|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | **POLITEKNIK MEDICA FARMA HUSADA MATARAM,**  **PRODI D4 TEKNOLOGI REKAYASA PERANGKAT LUNAK** | | | | | | | | | | | | | | | | | **Kode Dokumen** | |
| **RENCANA PEMBELAJARAN SEMESTER** | | | | | | | | | | | | | | | | | | | | |
| **MATA KULIAH (MK)** | | | | | | **KODE** | | | **Rumpun MK** | | | | | **BOBOT (sks)** | | | **SEMESTER** | | **Tgl Penyusunan** | |
| **PEMROGRAMAN BERORINTASI OBJEK** | | | | | | RPL401 | | |  | | | | | **T=2** | | **P=1** | 3 | |  | |
| **OTORISASI** | | | | | | **Pengembang RPS** | | | | | **Koordinator** | | | | | | **Ketua PRODI** | | | |
|  | | | | |  | | | | | | **Hendra Setiawan, M.Kom.** | | | |
| **Capaian Pembelajaran (CP)** | | **CPL-PRODI yang dibebankan pada MK** | | | | | | | | |  | | | | | | | | | |
| CPL -1 | | | Memiliki pengetahuan komprehensip tentang teori, prinsip dan konsep dasar rekayasa perangkat lunak | | | | | | | | | | | | | | | |
| CPL -2 | | | Menguasai konsep pengembangan rekayasa perangkat lunak | | | | | | | | | | | | | | | |
| CPL -3 | | | Mampu beradaptasi terhadap penggunaan metode baru pada konteks permasalahan yang dinamis | | | | | | | | | | | | | | | |
| CPL -4 | | | Mampu menyajikan solusi atas permasalahan pada dunia industri dan masyarakat berdasarkan pengetahuan bidang rekayasa perangkat lunak | | | | | | | | | | | | | | | |
| CPL -5 | | | Mampu merancang, mengimplemntasikan dan mengevaluasi solusi inovatif terhadap pengembangan teknologi rekayasa perangkat lunak | | | | | | | | | | | | | | | |
| CPL -6 | | | Mampu berkolaborasi dengan berbagai pihak dari disiplin ilmu lain yang relevan secara efektif | | | | | | | | | | | | | | | |
| CPL -7 | | | Mampu berkomunikasi secara efektif untuk menyampaikan solusi dalam menjalankan profesinya | | | | | | | | | | | | | | | |
| CPL -8 | | | Memiliki komitmen terhadap prinsip dan nilai profesional sebagai landasan hidup dalamkonteks individu maupun organisasi | | | | | | | | | | | | | | | |
| CPL -9 | | | Memiliki jiwa pembelajar sepanjang hayat yang kreatif dan inovatif | | | | | | | | | | | | | | | |
|
| **Capaian Pembelajaran Mata Kuliah (CPMK)** | | | | | | | | |  | | | | | | | | | |
| CPMK -1 | | | Mahasiswa mengetahui Pengenalan terhadap konsep seperti class, object, inheritance, polymorphism, dan encapsulation. | | | | | | | | | | | | | | | |
| CPMK -2 | | | Mahasiswa mengetahui Metode dan prinsip dalam merancang sistem berorientasi objek yang efektif dan mudah dipelihara. | | | | | | | | | | | | | | | |
| CPMK -3 | | | Mahasiswa mengetahui Penggunaan bahasa pemrograman yang mendukung OOP seperti Java, C++, atau Python untuk mengimplementasikan konsep-konsep OOP. | | | | | | | | | | | | | | | |
| CPMK -4 | | | Mahasiswa memiliki pengetahuan tentang Pengenalan terhadap pola desain yang umum digunakan dalam OOP untuk menyelesaikan masalah desain yang umum. | | | | | | | | | | | | | | | |
| CPMK -5 | | | Mahasiswa mengetahui Praktik terbaik dalam menulis kode yang bersih, terstruktur, dan dapat digunakan kembali (reusable). | | | | | | | | | | | | | | | |
| CPMK -6 | | | Mahasiswa memiliki pengetahuan mengenai Studi kasus dan proyek-proyek untuk menerapkan konsep-konsep OOP dalam pengembangan perangkat lunak yang kompleks dan nyata. | | | | | | | | | | | | | | | |
| CPL->Sub-CPMK | | | | | | | | | | | | | | | | | | |
|  | |  | | | | | | | | | | | | | | | | |
|  | |  | | | | | | | | | | | | | | | | |
| Korelasi CPMK Terhadap Sub-CPMK | | | | | | | | | | | | | | | | | | |
|  | |  | Sub-CPMK 1 | | | | Sub-CPMK 2 | | Sub-CPMK 3 | | | Sub-CPMK 4 | | | Sub-CPMK 5 | | | Sub-CPMK 6 | | |
|  | | CPMK |  | | | |  | |  | | |  | | |  | | |  | | |
| **Deskripsi Singkat MK** | | Mata kuliah ini mempelajari konsep-konsep dasar pada object oriented programming (OOP), seperti class-object, encapsulation, inheritance, polymorphism, interface, abstract class. Secara spesifik konsep-konsep tersebut akan dipelajari dengan bantuan bahasa pemrograman Java. Selain konsep-konsep dasar OOP, pada mata kuliah ini juga akan dipelajari konsep exception handling, multithreading, object persistence, berbagai Collection, Java API, database connection, GUI (Graphical User Interface), | | | | | | | | | | | | | | | | | | |
| **Bahan Kajian / Materi Pembelajaran** | | 1. Konsep OOP 2. Classes and Object 3. Encapsulation 4. Class Relationship 5. Inheritance dan Polymorphism 6. Interface and Abstract Class 7. Static & Final method & Variables 8. Exception Handling 9. Object Persistence | | | | | | | | | | | | | | | | | | |
| **Pustaka** | | **Utama :** | | | |  | | | | | | | | | | | | | | |
|  | | | | | | | | | | | | | | | | | | |
| **Pendukung :** | | | |  | | | | | | | | | | | | | | |
|  | | | | | | | | | | | | | | | | | | |
| **Dosen Pengampu** | |  | | | | | | | | | | | | | | | | | | |
| **Matakuliah syarat** | | - | | | | | | | | | | | | | | | | | | |
| **Mg Ke-** | **Sub-CPMK**  **(Kemampuan akhir tiap tahapan belajar)** | | | | **Penilaian** | | | | | **Bantuk Pembelajaran,**  **Metode Pembelajaran,**  **Penugasan Mahasiswa,**  **[ Estimasi Waktu]** | | | | | | | **Materi Pembelajaran**  **[ Pustaka ]** | | | **Bobot Penilaian (%)** |
| **Indikator** | | | **Kriteria & Bentuk** | | **Luring (*offline*)** | | | **Daring (*online*)** | | | |
| **(1)** | **(2)** | | | | **(3)** | | | **(4)** | | **(5)** | | | **(6)** | | | | **(7)** | | | **(8)** |
| **1** | Mahasiswa mampu  memahami Konsep OOP digunakan untuk penyelesaian masalah | | | | Indikator : Ketepatan  memahami konsep PBO | | | Kriteria penilaian : Ketepatan dan Penguasaan materi Bentuk penilaian : Non-Test : Ketepatan dalam menjelaskan konsep. | | Ceramah Diskusi  Mahasiswa menjawab soal lisan berkaitan dengan perbedaan proseduran dengan oop. [150 Menit] | | | **Materi dan diskusi**  **Via Zoom meeting** | | | | 1. Sejarah OOP 2. Perbandingan Prosedural dengan OOP 3. OOP dalam Java | | |  |
| **2** | Mahasiswa mampu  menjelaskan Perbedaan antara Class dan object, implementasi class dan object, method dalam bahasa Java serta type data nya. | | | | Indikator : Ketepatan merancang class denga attributnya. | | | Kriteria penilaian : Ketepatan analisis, rancangan dan Penguasaan materi Bentuk penilaian : Tertulis :  Ketepatan dalam membuat class dan object yang di rancang. | | Ceramah dan demo  Tugas 1 : Membuat contoh class dari kasus real beikut atrribut nya (2 x 60 menit)  [2 x 150 Menit] | | | **Materi dan diskusi**  **Via Zoom meeting** | | | | 1. Class 2. Object 3. Attribut  4. ADT Java 5. Object Cycle  6. Constructor | | | **10%** |
| **3** | Mahasiswa mampu  menjelaskan Perbedaan antara Class dan object, implementasi class dan object, method dalam bahasa Java serta type data nya. | | | | Indikator : Ketepatan merancang class denga attributnya. | | | Kriteria penilaian : Ketepatan analisis, rancangan dan Penguasaan materi Bentuk penilaian : Tertulis :  Ketepatan dalam membuat class dan object yang di rancang. | | Ceramah dan demo  Tugas 1 : Membuat contoh class dari kasus real beikut atrribut nya (2 x 60 menit)  [2 x 150 Menit] | | | **Materi dan diskusi**  **Via Zoom meeting** | | | | 1. Class 2. Object 3. Attribut  4. ADT Java 5. Object Cycle  6. Constructor | | | **10%** |
| **4** | Mahasiswa mampu mejelaskan konsep Enkapsulasi dan implementasinya dalam PBO. | | | | Indikator : Ketepatan merancang class dan methodnya. | | | Kriteria penilaian : Ketepatan analisis, rancangan dan Penguasaan materi Bentuk penilaian : Tertulis :  Ketepatan dalam membuat class dan object dan method | | Ceramah dan demo  Mahasiswa menjelaskan rancangan implementasi sederhana class dan object dengan method nya. [150 Menit] | | | **Materi dan diskusi**  **Via Zoom meeting** | | | | 1. Object state 2. Instance variable vs local variables 3. Methods argument & return type 4. Pass by value 5. Getters and setters 6. Sample class diagram 7. representing encapsulation | | | **5%** |
| **5** | Mahasiwa mampu merancang  Diagram Class (Class  Relationship) | | | | Indikator : Ketepatan merancang diagram classpada object | | | Kriteria penilaian : Ketepatan analisis, rancangan dan Penguasaan materi Bentuk penilaian : Tertulis :Ketepatan rancangan diagramclass. | | Ceramah dan Problem Based  Learning  Tugas 2 : Membuat rangcangan diagram class sederha berdasarkan kasus nyata. (2 x 60 menit)  [2 x150 Menit] | | | **Materi dan diskusi**  **Via Zoom meeting** | | | | 1. association, aggregation, composition 2. simple class diagram & code in javarelated | | | **10%** |
| **6** | Mahasiwa mampu merancang  Diagram Class (Class  Relationship) | | | | Indikator : Ketepatan merancang diagram classpada object | | | Kriteria penilaian : Ketepatan analisis, rancangan dan Penguasaan materi Bentuk penilaian : Tertulis :Ketepatan rancangan diagramclass. | | Ceramah dan Problem Based  Learning  Tugas 2 : Membuat rangcangan diagram class sederha berdasarkan kasus nyata. (2 x 60 menit)  [2 x150 Menit] | | | **Materi dan diskusi**  **Via Zoom meeting** | | | | 1. association, aggregation, composition 2. simple class diagram & code in javarelated | | | **10%** |
| **7** | Mahasiswa mampu menjelaskan Konsep Inheritance dalam PBO | | | | Indikator : Ketepatan mensarikan sumber bacaan, sistematika dan kerapian menyusun power point | | | Kriteria : Ketepatan Penguasan materi Bentuk Penilaian : Non test : 1. Ketepatan materi PPT, kerapian PPT 2. Presentasi 3. Diskusi | | Ceramah dan Diskusi, Demo  Tugas 3 : Membuat rangkuman materi class, object, method dalam bentuk powerpoint (2 x 60 menit)  [150 Menit] | | | **Materi dan diskusi**  **Via Zoom meeting** | | | | 1. Designing for inheritance 2. Superclass and subclass relationship & access modifier 3. Method Overriding | | | **5%** |
| **8** | **Evaluasi Tengah Semester / Ujian Tengan Semester** | | | | | | | | | | | | | | | | | | | **30** |
| **9** | Mahasiswa mampu menjelaskan Konsep Polymorps dalam PBO | | | | Indikator : Ketepatan merancang aplikasi | | | Kriteria penilaian : Ketepatan analisis, rancangan dan Penguasaan materi Bentuk penilaian : Tertulis :Ketepatan rancangan aplikasi | | Ceramah dan Diskusi, Demo  Dapat membuat rancangan aplikasi sederhana dengan konsep polymorps  [150 Menit] | | | **Materi dan diskusi**  **Via Zoom meeting** | | | | 1. Polymorphism: in object declaration & assignment, in method argument & return type. 2. Method overloading3.  Constructor overloading (revisited) & superclass constructor 4. Calls to super(), this () 5. Sample classdiagram representinginheritance & polymorphism | | | **5%** |
| **10** | Mahasiswa mampu menjelaskan Konsep Interface dan Abstract Class dalam PBO | | | | Indikator : Ketepatan kode program | | | Kriteria penilaian : Ketepatan rancangan dan Penguasaan materi Bentuk penilaian : Tertulis : Ketepatan output kode program Non-test : Ketepatan menjelaskan program. | | Ceramah dan Diskusi  Tugas 4 :  Mengimplementasikan contoh class diagram kedalam kode program (2 x 60 menit)  [2 x 150 Menit] | | | **Materi dan diskusi**  **Via Zoom meeting** | | | | 1. Abstract classess and methods 2. Overloading and overriding (revisited) 3. Casting 4. Interface: making & using 5. Sample class diagram representing interface & abstract class | | |  |
| **11** | Mahasiswa mampu menjelaskan Konsep Interface dan Abstract Class dalam PBO | | | | Indikator : Ketepatan kode program | | | Kriteria penilaian : Ketepatan rancangan dan Penguasaan materi Bentuk penilaian : Tertulis : Ketepatan output kode program Non-test : Ketepatan menjelaskan program. | | Ceramah dan Diskusi  Tugas 4 :  Mengimplementasikan contoh class diagram kedalam kode program (2 x 60 menit)  [2 x 150 Menit] | | | **Materi dan diskusi**  **Via Zoom meeting** | | | | 1. Abstract classess and methods 2. Overloading and overriding (revisited) 3. Casting 4. Interface: making & using 5. Sample class diagram representing interface & abstract class | | |  |
| **12** | Mahasiswa mampu  menjelaskan danImplementasi Static & finalmethod &  variables | | | | Indikator : Ketepatan mejelaskan program | | | Kriteria penilaian : Ketepatan dan Penguasaan materiBentuk penilaian : Non-test :Makalah | | Ceramah, Contectual Learning  Tugas 5 : Mencari contoh kode program java yang menggukan sesuai materiyang di bahas danmenejelaskan dalam bentukmakalah. (2 x 60 menit) [150 Menit] | | | **Materi dan diskusi**  **Via Zoom meeting** | | | | 1. Static method 2. Static variable 3. Constants (static final 4. variable) 5. Final method & class 6. Wrapper classes & 7. Autoboxing 8.  Static imports 9. Example: Math class 10. Instance variable vs static variable | | | **5%** |
| **13** | Mahasiswa mampu menjelaskan Exception Hadling dan Class Exception  Handling | | | | Indikator : Ketepatan mejelaskan kodeprogram dengan diagram class | | | Kriteria penilaian : Ketepatan dan Penguasaan materi Bentuk penilaian : Non-test : Makalah | | Ceramah Small Grub Diskusi  Tugas 6 : Mencari contoh kode program java yang menggukan sesuai materi yang di bahas dan membuat diagram class nya dalam bentuk makalah. (2 x 60 menit)  [2 x 150 Menit] | | | **Materi dan diskusi**  **Via Zoom meeting** | | | | 1. Exception as an object 2. try-catch 3. throws 4. Trycatch- finally 5. Multiple exception 6. Polymorphism in exception 7. Making your own exception | | | **10%** |
| **14** | Mahasiswa mampu menjelaskan Exception Hadling dan Class Exception  Handling | | | | Indikator : Ketepatan mejelaskan kodeprogram dengan diagram class | | | Kriteria penilaian : Ketepatan dan Penguasaan materi Bentuk penilaian : Non-test : Makalah | | Ceramah Small Grub Diskusi  Tugas 6 : Mencari contoh kode program java yang menggukan sesuai materi yang di bahas dan membuat diagram class nya dalam bentuk makalah. (2 x 60 menit)  [2 x 150 Menit] | | | **Materi dan diskusi**  **Via Zoom meeting** | | | | 1. Exception as an object 2. try-catch 3. throws 4. Trycatch- finally 5. Multiple exception 6. Polymorphism in exception 7. Making your own exception | | | **10%** |
| **15** | Mahasiswa mampu menjelaskan konsep Multithreading pada java | | | | Indikator : Ketepatan analisais, sistematika dan kerapian menyusun Makalah | | | Kriteria : Ketepatan analisa dan Penguasan materi Bentuk Penilaian : Non test : 1. Demo rancangan dan aplikasi 2. Presentasi 3. Makalah | | Ceramah dan Diskusi  Tugas 7 : Menyusun makalah laporan akhir, rancangan dan implementasi PBO dengan java.  [150 Menit] | | | **Materi dan diskusi**  **Via Zoom meeting** | | | | 1. Launching new thread 2. Runnable interface 3. Thread' state 4. Runnable - running loop 5. Thread scheduler 6. Making and starting two thread | | | **5%** |
| **16** | **Evaluasi Akhir Semester / Ujian Akhir Semester** | | | | | | | | | | | | | | | | | | |  |
| **BOBOT PENILAIAN** | | | | | **Kehadiran dan Partisipasi dalam kelas : 10%**  **Hasil Praktikum dan Penugasan : 30%**  **Ujian Tengah Semester : 30%**  **Ujian Akhir Semester : 30%**  **Jumlah : 100%** | | | | | | | | | | | | | | | | |