. Barker, *Beginning Java Object: From Concept to Code, 2nd Edition*, Apress: 2005.
2. B. Bates, K. Sierra, *Head First Java, 2nd Edition*, O'Reilly Media: 2005.
3. H. Deitel, P. Deitel, *How to Program, 9th Edition*, Prentice Hall: 2010.
4. http://www.tutorialspoint.com/computer_programming_tutorials.htm
5. Manual dan tutorial JDK 7 API DOCS
6. MySQL documentations
7. Oracle documentations

B. RENCANA PEMBELAJARAN SEMESTER (RPS)

Pekan ke-	Kemampuan Akhir yang Diharapkan	Bahan Kajian (Materi Ajar)	Bentuk/ Metode/ Strategi Pembelajaran	Kriteria Penilaian (Indikator)	Bobot Nilai
1	<ul style="list-style-type: none"> Memahami konsep pemrograman berorientasi objek serta terminologi-terminologi yang terdapat di dalamnya. Mampu menggunakan <i>class diagram</i> sebagai salah satu aspek dalam perancangan perangkat lunak. Mampu menjalankan program dalam bahasa Java dengan <i>compiler</i> yang berbasis <i>command prompt</i>/terminal. Memahami definisi <i>variable</i>, tipe data, percabangan, dan perulangan pada Java. 	<ol style="list-style-type: none"> Konsep dasar pemrograman berorientasi objek. <i>Class diagram</i>. Java. 	<ul style="list-style-type: none"> Kuliah yang memuat ceramah, diskusi, dan latihan. Tugas mingguan. <p><mohon diperiksa kembali></p>	<p>Mahasiswa memahami:</p> <ol style="list-style-type: none"> konsep pemrograman berorientasi objek, serta perbedaan antara <i>object</i> dan <i>class</i> perancangan perangkat lunak dengan <i>class diagram</i>, yang meliputi: <i>class</i>, <i>attribute</i>, <i>method</i>, dan <i>constructor</i>. penggunaan <i>tool</i> star UML untuk membuat <i>class diagram</i> cara instalasi Java, cara <i>compile</i>, dan cara <i>run</i> program yang ditulis dalam Java; <i>compiler</i> diharuskan berbasis <i>command prompt</i> atau terminal; penggunaan IDE hanya untuk tugas besar definisi <i>variable</i>, tipe data (<i>data type</i>), percabangan (<i>conditionals</i>), loop (<i>perulangan</i>) pada Java. 	<harap diisi dalam %>
2	<ul style="list-style-type: none"> Memahami konsep <i>constructor</i> dan cara 	<ol style="list-style-type: none"> <i>Class driver</i>. <i>Constructor</i>. 	<ul style="list-style-type: none"> Kuliah yang memuat 	<p>Mahasiswa memahami:</p> <ol style="list-style-type: none"> konsep <i>constructor</i> sebagai <i>method</i> 	<harap diisi dalam %>

Pekan ke-	Kemampuan Akhir yang Diharapkan	Bahan Kajian (Materi Ajar)	Bentuk/ Metode/ Strategi Pembelajaran	Kriteria Penilaian (Indikator)	Bobot Nilai
	<p>menginstansiasi <i>object</i> dari <i>class</i></p> <ul style="list-style-type: none"> Memahami <i>class driver</i> Memahami perbedaan akses <i>public</i> dan <i>private</i> Memahami <i>method get</i> dan <i>set</i> dan dapat mengimplementasikannya Mampu mengimplementasikan <i>class diagram</i> dalam bahasa Java Memahami standar disiplin pemrograman secara umum maupun yang dipakai di OOP. 	<p>3. Akses <i>Public</i> dan <i>private</i>.</p> <p>4. <i>Method Get</i> dan <i>method set</i>.</p> <p>5. Implementasi <i>class diagram</i> dalam Java.</p>	<p>ceramah, diskusi, dan latihan.</p> <ul style="list-style-type: none"> Tugas mingguan. <p><mohon diperiksa kembali></p>	<p>khusus untuk menginstansiasi <i>object</i></p> <p>2. cara menginstansiasi <i>object</i> dari <i>class</i> tertentu</p> <p>3. cara memanipulasi data atribut pada <i>object</i> yang diinstansiasi dari suatu <i>class</i></p> <p>4. cara memanggil <i>method</i> dari sebuah <i>class</i> untuk <i>object</i> yang dinstansiasi dari suatu <i>class</i></p> <p>5. perbedaan antara akses <i>public</i> dan <i>private</i>; <i>attribute</i> umumnya <i>private</i>, sedangkan <i>method</i> umumnya <i>public</i></p> <p>6. pengertian <i>method get</i> dan <i>set</i> serta cara pemakaiannya</p> <p>7. konstanta (<i>final attribute</i>)</p> <p>8. cara implementasi <i>class diagram</i> ke dalam bahasa Java</p> <p>9. standar disiplin pemrograman secara umum maupun yang dipakai di OOP.</p>	%>
3	<ul style="list-style-type: none"> Memahami konsep <i>inheritance</i>, pewarisan <i>attribute</i> maupun <i>method</i>, dan cara pemanggilan <i>method</i> 	<p>1. <i>Inheritance</i></p> <p>2. <i>Overriding</i></p> <p>3. <i>Overloading</i></p>	<ul style="list-style-type: none"> Kuliah yang memuat ceramah, diskusi, dan latihan. Tugas mingguan. 	<p>Mahasiswa memahami:</p> <p>1. konsep <i>inheritance</i> (<i>class parent</i> dan <i>class child</i>)</p> <p>2. pewarisan <i>attribute/ method</i></p> <p>3. cara pemanggilan <i>method</i> dari <i>class</i></p>	<harap diisi dalam %>

Pekan ke-	Kemampuan Akhir yang Diharapkan	Bahan Kajian (Materi Ajar)	Bentuk/ Metode/ Strategi Pembelajaran	Kriteria Penilaian (Indikator)	Bobot Nilai
	<p>dari <i>class</i>.</p> <ul style="list-style-type: none"> Memahami konsep <i>overriding</i> dan mampu menerapkannya. Memahami konsep <i>overloading</i> dan mampu menerapkannya. 		<mohon diperiksa kembali>	<p><i>child</i></p> <ol style="list-style-type: none"> memahami terminologi <i>access protected</i> memahami konsep <i>overriding</i> dan mampu menerapkannya memahami konsep final pada <i>attribute, method, dan class</i> memahami konsep <i>this</i> dan <i>super</i> memahami konsep <i>overloading</i> dan mampu menerapkannya. 	
4	<ul style="list-style-type: none"> Memahami pengertian hubungan antar kelas (agregasi) pada pemrograman berorientasi objek. Mampu menerapkan hubungan antar kelas pada pemrograman berorientasi objek dalam bahasa Java. 	Relasi antar kelas pada pemrograman berorientasi objek.	<ul style="list-style-type: none"> Kuliah yang memuat ceramah, diskusi, dan latihan. Tugas mingguan. <p><mohon diperiksa kembali></p>	<p>Mahasiswa memahami:</p> <ol style="list-style-type: none"> hubungan antar kelas (agregasi) pada pemrograman berorientasi objek cara penerapan hubungan antar kelas pada pemrograman berorientasi objek dalam bahasa Java. 	<harap diisi dalam %>
5	<ul style="list-style-type: none"> Memahami konsep hubungan antar kelas (agregasi) yang melibatkan <i>array</i> pada pemrograman 	Relasi antar kelas pada pemrograman berorientasi objek yang melibatkan <i>array</i> .	<ul style="list-style-type: none"> Kuliah yang memuat ceramah, diskusi, dan latihan. Tugas mingguan. 	<p>Mahasiswa memahami:</p> <ol style="list-style-type: none"> hubungan antar kelas (agregasi) yang melibatkan <i>array</i> pada pemrograman berorientasi objek cara penerapan hubungan antar kelas 	<harap diisi dalam %>

Pekan ke-	Kemampuan Akhir yang Diharapkan	Bahan Kajian (Materi Ajar)	Bentuk/ Metode/ Strategi Pembelajaran	Kriteria Penilaian (Indikator)	Bobot Nilai
	<p>berorientasi objek.</p> <ul style="list-style-type: none"> Mampu menerapkan hubungan antar kelas yang melibatkan <i>array</i> pada pemrograman berorientasi objek dalam bahasa Java. 		<mohon diperiksa kembali>	yang melibatkan <i>array</i> pada pemrograman berorientasi objek dalam bahasa Java.	
6	<ul style="list-style-type: none"> Memahami konsep <i>abstract class</i> dan <i>polymorphism</i> dan penerapannya dalam bahasa Java. Memahami konsep <i>interface</i> dan penerapannya dalam bahasa Java. Memahami konsep <i>exception try, catch, dan throw</i> pada bahasa Java dan mampu membuat <i>exception</i> pada Java. 	<ol style="list-style-type: none"> <i>Abstract class</i>. <i>Interface</i>. <i>Exception</i>. 	<ul style="list-style-type: none"> Kuliah yang memuat ceramah, diskusi, dan latihan. Tugas mingguan. <mohon diperiksa kembali>	Mahasiswa memahami: <ol style="list-style-type: none"> konsep <i>abstract class</i> konsep <i>interface</i> konsep <i>polymorphism</i> pada <i>source code</i> Java konsep <i>exception try, catch, dan throw</i> cara membuat <i>exception</i> pada bahasa Java. 	<harap diisi dalam %>
7	<ul style="list-style-type: none"> Memahami penggunaan Java API DOCS dan dapat 	<ol style="list-style-type: none"> Java API DOCS. Java <i>library</i>. IO <i>stream</i> pada Java. 	<ul style="list-style-type: none"> Kuliah yang memuat ceramah, diskusi, 	Mahasiswa memahami: <ol style="list-style-type: none"> penggunaan Java API DOCS dan dapat mengeksplorasi Java API DOCS secara 	<harap diisi dalam %>

Pekan ke-	Kemampuan Akhir yang Diharapkan	Bahan Kajian (Materi Ajar)	Bentuk/ Metode/ Strategi Pembelajaran	Kriteria Penilaian (Indikator)	Bobot Nilai
	<p>melakukan eksplorasi secara mandiri.</p> <ul style="list-style-type: none"> Mengenal Java <i>library</i> dan dapat memakainya sesuai dengan kebutuhan pemrograman yang sesuai. Memahami IO <i>stream</i> pada Java dan dapat menggunakannya. 		<p>dan latihan.</p> <ul style="list-style-type: none"> Tugas mingguan. <p><mohon diperiksa kembali></p>	<p>mandiri</p> <ol style="list-style-type: none"> penggunaan Java <i>library</i> dan dapat menentukan <i>library</i> yang sesuai dengan kebutuhan pemrograman yang dilakukan (contohnya Java lang, Java util, Java collection, dan lain-lain) IO <i>stream</i> pada Java dan dapat menggunakannya fungsi-fungsi dari <i>wrapper</i>, <i>type casting</i>, <i>package</i>, JAR, dan JAVADOC. 	
8	<ul style="list-style-type: none"> Mampu menggunakan GUI yang meliputi AWT, <i>swing</i>, dan <i>action listener</i>. Mampu membuat GUI tanpa memakai IDE. Memahami konsep-konsep dasar <i>JFrame</i>, <i>JButton</i>, <i>JLabel</i>, <i>JTextField</i>, dan lain-lain serta dapat menerapkannya. Mampu menerapkan paradigma orientasi 	<ol style="list-style-type: none"> <i>Graphical user interface</i> (GUI). Pemrograman berorientasi objek dengan C++. 	<ul style="list-style-type: none"> Kuliah yang memuat ceramah, diskusi, dan latihan. Tugas mingguan. <p><mohon diperiksa kembali></p>	<p>Mahasiswa memahami:</p> <ol style="list-style-type: none"> cara penggunaan GUI yang meliputi AWT, <i>swing</i>, dan <i>action listener</i> cara membuat GUI tanpa memakai IDE konsep-konsep dasar <i>JFrame</i>, <i>JButton</i>, <i>JLabel</i>, <i>JTextFied</i>, dan lain-lain serta dapat menerapkannya cara menerapkan OOP dalam bahasa C++ <i>standard library</i> pada C++ dan cara penggunaannya. 	<harap diisi dalam %>

Pekan ke-	Kemampuan Akhir yang Diharapkan	Bahan Kajian (Materi Ajar)	Bentuk/ Metode/ Strategi Pembelajaran	Kriteria Penilaian (Indikator)	Bobot Nilai
	<p>objek dalam bahasa C++.</p> <ul style="list-style-type: none"> Mengenal <i>standard library</i> pada C++ dan mampu memakainya. 				
9	<ul style="list-style-type: none"> Memahami ER <i>model</i> dari suatu perangkat lunak. Mampu membuat ER <i>diagram</i> dalam proses perancangan perangkat lunak. Mengenal ORACLE dan dapat memakainya. Mengenal SQL dan dapat memakai <i>query-query</i> dasar pada SQL. 	<ol style="list-style-type: none"> ER <i>model</i> dari suatu perangkat lunak. Pengenalan ORACLE. Dasar-dasar <i>query</i> pada SQL. 	<ul style="list-style-type: none"> Kuliah yang memuat ceramah, diskusi, dan latihan. Tugas mingguan. <p><mohon diperiksa kembali></p>	<p>Mahasiswa memahami:</p> <ol style="list-style-type: none"> <i>entity relationship diagram</i> (ER <i>diagram</i>), salah satunya dalam bentuk tabel relasi <i>primary key</i> dan <i>foreign key</i> pada ER <i>diagram</i> dan dapat memberikan contohnya cara instalasi ORACLE pada komputer struktur tabel ORACLE HRM (<i>human resource model</i>) cara merancang <i>query</i> SQL sederhana pada sebuah tabel, contohnya SELECT-WHERE-FROM, menggunakan skema ORACLE HRM <i>query</i> SQL yang melibatkan dua tabel atau lebih menggunakan klausa JOIN penggunaan <i>tool Power Designer</i> untuk merancang tabel relasi (generated DLL) cara pengurutan data pada SQL menggunakan ORDER BY 	<harap diisi dalam %>

Pekan ke-	Kemampuan Akhir yang Diharapkan	Bahan Kajian (Materi Ajar)	Bentuk/ Metode/ Strategi Pembelajaran	Kriteria Penilaian (Indikator)	Bobot Nilai
				9. cara melakukan kalkulasi aritmetika di dalam <i>query</i> SQL.	
10	<ul style="list-style-type: none"> Memahami dan mengenal contoh dari <i>aggregate function</i> sebagai <i>query</i> SQL yang lebih kompleks. Memahami klausa GROUP BY dan dapat menerapkannya. Memahami klausa JOIN dan dapat menerapkannya. 	<ol style="list-style-type: none"> <i>Aggregate function</i> sebagai <i>query</i> SQL yang lebih kompleks (COUNT, MAX, MIN, SUM, AVG). Klausa GROUP BY dan implementasinya. Klausa JOIN dan implementasinya. 	<ul style="list-style-type: none"> Kuliah yang memuat ceramah, diskusi, dan latihan. Tugas mingguan. <p><mohon diperiksa kembali></p>	<p>Mahasiswa memahami:</p> <ol style="list-style-type: none"> <i>aggregate function</i> sebagai <i>query</i> SQL yang lebih kompleks, contohnya COUNT, MAX, MIN, SUM, AVG klausa GROUP BY dan cara memakainya klausa HAVING dan cara memakainya klausa JOIN, jenis, dan contohnya, yaitu: <i>natural join</i> (USING, ON), <i>self-join</i>, <i>outer join</i> (left, right, full), dan <i>cross join</i>(Cartesian product). 	<harap diisi dalam %>
11	<ul style="list-style-type: none"> Memahami perintah-perintah pada <i>data manipulation language</i> (DML) dan dapat memakainya. Memahami perintah-perintah pada <i>data definition language</i> (DDL) dan dapat memakainya. Memahami <i>statement</i> DROP TABLE. 	<ol style="list-style-type: none"> <i>Data manipulation language</i> (DML). <i>Data definition language</i> (DDL). <i>Statement</i> DROP TABLE. 	<ul style="list-style-type: none"> Kuliah yang memuat ceramah, diskusi, dan latihan. Tugas mingguan. <p><mohon diperiksa kembali></p>	<p>Mahasiswa memahami:</p> <ol style="list-style-type: none"> <i>data manipulation language</i> (DML) yang meliputi perintah INSERT, UPDATE, dan DELETE <i>data definition language</i> (DDL) yang meliputi perintah CREATE, tipe data yang digunakan untuk <i>column table</i>, dan jenis-jenis <i>constraints</i> seperti NOT NULL, PRIMARY KEY, dan FOREIGN KEY <i>statement</i> DROP TABLE. 	<harap diisi dalam %>

Pekan ke-	Kemampuan Akhir yang Diharapkan	Bahan Kajian (Materi Ajar)	Bentuk/ Metode/ Strategi Pembelajaran	Kriteria Penilaian (Indikator)	Bobot Nilai
12	<ul style="list-style-type: none"> Memahami aspek-aspek pada <i>database administration</i>. Mampu menggunakan <i>tools IDE</i> untuk membuat program berorientasi objek. Mampu merancang GUI menggunakan Java Swing. 	<ol style="list-style-type: none"> <i>Database administration</i>. <i>Java database connectivity</i> (JDBC). <i>Tools IDE</i>: Eclipse dan NetBeans. GUI (Java Swing). 	<ul style="list-style-type: none"> Kuliah yang memuat ceramah, diskusi, dan latihan. Tugas mingguan. <mohon diperiksa kembali> 	<p>Mahasiswa memahami:</p> <ol style="list-style-type: none"> <i>database administration</i> yang meliputi cara <i>backup data (dump database)</i>, <i>import/ export</i>, dan lain-lain yang dilakukan melalui terminal <i>command line</i> maupun <i>interface PHPMyAdmin</i> cara menginstal MySQL sever (XAMPP) cara menggunakan <i>tools IDE</i> seperti Eclipse dan NetBeans cara merancang GUI menggunakan Java Swing. 	<harap diisi dalam %>
13 dan 14	Mampu menggunakan gabungan dari pemrograman berorientasi objek (<i>object oriented programming</i>) dan manajemen sistem basis data (DBMS) untuk membuat suatu aplikasi yang berkualitas dan <i>reliable</i> .	Pemberian tugas proyek menggunakan Java dan DBMS Oracle/ MySQL. Tugas dikerjakan secara berkelompok dengan memperhatikan aspek originalitas, kompleksitas, kualitas, dan dokumentasi perangkat lunak.	Presentasi proyek besar. <mohon diperiksa kembali>	Mahasiswa mampu menerapkan paradigma pemrograman berorientasi objek (OOP) dan manajemen sistem basis data (DBMS) untuk menghasilkan perangkat lunak yang berkualitas dan <i>reliable</i> .	<harap diisi dalam %>

C. RANCANGAN INTERAKSI DOSEN–MAHASISWA

1. Materi pengenalan OOP dan Java, serta peran *class diagram* dalam OOP.

Kemampuan Akhir yang Diharapkan	<ul style="list-style-type: none"> • Memahami konsep pemrograman berorientasi objek serta terminologi-terminologi yang terdapat di dalamnya. • Mampu menggunakan <i>class diagram</i> sebagai salah satu aspek dalam perancangan perangkat lunak. • Mampu menjalankan program dalam bahasa Java dengan <i>compiler</i> yang berbasis <i>command prompt/ terminal</i>. • Memahami definisi <i>variable</i>, tipe data, percabangan, dan perulangan pada Java.
Nama Kajian	<ol style="list-style-type: none"> 1. Konsep dasar pemrograman berorientasi objek. 2. <i>Class diagram</i>. 3. Java.
Nama Strategi	<ul style="list-style-type: none"> • Kuliah yang memuat ceramah, diskusi, dan latihan. • Tugas mingguan. <p><mohon diperiksa kembali></p>
Pekan Penggunaan Strategi (Metode)	1
Deskripsi Singkat Strategi (Metode) pembelajaran	Dosen memberikan ceramah mengenai materi yang diajarkan; diskusi dilakukan di kelas maupun IDEA sebagai media <i>e-learning</i> ; pemberian tugas dilakukan untuk membantu pemahaman mahasiswa. <mohon diperiksa kembali>
RANCANGAN INTERAKSI DOSEN–MAHASISWA	
Aktivitas Dosen	Aktivitas Mahasiswa
<p><harap diisi dengan aktivitas dosen yang pada pertemuan 1 yang sesuai dengan silabus></p> <p>Menjelaskan tentang tujuan pembelajaran dari kegiatan pembelajaran.</p> <p>Mengarahkan mahasiswa untuk melibatkan diri dan aktif dalam kegiatan pembelajaran.</p> <p>Membahas materi.</p>	<p><berisi aksi/ respon mahasiswa terhadap aktivitas dosen yang berada di kolom kiri></p> <p>Menyimak penjelasan dosen.</p> <p>Menyiapkan diri menerima materi yang akan disampaikan.</p> <p>Menyimak dan mencatat hal-hal penting dari materi yang disampaikan oleh dosen.</p>

Mengajukan sejumlah pertanyaan terkait materi yang telah diberikan	Bertanya apabila ada materi yang kurang jelas. Menjawab pertanyaan yang diberikan.
Memberikan tugas sebagai sarana berlatih dan evaluasi diri kepada mahasiswa.	Mengerjakan tugas dengan baik sesuai dengan arahan dosen, tidak melakukan tindak plagiarisme dalam pengerjaan tugas.

2. Materi konsep *constructor*, *class driver*, method *get* dan *set*, serta implementasi *class diagram*.

Kemampuan Akhir yang Diharapkan	<ul style="list-style-type: none"> • Memahami konsep <i>constructor</i> dan cara menginstansiasi <i>object</i> dari <i>class</i> • Memahami <i>class driver</i> • Memahami perbedaan akses <i>public</i> dan <i>private</i> • Memahami <i>method get</i> dan <i>set</i> dan dapat mengimplementasikannya • Mampu mengimplementasikan <i>class diagram</i> dalam bahasa Java • Memahami standar disiplin pemrograman secara umum maupun yang dipakai di OOP.
Nama Kajian	<ol style="list-style-type: none"> 1. <i>Class driver</i>. 2. <i>Constructor</i>. 3. Akses <i>Public</i> dan <i>private</i>. 4. <i>Method Get</i> dan <i>method set</i>. 5. Implementasi <i>class diagram</i> dalam Java.
Nama Strategi	<ul style="list-style-type: none"> • Kuliah yang memuat ceramah, diskusi, dan latihan. • Tugas mingguan. <p><mohon diperiksa kembali></p>
Pekan Penggunaan Strategi (Metode)	2
Deskripsi Singkat Strategi (Metode) pembelajaran	Dosen memberikan ceramah mengenai materi yang diajarkan; diskusi dilakukan di kelas maupun IDEA sebagai media <i>e-learning</i> ; pemberian tugas dilakukan untuk membantu pemahaman mahasiswa. <mohon diperiksa kembali>
RANCANGAN INTERAKSI DOSEN–MAHASISWA	
Aktivitas Dosen	Aktivitas Mahasiswa

<harap diisi dengan aktivitas dosen yang pada pertemuan 1 yang sesuai dengan silabus>	<berisi aksi/ respon mahasiswa terhadap aktivitas dosen yang berada di kolom kiri>
Menjelaskan tentang tujuan pembelajaran dari kegiatan pembelajaran.	Menyimak penjelasan dosen.
Mengarahkan mahasiswa untuk melibatkan diri dan aktif dalam kegiatan pembelajaran.	Menyiapkan diri menerima materi yang akan disampaikan.
Membahas materi.	Menyimak dan mencatat hal-hal penting dari materi yang disampaikan oleh dosen. Bertanya apabila ada materi yang kurang jelas.
Mengajukan sejumlah pertanyaan terkait materi yang telah diberikan	Menjawab pertanyaan yang diberikan.
Memberikan tugas sebagai sarana berlatih dan evaluasi diri kepada mahasiswa.	Mengerjakan tugas dengan baik sesuai dengan arahan dosen, tidak melakukan tindak plagiarisme dalam pengerjaan tugas.

3. Materi *inheritance*, *overriding*, dan *overloading*.

Kemampuan Akhir yang Diharapkan	<ul style="list-style-type: none"> • Memahami konsep <i>inheritance</i>, pewarisan <i>attribute</i> maupun <i>method</i>, dan cara pemanggilan <i>method</i> dari <i>class</i>. • Memahami konsep <i>overriding</i> dan mampu menerapkannya. • Memahami konsep <i>overloading</i> dan mampu menerapkannya.
Nama Kajian	<ol style="list-style-type: none"> 1. <i>Inheritance</i> 2. <i>Overriding</i> 3. <i>Overloading</i>
Nama Strategi	<ul style="list-style-type: none"> • Kuliah yang memuat ceramah, diskusi, dan latihan. • Tugas mingguan. <p><mohon diperiksa kembali></p>
Pertemuan Penggunaan Strategi (Metode)	3
Deskripsi Singkat Strategi (Metode) pembelajaran	Dosen memberikan ceramah mengenai materi yang diajarkan; diskusi dilakukan di kelas maupun IDEA sebagai media <i>e-learning</i> ; pemberian tugas dilakukan untuk membantu pemahaman mahasiswa. <mohon diperiksa kembali>

RANCANGAN INTERAKSI DOSEN–MAHASISWA	
Aktivitas Dosen	Aktivitas Mahasiswa
<p><harap diisi dengan aktivitas dosen yang pada pertemuan 1 yang sesuai dengan silabus></p> <p>Menjelaskan tentang tujuan pembelajaran dari kegiatan pembelajaran.</p> <p>Mengarahkan mahasiswa untuk melibatkan diri dan aktif dalam kegiatan pembelajaran. Membahas materi.</p> <p>Mengajukan sejumlah pertanyaan terkait materi yang telah diberikan</p> <p>Memberikan tugas sebagai sarana berlatih dan evaluasi diri kepada mahasiswa.</p>	<p><berisi aksi/ respon mahasiswa terhadap aktivitas dosen yang berada di kolom kiri></p> <p>Menyimak penjelasan dosen.</p> <p>Menyiapkan diri menerima materi yang akan disampaikan.</p> <p>Menyimak dan mencatat hal-hal penting dari materi yang disampaikan oleh dosen.</p> <p>Bertanya apabila ada materi yang kurang jelas.</p> <p>Menjawab pertanyaan yang diberikan.</p> <p>Mengerjakan tugas dengan baik sesuai dengan arahan dosen, tidak melakukan tindak plagiarisme dalam pengerjaan tugas.</p>

4. Materi relasi antar kelas pada pemrograman berorientasi objek.

Kemampuan Akhir yang Diharapkan	<ul style="list-style-type: none"> Memahami pengertian hubungan antar kelas (agregasi) pada pemrograman berorientasi objek. Mampu menerapkan hubungan antar kelas pada pemrograman berorientasi objek dalam bahasa Java.
Nama Kajian	Relasi antar kelas pada pemrograman berorientasi objek.
Nama Strategi	Kuliah yang memuat ceramah dan diskusi . <mohon diperiksa kembali>
Pertemuan Penggunaan Strategi (Metode)	4
Deskripsi Singkat Strategi (Metode) pembelajaran	Dosen memberikan ceramah mengenai materi yang diajarkan; diskusi dilakukan di kelas maupun IDEA sebagai media <i>e-learning</i> ; pemberian tugas dilakukan untuk membantu pemahaman mahasiswa. <mohon diperiksa kembali>
RANCANGAN INTERAKSI DOSEN–MAHASISWA	

Aktivitas Dosen	Aktivitas Mahasiswa
<p><harap diisi dengan aktivitas dosen yang pada pertemuan 1 yang sesuai dengan silabus></p> <p>Menjelaskan tentang tujuan pembelajaran dari kegiatan pembelajaran. Mengarahkan mahasiswa untuk melibatkan diri dan aktif dalam kegiatan pembelajaran. Membahas materi.</p> <p>Mengajukan sejumlah pertanyaan terkait materi yang telah diberikan Memberikan tugas sebagai sarana berlatih dan evaluasi diri kepada mahasiswa.</p>	<p><berisi aksi/ respon mahasiswa terhadap aktivitas dosen yang berada di kolom kiri></p> <p>Menyimak penjelasan dosen.</p> <p>Menyiapkan diri menerima materi yang akan disampaikan.</p> <p>Menyimak dan mencatat hal-hal penting dari materi yang disampaikan oleh dosen.</p> <p>Bertanya apabila ada materi yang kurang jelas.</p> <p>Menjawab pertanyaan yang diberikan.</p> <p>Mengerjakan tugas dengan baik sesuai dengan arahan dosen, tidak melakukan tindak plagiarisme dalam pengerjaan tugas.</p>

5. Materi relasi antar kelas pada pemrograman berorientasi objek yang melibatkan *array*.

Kemampuan Akhir yang Diharapkan	<ul style="list-style-type: none"> • Memahami konsep hubungan antar kelas (agregasi) yang melibatkan <i>array</i> pada pemrograman berorientasi objek. • Mampu menerapkan hubungan antar kelas yang melibatkan <i>array</i> pada pemrograman berorientasi objek dalam bahasa Java.
Nama Kajian	Relasi antar kelas pada pemrograman berorientasi objek yang melibatkan <i>array</i> .
Nama Strategi	<ul style="list-style-type: none"> • Kuliah yang memuat ceramah, diskusi, dan latihan. • Tugas mingguan. <p><mohon diperiksa kembali></p>
Pertemuan Penggunaan Strategi (Metode)	5
Deskripsi Singkat Strategi (Metode) pembelajaran	Dosen memberikan ceramah mengenai materi yang diajarkan; diskusi dilakukan di kelas maupun IDEA sebagai media <i>e-learning</i> ; pemberian tugas dilakukan untuk membantu pemahaman mahasiswa. <mohon diperiksa kembali>
RANCANGAN INTERAKSI DOSEN–MAHASISWA	

Aktivitas Dosen	Aktivitas Mahasiswa
<p><harap diisi dengan aktivitas dosen yang pada pertemuan 1 yang sesuai dengan silabus></p> <p>Menjelaskan tentang tujuan pembelajaran dari kegiatan pembelajaran.</p> <p>Mengarahkan mahasiswa untuk melibatkan diri dan aktif dalam kegiatan pembelajaran.</p> <p>Membahas materi.</p> <p>Mengajukan sejumlah pertanyaan terkait materi yang telah diberikan</p> <p>Memberikan tugas sebagai sarana berlatih dan evaluasi diri kepada mahasiswa.</p>	<p><berisi aksi/ respon mahasiswa terhadap aktivitas dosen yang berada di kolom kiri></p> <p>Menyimak penjelasan dosen.</p> <p>Menyiapkan diri menerima materi yang akan disampaikan.</p> <p>Menyimak dan mencatat hal-hal penting dari materi yang disampaikan oleh dosen.</p> <p>Bertanya apabila ada materi yang kurang jelas.</p> <p>Menjawab pertanyaan yang diberikan.</p> <p>Mengerjakan tugas dengan baik sesuai dengan arahan dosen, tidak melakukan tindak plagiarisme dalam pengerjaan tugas.</p>

6. Materi *abstract class*, *interface*, dan *exception*.

Kemampuan Akhir yang Diharapkan	<ul style="list-style-type: none"> • Memahami konsep <i>abstract class</i> dan <i>polymorphism</i> dan penerapannya dalam bahasa Java. • Memahami konsep <i>interface</i> dan penerapannya dalam bahasa Java. • Memahami konsep <i>exception try, catch</i>, dan <i>throw</i> pada bahasa Java dan mampu membuat <i>exception</i> pada Java.
Nama Kajian	<ol style="list-style-type: none"> 1. <i>Abstract class</i>. 2. <i>Interface</i>. 3. <i>Exception</i>.
Nama Strategi	<ul style="list-style-type: none"> • Kuliah yang memuat ceramah, diskusi, dan latihan. • Tugas mingguan. <p><mohon diperiksa kembali></p>
Pertemuan Penggunaan Strategi (Metode)	6
Deskripsi Singkat Strategi (Metode) pembelajaran	Dosen memberikan ceramah mengenai materi yang diajarkan; diskusi dilakukan di kelas maupun IDEA sebagai media <i>e-learning</i> ; pemberian tugas dilakukan untuk membantu

	pemahaman mahasiswa. <mohon diperiksa kembali>
RANCANGAN INTERAKSI DOSEN–MAHASISWA	
Aktivitas Dosen	Aktivitas Mahasiswa
<p><harap diisi dengan aktivitas dosen yang pada pertemuan 1 yang sesuai dengan silabus></p> <p>Menjelaskan tentang tujuan pembelajaran dari kegiatan pembelajaran.</p> <p>Mengarahkan mahasiswa untuk melibatkan diri dan aktif dalam kegiatan pembelajaran. Membahas materi.</p> <p>Mengajukan sejumlah pertanyaan terkait materi yang telah diberikan</p> <p>Memberikan tugas sebagai sarana berlatih dan evaluasi diri kepada mahasiswa.</p>	<p><berisi aksi/ respon mahasiswa terhadap aktivitas dosen yang berada di kolom kiri></p> <p>Menyimak penjelasan dosen.</p> <p>Menyiapkan diri menerima materi yang akan disampaikan.</p> <p>Menyimak dan mencatat hal-hal penting dari materi yang disampaikan oleh dosen.</p> <p>Bertanya apabila ada materi yang kurang jelas.</p> <p>Menjawab pertanyaan yang diberikan.</p> <p>Mengerjakan tugas dengan baik sesuai dengan arahan dosen, tidak melakukan tindak plagiarisme dalam pengerjaan tugas.</p>

7. Materi Java API DOCS, Java *library*, dan IO *stream* pada Java.

Kemampuan Akhir yang Diharapkan	<ul style="list-style-type: none"> • Memahami penggunaan Java API DOCS dan dapat melakukan eksplorasi secara mandiri. • Mengenal Java <i>library</i> dan dapat memakainya sesuai dengan kebutuhan pemrograman yang sesuai. • Memahami IO <i>stream</i> pada Java dan dapat menggunakannya.
Nama Kajian	<ol style="list-style-type: none"> 1. Java API DOCS. 2. Java <i>library</i>. 3. IO <i>stream</i> pada Java.
Nama Strategi	<ul style="list-style-type: none"> • Kuliah yang memuat ceramah, diskusi, dan latihan. • Tugas mingguan. <p><mohon diperiksa kembali></p>
Pertemuan Penggunaan Strategi (Metode)	7
Deskripsi Singkat Strategi (Metode) pembelajaran	Dosen memberikan ceramah mengenai materi yang diajarkan; diskusi dilakukan di kelas

	maupun IDEA sebagai media <i>e-learning</i> ; pemberian tugas dilakukan untuk membantu pemahaman mahasiswa. <mohon diperiksa kembali>
RANCANGAN INTERAKSI DOSEN–MAHASISWA	
Aktivitas Dosen	Aktivitas Mahasiswa
<p><harap diisi dengan aktivitas dosen yang pada pertemuan 1 yang sesuai dengan silabus></p> <p>Menjelaskan tentang tujuan pembelajaran dari kegiatan pembelajaran.</p> <p>Mengarahkan mahasiswa untuk melibatkan diri dan aktif dalam kegiatan pembelajaran. Membahas materi.</p> <p>Mengajukan sejumlah pertanyaan terkait materi yang telah diberikan</p> <p>Memberikan tugas sebagai sarana berlatih dan evaluasi diri kepada mahasiswa.</p>	<p><berisi aksi/ respon mahasiswa terhadap aktivitas dosen yang berada di kolom kiri></p> <p>Menyimak penjelasan dosen.</p> <p>Menyiapkan diri menerima materi yang akan disampaikan.</p> <p>Menyimak dan mencatat hal-hal penting dari materi yang disampaikan oleh dosen.</p> <p>Bertanya apabila ada materi yang kurang jelas.</p> <p>Menjawab pertanyaan yang diberikan.</p> <p>Mengerjakan tugas dengan baik sesuai dengan arahan dosen, tidak melakukan tindak plagiarisme dalam pengerjaan tugas.</p>

8. Materi GUI dan pemrograman berorientasi objek dengan C++.

Kemampuan Akhir yang Diharapkan	<ul style="list-style-type: none"> • Mampu menggunakan GUI yang meliputi <i>AWT, swing, dan action listener</i>. • Mampu membuat GUI tanpa memakai IDE. • Memahami konsep-konsep dasar <i>JFrame, JButton, JLabel, JTextField</i>, dan lain-lain serta dapat menerapkannya. • Mampu menerapkan paradigma orientasi objek dalam bahasa C++. • Mengenal <i>standard library</i> pada C++ dan mampu memakainya.
Nama Kajian	<ol style="list-style-type: none"> 1. <i>Graphical user interface</i> (GUI). 2. Pemrograman berorientasi objek dengan C++.
Nama Strategi	<ul style="list-style-type: none"> • Kuliah yang memuat ceramah, diskusi, dan latihan.

	<ul style="list-style-type: none"> Tugas mingguan. <p><mohon diperiksa kembali></p>
Pertemuan Penggunaan Strategi (Metode)	8
Deskripsi Singkat Strategi (Metode) pembelajaran	Dosen memberikan ceramah mengenai materi yang diajarkan; diskusi dilakukan di kelas maupun IDEA sebagai media <i>e-learning</i> ; pemberian tugas dilakukan untuk membantu pemahaman mahasiswa. <mohon diperiksa kembali>
RANCANGAN INTERAKSI DOSEN–MAHASISWA	
Aktivitas Dosen	Aktivitas Mahasiswa
<p><harap diisi dengan aktivitas dosen yang pada pertemuan 1 yang sesuai dengan silabus></p> <p>Menjelaskan tentang tujuan pembelajaran dari kegiatan pembelajaran.</p> <p>Mengarahkan mahasiswa untuk melibatkan diri dan aktif dalam kegiatan pembelajaran.</p> <p>Membahas materi.</p> <p>Mengajukan sejumlah pertanyaan terkait materi yang telah diberikan</p> <p>Memberikan tugas sebagai sarana berlatih dan evaluasi diri kepada mahasiswa.</p>	<p><berisi aksi/ respon mahasiswa terhadap aktivitas dosen yang berada di kolom kiri></p> <p>Menyimak penjelasan dosen.</p> <p>Menyiapkan diri menerima materi yang akan disampaikan.</p> <p>Menyimak dan mencatat hal-hal penting dari materi yang disampaikan oleh dosen.</p> <p>Bertanya apabila ada materi yang kurang jelas.</p> <p>Menjawab pertanyaan yang diberikan.</p> <p>Mengerjakan tugas dengan baik sesuai dengan arahan dosen, tidak melakukan tindak plagiarisme dalam pengerjaan tugas.</p>

9. Materi ER *diagram* serta pengenalan ORACLE dan SQL.

Kemampuan Akhir yang Diharapkan	<ul style="list-style-type: none"> Memahami ER <i>model</i> dari suatu perangkat lunak. Mampu membuat ER <i>diagram</i> dalam proses perancangan perangkat lunak. Mengenal ORACLE dan dapat memakainya. Mengenal SQL dan dapat memakai <i>query-query</i> dasar pada SQL.
Nama Kajian	<ol style="list-style-type: none"> ER <i>model</i> dari suatu perangkat lunak. Pengenalan ORACLE. Dasar-dasar <i>query</i> pada SQL.

Nama Strategi	<ul style="list-style-type: none"> Kuliah yang memuat ceramah, diskusi, dan latihan. Tugas mingguan. <p><mohon diperiksa kembali></p>
Pertemuan Penggunaan Strategi (Metode)	9
Deskripsi Singkat Strategi (Metode) pembelajaran	Dosen memberikan ceramah mengenai materi yang diajarkan; diskusi dilakukan di kelas maupun IDEA sebagai media <i>e-learning</i> ; pemberian tugas dilakukan untuk membantu pemahaman mahasiswa. <mohon diperiksa kembali>
RANCANGAN INTERAKSI DOSEN–MAHASISWA	
Aktivitas Dosen	Aktivitas Mahasiswa
<p><harap diisi dengan aktivitas dosen yang pada pertemuan 1 yang sesuai dengan silabus></p> <p>Menjelaskan tentang tujuan pembelajaran dari kegiatan pembelajaran.</p> <p>Mengarahkan mahasiswa untuk melibatkan diri dan aktif dalam kegiatan pembelajaran. Membahas materi.</p> <p>Mengajukan sejumlah pertanyaan terkait materi yang telah diberikan</p> <p>Memberikan tugas sebagai sarana berlatih dan evaluasi diri kepada mahasiswa.</p>	<p><berisi aksi/ respon mahasiswa terhadap aktivitas dosen yang berada di kolom kiri></p> <p>Menyimak penjelasan dosen.</p> <p>Menyiapkan diri menerima materi yang akan disampaikan.</p> <p>Menyimak dan mencatat hal-hal penting dari materi yang disampaikan oleh dosen.</p> <p>Bertanya apabila ada materi yang kurang jelas.</p> <p>Menjawab pertanyaan yang diberikan.</p> <p>Mengerjakan tugas dengan baik sesuai dengan arahan dosen, tidak melakukan tindak plagiarisme dalam pengerjaan tugas.</p>

10. Materi *aggregate function* pada SQL, klausa GROUP BY, dan klausa JOIN.

Kemampuan Akhir yang Diharapkan	<ul style="list-style-type: none"> Memahami dan mengenal contoh dari <i>aggregate function</i> sebagai <i>query</i> SQL yang lebih kompleks. Memahami klausa GROUP BY dan dapat menerapkannya. Memahami klausa JOIN dan dapat menerapkannya.
Nama Kajian	1. <i>Aggregate function</i> sebagai <i>query</i> SQL yang lebih kompleks (COUNT, MAX, MIN, SUM,

	AVG). 2. Klausur GROUP BY dan implementasinya. 3. Klausur JOIN dan implementasinya.
Nama Strategi	<ul style="list-style-type: none"> Kuliah yang memuat ceramah, diskusi, dan latihan. Tugas mingguan. <p><mohon diperiksa kembali></p>
Pertemuan Penggunaan Strategi (Metode)	10
Deskripsi Singkat Strategi (Metode) pembelajaran	Dosen memberikan ceramah mengenai materi yang diajarkan; diskusi dilakukan di kelas maupun IDEA sebagai media <i>e-learning</i> ; pemberian tugas dilakukan untuk membantu pemahaman mahasiswa. <mohon diperiksa kembali>
RANCANGAN INTERAKSI DOSEN–MAHASISWA	
Aktivitas Dosen	Aktivitas Mahasiswa
<p><harap diisi dengan aktivitas dosen yang pada pertemuan 1 yang sesuai dengan silabus></p> <p>Menjelaskan tentang tujuan pembelajaran dari kegiatan pembelajaran.</p> <p>Mengarahkan mahasiswa untuk melibatkan diri dan aktif dalam kegiatan pembelajaran.</p> <p>Membahas materi.</p> <p>Mengajukan sejumlah pertanyaan terkait materi yang telah diberikan</p> <p>Memberikan tugas sebagai sarana berlatih dan evaluasi diri kepada mahasiswa.</p>	<p><berisi aksi/ respon mahasiswa terhadap aktivitas dosen yang berada di kolom kiri></p> <p>Menyimak penjelasan dosen.</p> <p>Menyiapkan diri menerima materi yang akan disampaikan.</p> <p>Menyimak dan mencatat hal-hal penting dari materi yang disampaikan oleh dosen.</p> <p>Bertanya apabila ada materi yang kurang jelas.</p> <p>Menjawab pertanyaan yang diberikan.</p> <p>Mengerjakan tugas dengan baik sesuai dengan arahan dosen, tidak melakukan tindak plagiarisme dalam pengerjaan tugas.</p>

11. Materi DML, DDL, dan *statement* DROP TABLE.

Kemampuan Akhir yang Diharapkan	<ul style="list-style-type: none"> Memahami perintah-perintah pada <i>data manipulation language</i> (DML) dan dapat memakainya. Memahami perintah-perintah pada <i>data definition language</i> (DDL) dan dapat memakainya.
---------------------------------	--

	<ul style="list-style-type: none"> Memahami <i>statement</i> DROP TABLE.
Nama Kajian	<ol style="list-style-type: none"> <i>Data manipulation language</i> (DML). <i>Data definition language</i> (DDL). <i>Statement</i> DROP TABLE.
Nama Strategi	<ul style="list-style-type: none"> Kuliah yang memuat ceramah, diskusi, dan latihan. Tugas mingguan. <p><mohon diperiksa kembali></p>
Pertemuan Penggunaan Strategi (Metode)	11
Deskripsi Singkat Strategi (Metode) pembelajaran	Dosen memberikan ceramah mengenai materi yang diajarkan; diskusi dilakukan di kelas maupun IDEA sebagai media <i>e-learning</i> ; pemberian tugas dilakukan untuk membantu pemahaman mahasiswa. <mohon diperiksa kembali>
RANCANGAN INTERAKSI DOSEN–MAHASISWA	
Aktivitas Dosen	Aktivitas Mahasiswa
<p><harap diisi dengan aktivitas dosen yang pada pertemuan 1 yang sesuai dengan silabus></p> <p>Menjelaskan tentang tujuan pembelajaran dari kegiatan pembelajaran.</p> <p>Mengarahkan mahasiswa untuk melibatkan diri dan aktif dalam kegiatan pembelajaran. Membahas materi.</p> <p>Mengajukan sejumlah pertanyaan terkait materi yang telah diberikan</p> <p>Memberikan tugas sebagai sarana berlatih dan evaluasi diri kepada mahasiswa.</p>	<p><berisi aksi/ respon mahasiswa terhadap aktivitas dosen yang berada di kolom kiri></p> <p>Menyimak penjelasan dosen.</p> <p>Menyiapkan diri menerima materi yang akan disampaikan.</p> <p>Menyimak dan mencatat hal-hal penting dari materi yang disampaikan oleh dosen.</p> <p>Bertanya apabila ada materi yang kurang jelas.</p> <p>Menjawab pertanyaan yang diberikan.</p> <p>Mengerjakan tugas dengan baik sesuai dengan arahan dosen, tidak melakukan tindak plagiarisme dalam pengerjaan tugas.</p>

12. Materi *database administration*, JDBC, *tools* IDE, dan GUI memakai Java Swing.

Kemampuan Akhir yang Diharapkan	<ul style="list-style-type: none"> Memahami aspek-aspek pada <i>database administration</i>. Mampu menggunakan <i>tools</i> IDE untuk membuat program berorientasi objek.
---------------------------------	---

	<ul style="list-style-type: none"> Mampu merancang GUI menggunakan Java Swing.
Nama Kajian	<ol style="list-style-type: none"> Database administration. Java database connectivity (JDBC). Tools IDE: Eclipse dan NetBeans. GUI (Java Swing).
Nama Strategi	<ul style="list-style-type: none"> Kuliah yang memuat ceramah, diskusi, dan latihan. Tugas mingguan. <p><mohon diperiksa kembali></p>
Pertemuan Penggunaan Strategi (Metode)	12
Deskripsi Singkat Strategi (Metode) pembelajaran	Dosen memberikan ceramah mengenai materi yang diajarkan; diskusi dilakukan di kelas maupun IDEA sebagai media <i>e-learning</i> ; pemberian tugas dilakukan untuk membantu pemahaman mahasiswa. <mohon diperiksa kembali>
RANCANGAN INTERAKSI DOSEN–MAHASISWA	
Aktivitas Dosen	Aktivitas Mahasiswa
<p><harap diisi dengan aktivitas dosen yang pada pertemuan 1 yang sesuai dengan silabus></p> <p>Menjelaskan tentang tujuan pembelajaran dari kegiatan pembelajaran.</p> <p>Mengarahkan mahasiswa untuk melibatkan diri dan aktif dalam kegiatan pembelajaran.</p> <p>Membahas materi.</p> <p>Mengajukan sejumlah pertanyaan terkait materi yang telah diberikan</p> <p>Memberikan tugas sebagai sarana berlatih dan evaluasi diri kepada mahasiswa.</p>	<p><berisi aksi/ respon mahasiswa terhadap aktivitas dosen yang berada di kolom kiri></p> <p>Menyimak penjelasan dosen.</p> <p>Menyiapkan diri menerima materi yang akan disampaikan.</p> <p>Menyimak dan mencatat hal-hal penting dari materi yang disampaikan oleh dosen.</p> <p>Bertanya apabila ada materi yang kurang jelas.</p> <p>Menjawab pertanyaan yang diberikan.</p> <p>Mengerjakan tugas dengan baik sesuai dengan arahan dosen, tidak melakukan tindak plagiarisme dalam pengerjaan tugas.</p>

13. Materi tugas besar pembuatan perangkat lunak yang menggabungkan OOP dan DBMS.

Kemampuan Akhir yang Diharapkan	Mampu menggunakan gabungan dari pemrograman berorientasi objek (<i>object</i>
---------------------------------	--

	<i>oriented programming</i>) dan manajemen sistem basis data (DBMS) untuk membuat suatu aplikasi yang berkualitas dan <i>reliable</i> .
Nama Kajian	Pemberian tugas proyek menggunakan Java dan DBMS Oracle/ MySQL. Tugas dikerjakan secara berkelompok dengan memperhatikan aspek originalitas, kompleksitas, kualitas, dan dokumentasi perangkat lunak.
Nama Strategi	Presentasi proyek besar. <mohon diperiksa kembali>
Pertemuan Penggunaan Strategi (Metode)	13-14
Deskripsi Singkat Strategi (Metode) pembelajaran	Dosen memberikan tugas besar yang berupa proyek pembuatan perangkat lunak. Perangkat lunak yang dibangun harus mengkombinasikan OOP dan DBMS. Penilaian perangkat lunak dilakukan dengan memperhatikan aspek originalitas, kompleksitas, kualitas, dan dokumentasi dari perangkat lunak tersebut. <mohon diperiksa kembali>
RANCANGAN INTERAKSI DOSEN–MAHASISWA	
Aktivitas Dosen	Aktivitas Mahasiswa
<p><harap diisi dengan aktivitas dosen yang pada pertemuan 1 yang sesuai dengan silabus></p> <p>Menjelaskan tentang tujuan pembelajaran dari kegiatan pembelajaran.</p> <p>Mengarahkan mahasiswa untuk melibatkan diri dan aktif dalam kegiatan pembelajaran. Membahas materi.</p> <p>Mengajukan sejumlah pertanyaan terkait materi yang telah diberikan</p> <p>Memberikan tugas sebagai sarana berlatih dan evaluasi diri kepada mahasiswa.</p>	<p><berisi aksi/ respon mahasiswa terhadap aktivitas dosen yang berada di kolom kiri></p> <p>Menyimak penjelasan dosen.</p> <p>Menyiapkan diri menerima materi yang akan disampaikan.</p> <p>Menyimak dan mencatat hal-hal penting dari materi yang disampaikan oleh dosen.</p> <p>Bertanya apabila ada materi yang kurang jelas.</p> <p>Menjawab pertanyaan yang diberikan.</p> <p>Mengerjakan tugas dengan baik sesuai dengan arahan dosen, tidak melakukan tindak plagiarisme dalam pengerjaan tugas.</p>

D. RANCANGAN TUGAS

<mohon dilengkapi oleh tim dosen pengajar dengan rancangan tugas yang akan diberikan kepada mahasiswa>

1. Materi pengenalan OOP dan Java, serta peran *class diagram* dalam OOP.

Kode mata Kuliah	IKG2I4
Nama Mata Kuliah	<i>Software Project 1</i>
Kemampuan Akhir yang Diharapkan	<ul style="list-style-type: none"> • Memahami konsep pemrograman berorientasi objek serta terminologi-terminologi yang terdapat di dalamnya. • Mampu menggunakan <i>class diagram</i> sebagai salah satu aspek dalam perancangan perangkat lunak. • Mampu menjalankan program dalam bahasa Java dengan <i>compiler</i> yang berbasis <i>command prompt/terminal</i>. • Memahami definisi <i>variable</i>, tipe data, percabangan, dan perulangan pada Java.
Minggu/Pertemuan ke	1 (?) <mohon diisi sesuai rancangan tugas yang diberikan kepada mahasiswa>
Tugas ke	1 (?) <mohon diisi sesuai rancangan tugas yang diberikan kepada mahasiswa>
<p>1. Tujuan tugas:</p> <p>2. Uraian Tugas:</p> <ol style="list-style-type: none"> a. Objek garapan: b. Yang harus dikerjakan dan batasan-batasan: c. Metode/ cara pengerjaan, acuan yang digunakan: d. Deskripsi luaran (<i>output</i>) tugas yang dihasilkan/ dikerjakan: <p>3. Kriteria penilaian:</p>	

2. Materi konsep *constructor*, *class driver*, method *get* dan *set*, serta implementasi *class diagram*.

Kode mata Kuliah	IKG2I4
Nama Mata Kuliah	<i>Software Project 1</i>
Kemampuan Akhir yang Diharapkan	<ul style="list-style-type: none"> • Memahami konsep <i>constructor</i> dan cara menginstansiasi <i>object</i> dari <i>class</i> • Memahami <i>class driver</i> • Memahami perbedaan akses <i>public</i> dan <i>private</i> • Memahami <i>method get</i> dan <i>set</i> dan dapat

	<p>mengimplementasikannya</p> <ul style="list-style-type: none"> • Mampu mengimplementasikan <i>class diagram</i> dalam bahasa Java • Memahami standar disiplin pemrograman secara umum maupun yang dipakai di OOP.
Minggu/Pertemuan ke	2 (?) <mohon diisi sesuai rancangan tugas yang diberikan kepada mahasiswa>
Tugas ke	<mohon diisi sesuai rancangan tugas yang diberikan kepada mahasiswa>
<p>1. Tujuan tugas:</p> <p>2. Uraian Tugas:</p> <p>a. Objek garapan: mengimplementasikan <i>class diagram</i> ke dalam Java, meliputi <i>class diagram</i> segiempat, segitiga, lingkaran, polygon, titik, titik 3D, <i>time</i>, <i>date</i> (tidak di-import dari java.util.Date).</p> <p>b. Yang harus dikerjakan dan batasan-batasan:</p> <p>c. Metode/ cara pengerjaan, acuan yang digunakan:</p> <p>d. Deskripsi luaran (<i>output</i>) tugas yang dihasilkan/ dikerjakan:</p> <p>3. Kriteria penilaian:</p>	

3. Materi *inheritance*, *overriding*, dan *overloading*.

Kode mata Kuliah	IKG214
Nama Mata Kuliah	<i>Software Project 1</i>
Kemampuan Akhir yang Diharapkan	<ul style="list-style-type: none"> • Memahami konsep <i>inheritance</i>, pewarisan <i>attribute</i> maupun <i>method</i>, dan cara pemanggilan <i>method</i> dari <i>class</i>. • Memahami konsep <i>overriding</i> dan mampu menerapkannya. • Memahami konsep <i>overloading</i> dan mampu menerapkannya.
Minggu/ Pertemuan ke	3 (?) <mohon diisi sesuai rancangan tugas yang diberikan kepada mahasiswa>
Tugas ke	<mohon diisi sesuai rancangan tugas yang diberikan kepada mahasiswa>
<p>1. Tujuan tugas:</p> <p>2. Uraian Tugas:</p> <p>a. Objek garapan: mengimplementasikan <i>class diagram</i> yang diberikan (<i>inheritance</i> balok, bola, prisma, limas, kerucut, titik 3D, dan sebagainya) dalam bahasa Java.</p> <p>b. Yang harus dikerjakan dan batasan-batasan:</p> <p>c. Metode/ cara pengerjaan, acuan yang digunakan:</p>	

d. Deskripsi luaran (*output*) tugas yang dihasilkan/ dikerjakan:

3. Kriteria penilaian:

4. Materi relasi antar kelas pada pemrograman berorientasi objek.

Kode mata Kuliah	IKG214
Nama Mata Kuliah	<i>Software Project 1</i>
Kemampuan Akhir yang Diharapkan	<ul style="list-style-type: none"> • Memahami pengertian hubungan antar kelas (agregasi) pada pemrograman berorientasi objek. • Mampu menerapkan hubungan antar kelas pada pemrograman berorientasi objek dalam bahasa Java.
Minggu/Pertemuan ke	4 (?) <mohon diisi sesuai rancangan tugas yang diberikan kepada mahasiswa>
Tugas ke	<mohon diisi sesuai rancangan tugas yang diberikan kepada mahasiswa>
<p>1. Tujuan tugas:</p> <p>2. Uraian Tugas:</p> <p>a. Objek garapan: mengimpelentasikan <i>class diagram</i> Lingkaran-Titik, Titik-Garis, PulseCounter, TitikSegiempat, dan ArrayMahasiswa ke dalam bahasa Java.</p> <p>b. Yang harus dikerjakan dan batasan-batasan:</p> <p>c. Metode/ cara pengerjaan, acuan yang digunakan:</p> <p>d. Deskripsi luaran (<i>output</i>) tugas yang dihasilkan/ dikerjakan:</p> <p>3. Kriteria penilaian:</p>	

5. Materi relasi antar kelas pada pemrograman berorientasi objek yang melibatkan *array*.

Kode mata Kuliah	IKG214
Nama Mata Kuliah	<i>Software Project 1</i>
Kemampuan Akhir yang Diharapkan	<ul style="list-style-type: none"> • Memahami konsep hubungan antar kelas (agregasi) yang melibatkan <i>array</i> pada pemrograman berorientasi objek. • Mampu menerapkan hubungan antar kelas yang melibatkan <i>array</i> pada pemrograman berorientasi objek dalam bahasa Java.
Minggu/Pertemuan ke	5 (?) <mohon diisi sesuai rancangan tugas yang diberikan kepada mahasiswa>
Tugas ke	<mohon diisi sesuai rancangan tugas yang diberikan kepada mahasiswa>

	kepada mahasiswa>
<p>1. Tujuan tugas:</p> <p>2. Uraian Tugas:</p> <p>a. Objek garapan: agregasi <i>array</i>, mengimplementasikan <i>class diagram Array of Titik, Array of Dosen, Array of Mahasiswa</i>, dan studi kasus lainnya.</p> <p>b. Yang harus dikerjakan dan batasan-batasan:</p> <p>c. Metode/ cara pengerjaan, acuan yang digunakan:</p> <p>d. Deskripsi luaran (<i>output</i>) tugas yang dihasilkan/ dikerjakan:</p> <p>3. Kriteria penilaian:</p>	

6. Materi *class, interface, dan exception*.

Kode mata Kuliah	IKG214
Nama Mata Kuliah	<i>Software Project 1</i>
Kemampuan Akhir yang Diharapkan	<ul style="list-style-type: none"> • Memahami konsep <i>abstract class</i> dan <i>polymorphism</i> dan penerapannya dalam bahasa Java. • Memahami konsep <i>interface</i> dan penerapannya dalam bahasa Java. • Memahami konsep <i>exception try, catch, dan throw</i> pada bahasa Java dan mampu membuat <i>exception</i> pada Java.
Minggu/Pertemuan ke	6 (?) <mohon diisi sesuai rancangan tugas yang diberikan kepada mahasiswa>
Tugas ke	<mohon diisi sesuai rancangan tugas yang diberikan kepada mahasiswa>
<p>1. Tujuan tugas:</p> <p>2. Uraian Tugas:</p> <p>a. Objek garapan:</p> <p>b. Yang harus dikerjakan dan batasan-batasan:</p> <p>c. Metode/ cara pengerjaan, acuan yang digunakan:</p> <p>d. Deskripsi luaran (<i>output</i>) tugas yang dihasilkan/ dikerjakan:</p> <p>3. Kriteria penilaian:</p>	

7. Materi Java API DOCS, Java *library*, dan IO *stream* pada Java.

Kode mata Kuliah	IKG214
Nama Mata Kuliah	<i>Software Project 1</i>

Kemampuan Akhir yang Diharapkan	<ul style="list-style-type: none"> • Memahami penggunaan Java API DOCS dan dapat melakukan eksplorasi secara mandiri. • Mengenal Java <i>library</i> dan dapat memakainya sesuai dengan kebutuhan pemrograman yang sesuai. • Memahami IO <i>stream</i> pada Java dan dapat menggunakannya.
Minggu/Pertemuan ke	7 (?) <mohon diisi sesuai rancangan tugas yang diberikan kepada mahasiswa>
Tugas ke	<mohon diisi sesuai rancangan tugas yang diberikan kepada mahasiswa>
<p>1. Tujuan tugas:</p> <p>2. Uraian Tugas:</p> <p>a. Objek garapan:</p> <p>b. Yang harus dikerjakan dan batasan-batasan:</p> <p>c. Metode/ cara pengerjaan, acuan yang digunakan:</p> <p>d. Deskripsi luaran (<i>output</i>) tugas yang dihasilkan/ dikerjakan:</p> <p>3. Kriteria penilaian:</p>	

8. Materi GUI dan pemrograman berorientasi objek dengan C++.

Kode mata Kuliah	IKG214
Nama Mata Kuliah	<i>Software Project 1</i>
Kemampuan Akhir yang Diharapkan	<ul style="list-style-type: none"> • Mampu menggunakan GUI yang meliputi AWT, <i>swing</i>, dan <i>action listener</i>. • Mampu membuat GUI tanpa memakai IDE. • Memahami konsep-konsep dasar <i>JFrame</i>, <i>JButton</i>, <i>JLabel</i>, <i>JTextField</i>, dan lain-lain serta dapat menerapkannya. • Mampu menerapkan paradigma orientasi objek dalam bahasa C++. • Mengenal <i>standard library</i> pada C++ dan mampu memakainya.
Minggu/Pertemuan ke	8 (?) <mohon diisi sesuai rancangan tugas yang diberikan kepada mahasiswa>
Tugas ke	<mohon diisi sesuai rancangan tugas yang diberikan kepada mahasiswa>
<p>1. Tujuan tugas:</p> <p>2. Uraian Tugas:</p> <p>a. Objek garapan:</p>	

<p>b. Yang harus dikerjakan dan batasan-batasan:</p> <p>c. Metode/ cara pengerjaan, acuan yang digunakan:</p> <p>d. Deskripsi luaran (<i>output</i>) tugas yang dihasilkan/ dikerjakan:</p> <p>3. Kriteria penilaian:</p>
--

9. Materi ER *diagram* serta pengenalan ORACLE dan SQL.

Kode mata Kuliah	IKG2I4
Nama Mata Kuliah	<i>Software Project 1</i>
Kemampuan Akhir yang Diharapkan	<ul style="list-style-type: none"> • Memahami ER <i>model</i> dari suatu perangkat lunak. • Mampu membuat ER <i>diagram</i> dalam proses perancangan perangkat lunak. • Mengenal ORACLE dan dapat memakainya. • Mengenal SQL dan dapat memakai <i>query-query</i> dasar pada SQL.
Minggu/Pertemuan ke	9 (?) <mohon diisi sesuai rancangan tugas yang diberikan kepada mahasiswa>
Tugas ke	<mohon diisi sesuai rancangan tugas yang diberikan kepada mahasiswa>
<p>1. Tujuan tugas:</p> <p>2. Uraian Tugas:</p> <p>a. Objek garapan:</p> <p>b. Yang harus dikerjakan dan batasan-batasan:</p> <p>c. Metode/ cara pengerjaan, acuan yang digunakan:</p> <p>d. Deskripsi luaran (<i>output</i>) tugas yang dihasilkan/ dikerjakan:</p> <p>3. Kriteria penilaian:</p>	

10. Materi *aggregate function* pada SQL, klausa GROUP BY, dan klausa JOIN.

Kode mata Kuliah	IKG2I4
Nama Mata Kuliah	<i>Software Project 1</i>
Kemampuan Akhir yang Diharapkan	<ul style="list-style-type: none"> • Memahami dan mengenal contoh dari <i>aggregate function</i> sebagai <i>query</i> SQL yang lebih kompleks. • Memahami klausa GROUP BY dan dapat menerapkannya. • Memahami klausa JOIN dan dapat menerapkannya.
Minggu/Pertemuan ke	10 (?) <mohon diisi sesuai rancangan tugas yang>

	diberikan kepada mahasiswa>
Tugas ke	<mohon diisi sesuai rancangan tugas yang diberikan kepada mahasiswa>
<p>1. Tujuan tugas:</p> <p>2. Uraian Tugas:</p> <p>a. Objek garapan:</p> <p>b. Yang harus dikerjakan dan batasan-batasan:</p> <p>c. Metode/ cara pengerjaan, acuan yang digunakan:</p> <p>d. Deskripsi luaran (<i>output</i>) tugas yang dihasilkan/ dikerjakan:</p> <p>3. Kriteria penilaian:</p>	

11. Materi DML, DDL, dan *statement* DROP TABLE.

Kode mata Kuliah	IKG2I4
Nama Mata Kuliah	<i>Software Project 1</i>
Kemampuan Akhir yang Diharapkan	<ul style="list-style-type: none"> • Memahami perintah-perintah pada <i>data manipulation language</i> (DML) dan dapat memakainya. • Memahami perintah-perintah pada <i>data definition language</i> (DDL) dan dapat memakainya. • Memahami <i>statement</i> DROP TABLE.
Minggu/Pertemuan ke	11 (?) <mohon diisi sesuai rancangan tugas yang diberikan kepada mahasiswa>
Tugas ke	<mohon diisi sesuai rancangan tugas yang diberikan kepada mahasiswa>
<p>1. Tujuan tugas:</p> <p>2. Uraian Tugas:</p> <p>a. Objek garapan:</p> <p>b. Yang harus dikerjakan dan batasan-batasan:</p> <p>c. Metode/ cara pengerjaan, acuan yang digunakan:</p> <p>d. Deskripsi luaran (<i>output</i>) tugas yang dihasilkan/ dikerjakan:</p> <p>3. Kriteria penilaian:</p>	

12. Materi database administration, JDBC, *tools* IDE, dan GUI memakai Java Swing.

Kode mata Kuliah	IKG2I4
Nama Mata Kuliah	<i>Software Project 1</i>
Kemampuan Akhir yang Diharapkan	• Memahami aspek-aspek pada <i>database</i>

	<i>administration.</i> <ul style="list-style-type: none"> • Mampu menggunakan <i>tools</i> IDE untuk membuat program berorientasi objek. • Mampu merancang GUI menggunakan Java Swing.
Minggu/Pertemuan ke	12 (?) <mohon diisi sesuai rancangan tugas yang diberikan kepada mahasiswa>
Tugas ke	<mohon diisi sesuai rancangan tugas yang diberikan kepada mahasiswa>
<p>1. Tujuan tugas:</p> <p>2. Uraian Tugas:</p> <p>a. Objek garapan:</p> <p>b. Yang harus dikerjakan dan batasan-batasan:</p> <p>c. Metode/ cara pengerjaan, acuan yang digunakan:</p> <p>d. Deskripsi luaran (<i>output</i>) tugas yang dihasilkan/ dikerjakan:</p> <p>3. Kriteria penilaian:</p>	

13. Materi pembuatan tugas besar (projek) perangkat lunak yang menggabungkan OOP dan DBMS.

Kode mata Kuliah	IKG214
Nama Mata Kuliah	<i>Software Project 1</i>
Kemampuan Akhir yang Diharapkan	Mampu menggunakan gabungan dari pemrograman berorientasi objek (<i>object oriented programming</i>) dan manajemen sistem basis data (DBMS) untuk membuat suatu aplikasi yang berkualitas dan <i>reliable</i> .
Minggu/Pertemuan ke	13-14 (?) <mohon diisi sesuai rancangan tugas yang diberikan kepada mahasiswa>
Tugas ke	<mohon diisi sesuai rancangan tugas yang diberikan kepada mahasiswa>
<p>1. Tujuan tugas:</p> <p>2. Uraian Tugas:</p> <p>a. Objek garapan:</p> <p>b. Yang harus dikerjakan dan batasan-batasan:</p> <p>c. Metode/ cara pengerjaan, acuan yang digunakan:</p> <p>d. Deskripsi luaran (<i>output</i>) tugas yang dihasilkan/ dikerjakan:</p> <p>3. Kriteria penilaian:</p>	

E. PENILAIAN DENGAN RUBRIK

<mohon dilengkapi oleh tim dosen pengajar dengan deskripsi penilaian yang sesuai dengan perkuliahan yang dilakukan>

Jenjang (Grade)	Angka (Skor)	Deskripsi Perilaku (Indikator)

F. PENENTUAN NILAI AKHIR MATA KULIAH

Berikut adalah penentuan indeks nilai akhir mata kuliah yang digunakan,

Nilai Skor Matakuliah (NSM)	Nilai Mata Kuliah (NMK)
$80 < \text{NSM}$	A
$70 < \text{NSM} \leq 80$	AB
$65 < \text{NSM} \leq 70$	B
$60 < \text{NSM} \leq 65$	BC
$50 < \text{NSM} \leq 60$	C
$40 < \text{NSM} \leq 50$	D
$\text{NSM} \leq 40$	E