



**GARIS GARIS BESAR PENGAJARAN
SATUAN ACARA PERKULIAHAN
KONTRAK KULIAH**

**REKAYASA PERANGKAT LUNAK
(PAC130)**

**PROGRAM STUDI ILMU KOMPUTER
JURUSAN MATEMATIK FAKULTAS MIPA
UNIVERSITAS DIPONEGORO**

KONTRAK PERKULIAHAN

Nama Mata Kuliah	: Rekayasa Perangkat Lunak (RPL)
Bobot	: 3 sks
Kode Mata Kuliah	: PAC130
Pengajar	: Aris Puji Widodo, MT. (APW)
Semester	: Ganjil
Hari Pertemuan	: Senin (12:30 – 15:30)
Tempat Pertemuan	: -

Deskripsi Mata Kuliah

RPL merupakan sarana yang digunakan untuk melakukan pembuatan dokumentasi pada setiap tahapan pengembangan perangkat lunak baik menggunakan metoda secara konvensional maupun berorientasi objek, dengan mengacu pada System Life Cycle Development (SDLC).

Tujuan Instruksioanal

1. Tujuan Instruksional Umum

Setelah mengikuti mata kuliah ini, mahasiswa diharapkan mempunyai pengetahuan tentang pembuatan dokumentasi pada setiap tahapan pengembangan perangkat lunak.

2. Tujuan Instruksional Khusus

- a. Jika diberikan pertanyaan tentang software, maka mahasiswa dapat memberikan penjelasan apa yang disebut dengan software minimal 90% benar.
- b. Jika diberikan pertanyaan tentang area proses pengembangan software, mahasiswa dapat menjelaskan pengertian dan urutan tahapan secara umum yang digunakan dalam pengembangan software minimal 90% benar.
- c. Jika diberikan pertanyaan tentang paradigma software engineering dari berbagai model, mahasiswa dapat menjelaskan berbagai model tersebut dengan karakteristiknya masing-masing minimal 90% benar.
- d. Jika diberikan permasalahan dan aturan bisnis yang akan dikembangkan perangkat lunak, maka mahasiswa akan dapat menyusun kebutuhan fungsional software dan termasuk melakukan interaksi komunikasi dengan user minimal 90% benar.
- e. Jika diberikan permasalahan dan aturan bisnis yang akan dikembangkan software, maka mahasiswa akan dapat menyusun dokumentasi yang dihasilkan pada tahapan analisis pada pendekatan secara konvensional minimal 90% benar.
- f. Jika diberikan permasalahan dan aturan bisnis yang akan dikembangkan software, maka mahasiswa akan dapat menyusun dokumentasi yang dihasilkan pada tahapan design pada pendekatan secara konvensional minimal 90% benar.

- g. Jika diberikan spesifikasi software, maka mahasiswa dapat menurunkan test case yang digunakan untuk melakukan pengujian software tersebut, termasuk membuat dokumentasi pengujian minimal 90% benar.
- h. Jika diberikan permasalahan dan aturan bisnis yang akan dikembangkan software, maka mahasiswa akan dapat menyusun dokumentasi yang dihasilkan pada pengembangan software menggunakan pendekatan berorientasi objek minimal 90% benar.

C. Metoda Pembelajaran

Untuk mencapai tujuan pendidikan dari mata kuliah ini, metode pembelajaran yang digunakan ialah: ceramah, pemberian contoh, tanya jawab, diskusi, dan penugasan. Penugasan dilakukan secara individu dan kelompok yang berorientasi pada proyek.

D. Buku Acuan

1. Bahrami, *Object Oriented System Development*, McGraw-Hill, 2000. [BAH2000]
2. Booch, G., J., Rumbaugh, Jacobson, *The Unified Modeling Language, User Guide*, Rational Software Corporation, Addison-Wesley, Massachusetts, USA, 1999. [BOO1999]
3. Date, C.J., *An Introduction to Database Systems*, Sixth Edition, New York: Addison Wesley Publ. Comp. Inc., 1995. [DAT1995]
4. Davis, M., Alan, *Software Requirement Object, Functions, and States*, Fifth Edition, Prentice Hall, 1993. [DAV1993]
5. Humphrey, Watts, S., et.al., *Managing The Software Process*, Addison Wesley, 1989. [HUM1989]
6. Ken Lunn , *Information Modeling and Development*, Prentice Hall, 2001. [KEN2001]
7. Pressman, Roger, S., *Software Engineering A Practitioner's Approach*, Fifth Edition, Mac Graw-Hill, 2000. [PRE2000]
8. Samorville, Ian, *Software Engineering*, Third Edition, Prentice Hall, 1996. [SAM1996]

E. Kriteria Penilaian

Nilai akhir dihitung dari komponen-komponen sebagai berikut:

No.	Komponen Penilaian	Bobot (%)
1.	Tugas Kecil	10
2.	Ujian Tengah Semester	30
3.	Tugas Besar (Proyek)	25
4.	Ujian Akhir Semester	35
Jumlah		100

F. Rencana Perkuliahan

Minggu / Pertemuan ke	Pokok Bahasan / Sub Pokok Bahasan	Pustaka	Pengampu
1	1. Konsep Dasar Software a. Pengertian Software b. Karakteristik Software c. Evolusi pengembangan Software d. Software Engineering e. Keuntungan dan Resiko pendekatan Software Engineering	a. [PRE2000] b. [SAM1996]	APW
2	2. Proses Software a. Pandangan Umum Pengembangan Software. b. Umbrella Activities. c. Maturity Process	a. [HUM1989] b. [PRE2000] c. [SAM1996]	APW
3, 4	3. Paradigma Software Engineering a. Waterfall / Linier Life Cycle Model. b. Spiral Model. c. Rapid Application Development Model. d. Incremental Model. e. Prototyping Model. f. V Model.	a. [PRE2000] b. [SAM1996]	APW
5, 6	4. Software Requirement Specification (SRS) a. Definisi SRS. b. Hal yang Harus Ada pada Dokumen SRS. c. Hal yang Tidak Boleh Ada pada Dokumen SRS. d. Atribut-atribut Dokumen SRS. e. Standard Dokumen SRS.	a. [DAV1993] b. [PRE2000] c. [SAM1996]	APW
7, 8	5. Tahapan Analisis secara Konvensional a. Data Object Description. b. Entity Relationship Diagram. c. Data Flow Diagram. d. Process Specification. e. State Transition Diagram. f. Control Specification. g. Kamus Data.	a. [DAT1995] b. [DAV1993] c. [PRE2000] d. [SAM1996]	APW
9	Ujian Tengah Semester (UTS)		APW
10, 11	6. Tahapan Design secara Konvensional (SQL) a. Perancangan Data. b. Perancangan Arsitektural. c. Perancangan Antarmuka. d. Perancangan Prosedur Detil.	a. [DAV1993] b. [PRE2000] c. [SAM1996]	APW
12,13, 14	7. Tahapan Pengujian Software a. Metoda Black Box Testing. b. Equivalance Class Testing. c. Endurances Testing. d. White Box Testing.	a. [DAV1993] b. [PRE2000] c. [SAM1996]	APW

Minggu / Pertemuan ke	Pokok Bahasan / Sub Pokok Bahasan	Pustaka	Pengampu
	e. Loop Testing. f. Conditional Testing. g. Basic Path Testing		
15, 16	8. Pengembangan Berorientasi Objek a. Konsep Berorientasi Objek. b. Metoda-metoda Berorientasi Objek. c. Use Case Diagram d. Sequences Diagram e. Class Diagram	a. [BAH2000] b. [BOO1999] c. [DAV1993] d. [KEN2001] e. [PRE2000] f. [SAM1996]	APW

GARIS-GARIS BESAR PROGRAM PENGAJARAN (GBPP)

Judul Mata Kuliah : Rekayasa Perangkat Lunak (RPL)
Kode Mata Kuliah : PAC130
Bobot sks : 3

A. Deskripsi Mata Kuliah

RPL merupakan sarana yang digunakan untuk melakukan pembuatan dokumentasi pada setiap tahapan pengembangan perangkat lunak baik menggunakan metoda secara konvensional maupun berorientasi objek, dengan mengacu pada System Life Cycle Development (SDLC).

B. Tujuan Instruksional Umum

Setelah mengikuti mata kuliah ini, mahasiswa diharapkan mempunyai pengetahuan tentang pembuatan dokumentasi pada setiap tahapan pengembangan perangkat lunak.

C. Satuan Acara Pengajaran

No.	Tujuan Instruksional Khusus	Pokok Bahasan	Sub Pokok Bahasan	Estimasi Waktu
1.	Jika diberikan pertanyaan tentang software, maka mahasiswa dapat memberikan penjelasan apa yang disebut dengan software minimal 90% benar.	Konsep Dasar Software	1. Pengertian Software 2. Karakteristik Software 3. Evolusi pengembangan Software 4. Software Engineering 5. Keuntungan dan Resiko pendekatan Software Engineering	3 x 50 menit
2.	Jika diberikan pertanyaan tentang area proses pengembangan software, mahasiswa dapat menjelaskan pengertian dan urutan tahapan secara umum yang digunakan dalam pengembangan software minimal 90% benar.	Proses Software	1. Pandangan Umum Pengembangan Software. 2. Umbrella Activities. 3. Maturity Process.	3 x 50 menit
3.	Jika diberikan pertanyaan tentang paradigma software engineering dari berbagai model, mahasiswa dapat menjelaskan berbagai model tersebut dengan karakteristiknya masing-masing minimal 90% benar.	Paradigma Software Engineering	1. Waterfall / Linier Life Cycle Model. 2. Spiral Model. 3. Rapid Application Development Model. 4. Incremental Model. 5. Prototyping Model. 6. V Model.	2 x 3 x 50 menit
4.	Jika diberikan permasalahan dan aturan bisnis yang akan dikembangkan perangkat	Software Requirement Specification	1. Definisi SRS 2. Hal yang Harus Ada	2 x 3 x 50 menit

No.	Tujuan Instruksional Khusus	Pokok Bahasan	Sub Pokok Bahasan	Estimasi Waktu
	lunak, maka mahasiswa akan dapat menyusun kebutuhan fungsional software dan termasuk melakukan interaksi komunikasi dengan user minimal 90% benar.	(SRS)	<p>pada Dokumen SRS.</p> <p>3. Hal yang Tidak Boleh Ada pada Dokumen SRS.</p> <p>4. Atribut-atribut Dokumen SRS.</p> <p>5. Standard Dokumen SRS.</p>	
5.	Jika diberikan permasalahan dan aturan bisnis yang akan dikembangkan software, maka mahasiswa akan dapat menyusun dokumentasi yang dihasilkan pada tahapan analisis pada pendekatan secara konvensional minimal 90% benar.	Tahapan Analisis secara Konvensional	<p>1. Data Object Description.</p> <p>2. Entity Relationship Diagram.</p> <p>3. Data Flow Diagram.</p> <p>4. Process Specification.</p> <p>5. State Transition Diagram.</p> <p>6. Control Specification.</p> <p>7. Kamus Data.</p>	2 x 3 x 50 menit
6.	Evaluasi proses belajar	Ujian Tengah Semester (UTS)		2 x 50 menit
7.	Jika diberikan permasalahan dan aturan bisnis yang akan dikembangkan software, maka mahasiswa akan dapat menyusun dokumentasi yang dihasilkan pada tahapan design pada pendekatan secara konvensional minimal 90% benar.	Tahapan Design secara Konvensional	<p>1. Perancangan Data.</p> <p>2. Perancangan Arsitektural.</p> <p>3. Perancangan Antarmuka.</p> <p>4. Perancangan Prosedur Detil.</p>	2 x 3 x 50 menit
8.	Jika diberikan spesifikasi software, maka mahasiswa dapat menurunkan test case yang digunakan untuk melakukan pengujian software tersebut, termasuk membuat dokumentasi pengujian minimal 90% benar.	Tahapan Pengujian Software	<p>1. Metoda Black Box Testing</p> <p>2. Equivalence Class Testing.</p> <p>3. Endurances Testing.</p> <p>4. White Box Testing.</p> <p>5. Loop Testing.</p> <p>6. Conditional Testing.</p> <p>7. Basic Path Testing</p>	3 x 3 x 50 menit
9.	Jika diberikan permasalahan dan aturan bisnis yang akan dikembangkan software, maka mahasiswa akan dapat menyusun dokumentasi yang dihasilkan pada pengembangan software menggunakan pendekatan berorientasi objek minimal 90% benar.	Pengembangan Berorientasi Objek	<p>1. Konsep Berorientasi Objek.</p> <p>2. Metoda-metoda Berorientasi Objek.</p> <p>3. Use Case Diagram</p> <p>4. Sequences Diagram</p> <p>5. Class Diagram.</p>	2 x 3 x 50 menit
			Jumlah	1.750 menit

SATUAN ACARA PENGAJARAN (SAP)

Mata Kuliah : Rekayasa Perangkat Lunak
 Kode Mata Kuliah : PAC130
 Bobot sks : 3
 Waktu Pertemuan : 3 x 50 menit
 Pertemuan ke : I

- A. Tujuan Instruksional :
1. Umum : Setelah mengikuti mata kuliah ini, mahasiswa diharapkan mempunyai pengetahuan tentang karakteristik software dan dapat membedakan antara software dengan produk-produk manufaktur.
 2. Khusus : Jika diberikan pertanyaan tentang software, maka mahasiswa dapat memberikan penjelasan apa yang disebut dengan software minimal 90% benar.
- B. Pokok Bahasan : Konsep Dasar Software
- C. Sub Pokok Bahasan : 1. Pengertian Software
 2. Karakteristik Software
 3. Evolusi pengembangan Software
 4. Software Engineering
 5. Keuntungan dan Resiko pendekatan Software Engineering

D. Kegiatan Belajar Mengajar dan Media Alat Pengajaran

Tahap	Kegiatan Pengajar	Kegiatan Mahasiswa	Media dan Alat Pengajar
1	2	3	4
Pendahuluan	1. Menjelaskan cakupan materi dalam pertemuan ke I. 2. Menjelaskan manfaat memahami konsep dasar software. 3. Menjelaskan kompetensi dalam TIU dan TIK untuk pertemuan ke I.	<ul style="list-style-type: none"> • Memperhatikan. • Memperhatikan. • Memperhatikan. 	<ul style="list-style-type: none"> • OHP • OHP • OHP
Penyajian	4. Karakteristik Software: a. Menjelaskan software tidak sama dengan produk manufaktur. b. Menjelaskan software dibangun tidak menggunakan	<ul style="list-style-type: none"> • Memperhatikan, dan mencatat. • Memperhatikan, dan 	<ul style="list-style-type: none"> • OHP, papan tulis. • OHP, papan tulis.

Tahap	Kegiatan Pengajar	Kegiatan Mahasiswa	Pengajar
1	2	3	4
	<p>komponen-komponen yang sudah ada.</p> <p>c. Menjelaskan software tidak mengenal masa kadaluwarsa</p> <p>5. Evolusi Pengembangan Software:</p> <p>a. Menjelaskan trend pengembangan software mulai tahun 1940-an sampai masa sekarang.</p> <p>b. Menjelaskan pengertian dan contoh software batch.</p> <p>c. Menjelaskan pengertian dan contoh software bisnis.</p> <p>6. Software Engineering:</p> <p>a. Menjelaskan pengertian Software Engineering.</p> <p>b. Menjelaskan tools yang digunakan untuk Software Engineering.</p> <p>c. Menjelaskan alasan perlu menggunakan Software Engineering.</p> <p>d. Menjelaskan bagian yang terlibat pada Software Engineering.</p> <p>7. Menjelaskan keuntungan dan resiko pendekatan Software Engineering.</p>	<p>mencatat.</p> <ul style="list-style-type: none"> • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. 	<ul style="list-style-type: none"> • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis.
Penutup	<p>8. Menutup pertemuan:</p> <p>a. Memberikan pertanyaan-pertanyaan kepada mahasiswa serta melakukan diskusi dalam rangka memberikan pemahaman secara lebih baik tentang materi yang diajarkan.</p> <p>b. Memberikan beberapa penguatan bagi mahasiswa.</p> <p>c. Menjelaskan ulang tentang hal-hal yang masih belum dimengerti oleh mahasiswa.</p> <p>d. Merangkum secara ringkas dan membahas keterkaitan dengan materi kuliah berikutnya.</p>	<ul style="list-style-type: none"> • Melaksanakan, mengemukakan pendapat. 	<ul style="list-style-type: none"> • OHP, papan tulis.

E. Evaluasi

1. Membenarkan tugas tulis kepada mahasiswa untuk menyajikan contoh-contoh software generasi pertama (1940-1950) yang ada sekarang, untuk mengetahui tingkat penyerapan materi oleh mahasiswa.
2. Penilaian dilakukan minggu berikutnya setelah mahasiswa menyerahkan tugas tulis yang diberikan.

F. Referensi

1. Pressman, Roger, S., *Software Engineering A Practitioner's Approach*, Fifth Edition, Mac Graw-Hill, 2000.
2. Samorville, Ian, *Software Engineering*, Third Edition, Prentice Hall, 1996.

SATUAN ACARA PENGAJARAN (SAP)

Mata Kuliah : Rekayasa Perangkat Lunak
 Kode Mata Kuliah : PAC130
 Bobot sks : 3
 Waktu Pertemuan : 3 x 50 menit
 Pertemuan ke : II

- A. Tujuan Instruksional :**
1. Umum : Setelah mengikuti mata kuliah ini memberikan pengetahuan tentang area proses pengembangan software, meliputi tahapan-tahapan yang proses pengembangan dari berbagai standard yang ada , dengan ditekankan pada pandangan secara umum.
 2. Khusus : Jika diberikan pertanyaan tentang area proses pengembangan software, mahasiswa dapat menjelaskan pengertian dan urutan tahapan secara umum yang digunakan dalam pengembangan software.
- B. Pokok Bahasan : Proses Software**
- C. Sub Pokok Bahasan :**
1. Pandangan Umum Pengembangan Software.
 2. Umbrella Activities.
 3. Maturity Process.

D. Kegiatan Belajar Mengajar dan Media Alat Pengajaran

Tahap	Kegiatan Mengajar	Kegiatan Mahasiswa	Media dan Alat Pengajar
1	2	3	4
Pendahuluan	1. Menjelaskan cakupan materi dalam pertemuan ke II. 2. Menjelaskan manfaat memahami Proses Software. 3. Menjelaskan kompetensi dalam TIU dan TIK untuk pertemuan ke II	<ul style="list-style-type: none"> • Memperhatikan. • Memperhatikan. • Memperhatikan. 	<ul style="list-style-type: none"> • OHP • OHP • OHP
Penyajian	4. Pandangan Umum Pengembangan Software: a. Menjelaskan tentang definition phase. b. Menjelaskan tentang development phase.	<ul style="list-style-type: none"> • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. 	<ul style="list-style-type: none"> • OHP, papan tulis. • OHP, papan tulis.

Tahap	Kegiatan Mengajar	Kegiatan Mahasiswa	Pengajar
1	2	3	4
	<p>c. Menjelaskan tentang maintenance phase.</p> <p>5. Umbrella Activities:</p> <p>a. Menjelaskan mekanisme dan tahapan pada Umbrella Activities.</p> <p>b. Menjelaskan persamaan Umbrella Activities dengan pandangan secara umum.</p> <p>6. Maturity Process::</p> <p>a. Menjelaskan mekanisme dan tahapan pada Maturity Process.</p> <p>b. Menjelaskan persamaan Maturity Process dengan pandangan secara umum.</p>	<ul style="list-style-type: none"> • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. 	<ul style="list-style-type: none"> • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis.
Penutup	<p>7. Menutup pertemuan:</p> <p>a. Memberikan pertanyaan-pertanyaan kepada mahasiswa serta melakukan diskusi dalam rangka memberikan pemahaman secara lebih baik tentang materi yang diajarkan.</p> <p>b. Memberikan beberapa penguatan bagi mahasiswa.</p> <p>c. Menjelaskan ulang tentang hal-hal yang masih belum dimengerti oleh mahasiswa.</p> <p>d. Merangkum secara ringkas dan membahas keterkaitan dengan materi kuliah mendatang.</p>	<ul style="list-style-type: none"> • Melaksanakan, mengemukakan pendapat. 	<ul style="list-style-type: none"> • OHP, papan tulis.

E. Evaluasi

1. Memberikan tugas tulis kepada mahasiswa untuk melakukan identifikasi tentang mekanisme dan alur tahapan-tahapan yang digunakan pada pengembangan perangkat lunak secara umum, untuk didiskusikan di dalam kelas.
2. Penilaian dilakukan minggu berikutnya setelah mahasiswa menyerahkan tugas tulis yang diberikan.

F. Referensi

1. Humphrey, Watts, S., et.al., *Managing The Software Process*, Addison Wesley, 1989.
2. Pressman, Roger, S., *Software Engineering A Practitioner's Approach*, Fifth Edition, Mac Graw-Hill, 2000.

SATUAN ACARA PENGAJARAN (SAP)

Mata Kuliah : Rekayasa Perangkat Lunak
 Kode Mata Kuliah : PAC130
 Bobot sks : 3
 Waktu Pertemuan : 2 x 3 x 50 menit
 Pertemuan ke : III + IV

- A. Tujuan Instruksional :
1. Umum : Setelah mengikuti mata kuliah ini memberikan pengetahuan tentang area proses tahapan / cara yang digunakan untuk mengembangkan software dari beberapa model dan karakteristik dari masing-masing model tersebut.
 2. Khusus : Jika diberikan pertanyaan tentang paradigma software engineering dari berbagai model, mahasiswa dapat menjelaskan berbagai model tersebut dengan karakteristiknya masing-masing.
- B. Pokok Bahasan : Paradigma Software Engineering
- C. Sub Pokok Bahasan : 1. Waterfall / Linier Life Cycle Model.
2. Spiral Model.
3. Rapid Application Development Model.
4. Incremental Model.
5. Prototyping Model.
6. V Model.

D. Kegiatan Belajar Mengajar dan Media Alat Pengajaran

Tahap	Kegiatan Mengajar	Kegiatan Mahasiswa	Media dan Alat Pengajar
1	2	3	4
Pendahuluan	1. Menjelaskan cakupan materi dalam pertemuan yang sedang berlangsung. 2. Menjelaskan manfaat memahami Paradigma Software Engineering. 3. Menjelaskan kompetensi dalam TIU dan TIK untuk pertemuan yang sedang berlangsung.	<ul style="list-style-type: none"> • Memperhatikan. • Memperhatikan. • Memperhatikan. 	<ul style="list-style-type: none"> • OHP • OHP • OHP

Isianap	Kegiatan Mengajar	Kegiatan Mahasiswa	Pengajar
1	2	3	4
Penyajian	<p>4. Model-model Paradigma Software Engineering:</p> <p>a. Menjelaskan Waterfall / Linier Life Cycle Model, dengan karakteristiknya.</p> <p>b. Menjelaskan Spiral Model, dengan karakteristiknya.</p> <p>c. Menjelaskan Rapid Application Development Model, dengan karakteristiknya.</p> <p>d. Menjelaskan Incremental Model, dengan karakteristiknya.</p> <p>e. Menjelaskan Prototyping Model, dengan karakteristiknya.</p> <p>f. Menjelaskan V Model.</p> <p>5. Perbandingan model:</p> <p>a. Menjelaskan kelebihan dan kekurangan pada setiap model.</p> <p>b. Menjelaskan pertimbangan yang digunakan dalam pemilihan model.</p>	<ul style="list-style-type: none"> • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. 	<ul style="list-style-type: none"> • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis.
Penutup	<p>6. Menutup pertemuan:</p> <p>a. Memberikan pertanyaan-pertanyaan kepada mahasiswa serta melakukan diskusi dalam rangka memberikan pemahaman secara lebih baik tentang materi yang diajarkan.</p> <p>b. Memberikan beberapa penguatan bagi mahasiswa.</p> <p>c. Menjelaskan ulang tentang hal-hal yang masih belum dimengerti oleh mahasiswa.</p> <p>d. Merangkum secara ringkas dan membahas keterkaitan dengan materi kuliah mendatang.</p>	<ul style="list-style-type: none"> • Melaksanakan, mengemukakan pendapat. 	<ul style="list-style-type: none"> • OHP, papan tulis.

E. Evaluasi

1. Memberikan tugas tulis kepada mahasiswa untuk menyajikan tulisan tentang model Paradigma Software Engineering yang populer digunakan pada pengembangan software, disertai deskripsi perkembangannya.
2. Penilaian dilakukan minggu berikutnya setelah mahasiswa menyerahkan tugas tulis yang diberikan.

F. Referensi

1. Pressman, Roger, S., *Software Engineering A Practitioner's Approach*, Fifth Edition, Mac Graw-Hill, 2000.
2. Samorville, Ian, *Software Engineering*, Third Edition, Prentice Hall, 1996.

UNIT KEGIATAN BELAJAR (UKB)

Mata Kuliah : Rekayasa Perangkat Lunak
 Kode Mata Kuliah : PAC130
 Bobot sks : 3
 Waktu Pertemuan : 2 x 3 x 50 menit
 Pertemuan ke : V + VI

- A. Tujuan Instruksional :
1. Umum : Setelah mengikuti mata kuliah ini memberikan pengetahuan tentang area penyusunan kebutuhan fungsional yang dibutuhkan dalam pengembangan software dan model interaksi komunikasi antara developer dan user.
 2. Khusus : Jika diberikan permasalahan dan aturan bisnis yang akan dikembangkan software, maka mahasiswa akan dapat menyusun kebutuhan fungsional software dan termasuk melakukan interaksi komunikasi dengan user.
- B. Pokok Bahasan : Software Requirement Specification (SRS)
- C. Sub Pokok Bahasan :
1. Definisi SRS.
 2. Hal yang Harus Ada pada Dokumen SRS.
 3. Hal yang Tidak Boleh Ada pada Dokumen SRS.
 4. Atribut-atribut Dokumen SRS.
 5. Standard Dokumen SRS.

D. Kegiatan Belajar Mengajar dan Media Alat Pengajaran

Tahap	Kegiatan Mengajar	Kegiatan Mahasiswa	Media dan Alat Pengajar
1	2	3	4
Pendahuluan	1. Menjelaskan cakupan materi dalam pertemuan yang sedang berlangsung. 2. Menjelaskan manfaat memahami SRS. 3. Menjelaskan kompetensi dalam TIU dan TIK untuk pertemuan yang sedang berlangsung.	<ul style="list-style-type: none"> • Memperhatikan. • Memperhatikan. • Memperhatikan. 	<ul style="list-style-type: none"> • OHP • OHP • OHP
Penyajian	4. Definisi SRS:		

1	2	3	4
	<p>a. Menjelaskan pengertian SRS.</p> <p>b. Menjelaskan arti penting dan manfaatnya SRS.</p> <p>c. Menanyakan kepada mahasiswa tentang SRS yang dapat dibangkitkan pada lingkungan pengembangan software (untuk berbagai macam model aplikasi).</p> <p>5. Hal yang Harus Ada pada Dokumen SRS:</p> <p>a. Menjelaskan tentang hal-hal yang harus dipenuhi dalam penyusunan dokumen SRS.</p> <p>b. Menjelaskan representasi dokumen SRS sebagai alat komunikasi antara developer dan user.</p> <p>6. Hal yang Tidak Boleh Ada pada Dokumen SRS:</p> <p>a. Menjelaskan tentang hal-hal yang tidak dimuat dalam dokumen SRS.</p> <p>b. Menjelaskan atribut-atribut yang digunakan untuk mengukur kualitas dokumen SRS.</p> <p>7. Standard Dokumen SRS:</p> <p>a. Menjelaskan dokumen SRS dari IEEE.</p> <p>b. Menjelaskan dokumen SRS dari DOD.</p> <p>c. Menjelaskan dokumen SRS dari ANSI.</p>	<ul style="list-style-type: none"> • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. 	<ul style="list-style-type: none"> • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis.
Penutup	<p>8. Menutup pertemuan:</p> <p>a. Memberikan pertanyaan-pertanyaan kepada mahasiswa serta melakukan diskusi dalam rangka memberikan pemahaman secara lebih baik tentang materi yang diajarkan.</p> <p>b. Memberikan beberapa penguatan bagi mahasiswa.</p> <p>c. Menjelaskan ulang tentang hal-hal yang masih belum dimengerti oleh mahasiswa.</p> <p>d. Merangkum secara ringkas dan membahas keterkaitan dengan materi kuliah mendatang.</p>	<ul style="list-style-type: none"> • Melaksanakan, dan mengemukakan pendapat. 	<ul style="list-style-type: none"> • OHP, papan tulis.

E. Evaluasi

1. Memberikan tugas kepada mahasiswa untuk menyusun dokumen SRS yang diambil dari permasalahan bisnis pada dunia nyata (*real world*).
2. Penilaian dilakukan minggu berikutnya setelah mahasiswa menyerahkan tugas tulis yang diberikan.

F. Referensi

1. Davis, M., Alan, ***Software Requirement Object, Functions, and States***, Fifth Edition, Prentice Hall, 1993.
2. Pressman, Roger, S., ***Software Engineering A Practitioner's Approach***, Fifth Edition, Mac Graw-Hill, 2000.
3. Samorville, Ian, ***Software Engineering***, Third Edition, Prentice Hall, 1996.

Mata Kuliah : Rekayasa Perangkat Lunak
Kode Mata Kuliah : PAC130
Bobot sks : 3
Waktu Pertemuan : 2 x 3 x 50 menit
Pertemuan ke : VII + VIII

- A. Tujuan Instruksional** :
1. Umum : Setelah mengikuti mata kuliah ini memberikan pengetahuan tentang area penyusunan dokumentasi yang dihasilkan pada pengembangan software dengan menggunakan pendekatan secara konvensional.
 2. Khusus : Jika diberikan permasalahan dan aturan bisnis yang akan dikembangkan software, maka mahasiswa akan dapat menyusun dokumentasi yang dihasilkan pada tahapan analisis pada pendekatan secara konvensional.
- B. Pokok Bahasan** : Tahapan Analisis secara Konvensional
- C. Sub Pokok Bahasan** :
1. Data Object Description.
 2. Entity Relationship Diagram.
 3. Data Flow Diagram.
 4. Process Specification.
 5. State Transition Diagram.
 6. Control Specification.
 7. Kamus Data.

D. Kegiatan Belajar Mengajar dan Media Alat Pengajaran

Tahap	Kegiatan Mengajar	Kegiatan Mahasiswa	Media dan Alat Pengajar
1	2	3	4
Pendahuluan	1. Menjelaskan cakupan materi dalam pertemuan yang sedang berlangsung. 2. Menjelaskan manfaat tahapan analisis. 3. Menjelaskan kompetensi dalam TIU dan TIK untuk pertemuan yang sedang berlangsung.	<ul style="list-style-type: none"> • Memperhatikan. • Memperhatikan. • Memperhatikan. 	<ul style="list-style-type: none"> • OHP • OHP • OHP

Isian	Kegiatan Mengajar	Kegiatan Mahasiswa	Kegiatan Pengajar
1	2	3	4
Penyajian	<p>4. Analisis secara Konvensional:</p> <p>a. Menjelaskan konsep pemodelan data.</p> <p>b. Menjelaskan konsep data object description.</p> <p>5. Entity Relationship Diagram:</p> <p>a. Menjelaskan entitas dan variannya.</p> <p>b. Menjelaskan atribut dan variannya.</p> <p>c. Menjelaskan relasi dan variannya.</p> <p>d. Menanyakan kepada mahasiswa bagaimana menyusun entitas, atribut, dan relasi untuk membangun Entity Relationship Diagram.</p> <p>6. Data Flow Diagram:</p> <p>a. Menjelaskan konsep pemodelan fungsi.</p> <p>b. Menjelaskan notasi eksternal entitas dan variannya.</p> <p>c. Menjelaskan notasi proses dan variannya.</p> <p>d. Menjelaskan notasi data store dan variannya.</p> <p>e. Menjelaskan process specification.</p> <p>7. State Transition Diagram:</p> <p>a. Menjelaskan konsep pemodelan status.</p> <p>b. Menjelaskan control specification</p> <p>c. Menjelaskan kamus data</p>	<ul style="list-style-type: none"> • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. 	<ul style="list-style-type: none"> • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis.
Penutup	<p>8. Menutup pertemuan:</p> <p>a. Memberikan pertanyaan-pertanyaan kepada mahasiswa</p>	<ul style="list-style-type: none"> • Melaksanakan, dan mengemukakan 	<ul style="list-style-type: none"> • OHP, papan tulis.

Tahap	Kegiatan Mengajar	Kegiatan Mahasiswa	Media dan Alat Pengajar
1	2	3	4
	<p>serta melakukan diskusi dalam rangka memberikan pemahaman secara lebih baik tentang materi yang diajarkan.</p> <p>b. Memberikan beberapa penguatan bagi mahasiswa.</p> <p>c. Menjelaskan ulang tentang hal-hal yang masih belum dimengerti oleh mahasiswa.</p> <p>d. Merangkum secara ringkas dan membahas keterkaitan dengan materi kuliah mendatang.</p>	pendapat.	

E. Evaluasi

1. Memberikan tugas kepada mahasiswa untuk menyusun dokumen pada tahap analisis yang diambil dari permasalahan bisnis pada dunia nyata (*real world*).
2. Penilaian dilakukan minggu berikutnya setelah mahasiswa menyerahkan tugas tulis yang diberikan.

F. Referensi

1. Date, C.J., *An Introduction to Database Systems*, Sixth Edition, New York: Addison Wesley Publ. Comp. Inc., 1995.
2. Davis, M., Alan, *Software Requirement Object, Functions, and States*, Fifth Edition, Prentice Hall, 1993.
3. Pressman, Roger, S., *Software Engineering A Practitioner's Approach*, Fifth Edition, Mac Graw-Hill, 2000.
4. Samorville, Ian, *Software Engineering*, Third Edition, Prentice Hall, 1996.

UNIT KEGIATAN BELAJAR (UKB)

Mata Kuliah : Rekayasa Perangkat Lunak
 Kode Mata Kuliah : PAC130
 Bobot sks : 3
 Waktu Pertemuan : 2 x 3 x 50 menit
 Pertemuan ke : X + XI

- A. Tujuan Instruksional :
1. Umum : Setelah mengikuti mata kuliah ini memberikan pengetahuan tentang area penyusunan dokumentasi yang dihasilkan pada pengembangan software dengan menggunakan pendekatan secara konvensional.
 2. Khusus : Jika diberikan permasalahan dan aturan bisnis yang akan dikembangkan software, maka mahasiswa akan dapat menyusun dokumentasi yang dihasilkan pada tahapan design pada pendekatan secara konvensional.
- B. Pokok Bahasan : Tahapan Design secara Konvensional
- C. Sub Pokok Bahasan : 1. Perancangan Data.
 2. Perancangan Arsitektural.
 3. Perancangan Antarmuka.
 4. Perancangan Prosedur Detil.

D. Kegiatan Belajar Mengajar dan Media Alat Pengajaran

Tahap	Kegiatan Mengajar	Kegiatan Mahasiswa	Media dan Alat Pengajar
1	2	3	4
Pendahuluan	1. Menjelaskan cakupan materi dalam pertemuan yang sedang berlangsung. 2. Menjelaskan manfaat tahapan design. 3. Menjelaskan kompetensi dalam TIU dan TIK untuk pertemuan yang sedang berlangsung.	<ul style="list-style-type: none"> • Memperhatikan. • Memperhatikan. • Memperhatikan. 	<ul style="list-style-type: none"> • OHP • OHP • OHP
Penyajian	4. Perancangan Data: a. Menjelaskan proses mapping dari Entity Relationship Diagram ke dalam bentuk fisik.	<ul style="list-style-type: none"> • Memperhatikan, dan mencatat. 	<ul style="list-style-type: none"> • OHP, papan tulis.

1. Tujuan	Kegiatan Mengajar	Kegiatan Mahasiswa	Kegiatan Pengajar
1	2	3	4
	<p>b. Menjelaskan spesifikasi data yang digunakan sebagai acuan untuk implementasi ke Data Management System (DBMS).</p> <p>5. Perancangan Arsitektural:</p> <p>a. Menjelaskan keterkaitan antar modul/proses yang dihasilkan pada Data Flow Diagram.</p> <p>b. Menjelaskan keterkaitan antar modul/proses yang direpresentasikan dengan struktur hirarki.</p> <p>6. Perancangan Antarmuka:</p> <p>a. Menjelaskan antarmuka ke modul.</p> <p>b. Menjelaskan antarmuka ke file.</p> <p>c. Menjelaskan antarmuka ke pengguna.</p> <p>7. Perancangan Prosedur Detil:</p> <p>a. Menjelaskan detil prosedur yang dihasilkan dengan cara menyusun algoritma dari process specification.</p> <p>b. Menjelaskan representasi detil prosedur menggunakan notasi algoritmik.</p>	<ul style="list-style-type: none"> • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. 	<ul style="list-style-type: none"> • OHP, papan tulis • OHP, papan tulis • OHP, papan tulis • OHP, papan tulis • OHP, papan tulis • OHP, papan tulis • OHP, papan tulis • OHP, papan tulis
Penutup	<p>8. Menutup pertemuan:</p> <p>a. Memberikan pertanyaan-pertanyaan kepada mahasiswa serta melakukan diskusi dalam rangka memberikan pemahaman secara lebih baik tentang materi yang diajarkan.</p> <p>b. Memberikan beberapa penguatan bagi mahasiswa.</p> <p>c. Menjelaskan ulang tentang hal-hal yang masih belum dimengerti oleh mahasiswa.</p> <p>d. Merangkum secara ringkas dan membahas keterkaitan dengan materi kuliah mendatang.</p>	<ul style="list-style-type: none"> • Melaksanakan, dan mengemukakan pendapat. 	<ul style="list-style-type: none"> • OHP, papan tulis.

E. Evaluasi

1. Memberikan tugas kepada mahasiswa untuk menyusun dokumen pada tahap design yang diambil dari permasalahan bisnis pada dunia nyata (*real world*).

F. Referensi

1. Davis, M., Alan, ***Software Requirement Object, Functions, and States***, Fifth Edition, Prentice Hall, 1993.
2. Pressman, Roger, S., ***Software Engineering A Practitioner's Approach***, Fifth Edition, Mac Graw-Hill, 2000.
3. Samorville, Ian, ***Software Engineering***, Third Edition, Prentice Hall, 1996.

Mata Kuliah : Rekayasa Perangkat Lunak
Kode Mata Kuliah : PAC130
Bobot sks : 3
Waktu Pertemuan : 3 x 3 x 50 menit
Pertemuan ke : XII + XIII + XIV

A. Tujuan Instruksional :
 3. Umum : Setelah mengikuti mata kuliah ini memberikan pengetahuan tentang area metoda pengujian software dengan menggunakan metoda black box dan white box serta menurunkan test case untuk masing-masing metoda tersebut.
 4. Khusus : Jika diberikan spesifikasi software, maka mahasiswa dapat menurunkan test case yang digunakan untuk melakukan pengujian software tersebut, termasuk membuat dokumentasi pengujian.

B. Pokok Bahasan : Tahapan Pengujian Software

C. Sub Pokok Bahasan :

1. Metoda Black Box Testing.
2. Equivalence Class Testing.
3. Endurances Testing.
4. White Box Testing.
5. Loop Testing.
6. Conditional Testing.
7. Basic Path Testing.

D. Kegiatan Belajar Mengajar dan Media Alat Pengajaran

Tahap	Kegiatan Mengajar	Kegiatan Mahasiswa	Media dan Alat Pengajar
1	2	3	4
Pendahuluan	<ol style="list-style-type: none"> 1. Menjelaskan cakupan materi dalam pertemuan yang berlangsung. 2. Menjelaskan manfaat pengujian software. 3. Menjelaskan kompetensi dalam TIU dan TIK untuk pertemuan yang berlangsung. 	<ul style="list-style-type: none"> • Memperhatikan. • Memperhatikan. • Memperhatikan. 	<ul style="list-style-type: none"> • OHP • OHP • OHP
Penyajian	4. Metoda Black Box Testing:		

Tahap	Kegiatan Mengajar	Kegiatan Mahasiswa	Pengajar
1	2	3	4
	<p>a. Menjelaskan pengertian black box testing.</p> <p>b. Menjelaskan Equivalence Class Testing.</p> <p>c. Menjelaskan Endurance Testing.</p> <p>d. Menjelaskan teknik yang digunakan untuk menurunkan test case.</p> <p>e. Menjelaskan pembuatan spesifikasi testing.</p> <p>f. Menjelaskan pelaksanaan pengujian.</p> <p>5. White Box Testing:</p> <p>a. Menjelaskan pengertian white box testing.</p> <p>b. Menjelaskan Loop Testing.</p> <p>c. Menjelaskan Conditional Testing.</p> <p>6. Basic Path Testing:</p> <p>a. Menjelaskan representasi alur logika dari suatu modul ke dalam flowgraph.</p> <p>b. Menjelaskan perhitungan kompleksitas dengan menggunakan formula jumlah titik dan garis pada flowgraph, region flowgraph, dan graph matrik flowgraph.</p> <p>c. Menjelaskan independent path.</p> <p>d. Menjelaskan penurunan test case untuk setiap independent path.</p>	<ul style="list-style-type: none"> • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. 	<ul style="list-style-type: none"> • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis.
Penutup	<p>7. Menutup pertemuan:</p> <p>a. Memberikan pertanyaan-pertanyaan kepada mahasiswa serta melakukan diskusi dalam rangka memberikan pemahaman secara lebih baik tentang materi yang diajarkan.</p>	<ul style="list-style-type: none"> • Melaksanakan, dan mengemukakan pendapat. 	<ul style="list-style-type: none"> • OHP, papan tulis

1. Analap	2. Kegiatan mengajar	3. Kegiatan mahasiswa	4. Pengajar
1	2	3	4
	b. Memberikan beberapa penguatan bagi mahasiswa. c. Menjelaskan ulang tentang hal-hal yang masih belum dimengerti oleh mahasiswa. d. Merangkum secara ringkas dan membahas keterkaitan dengan materi kuliah mendatang.		

E. Evaluasi

1. Memberikan tugas kepada mahasiswa dengan materi berupa penyusunan test case untuk setiap spesifikasi kebutuhan fungsional perangkat lunak.
2. Penilaian dilakukan minggu berikutnya setelah mahasiswa menyerahkan tugas yang diberikan.

F. Referensi

1. Davis, M., Alan, *Software Requirement Object, Functions, and States*, Fifth Edition, Prentice Hall, 1993.
2. Pressman, Roger, S., *Software Engineering A Practitioner's Approach*, Fifth Edition, Mac Graw-Hill, 2000.
3. Samorville, Ian, *Software Engineering*, Third Edition, Prentice Hall, 1996.

Mata Kuliah : Rekayasa Perangkat Lunak
 Kode Mata Kuliah : PAC130
 Bobot sks : 3
 Waktu Pertemuan : 2 x 3 x 50 menit
 Pertemuan ke : XV + XVI

- A. Tujuan Instruksional :
1. Umum : Setelah mengikuti mata kuliah ini memberikan pengetahuan tentang area metoda pengembangan software dengan metoda berorientasi objek dengan memperkenalkan berbagai metoda yang digunakan pada pengembangan berorientasi objek.
 2. Khusus : Jika diberikan permasalahan dan aturan bisnis yang akan dikembangkan software, maka mahasiswa akan dapat menyusun dokumentasi yang dihasilkan pada pengembangan software menggunakan pendekatan berorientasi objek.
- B. Pokok Bahasan : Pengembangan Berorientasi Objek
- C. Sub Pokok Bahasan : 1. Konsep Berorientasi Objek.
2. Metoda-metoda Berorientasi Objek.
3. Use Case Diagram
4. Sequences Diagram
5. Class Diagram.

D. Kegiatan Belajar Mengajar dan Media Alat Pengajaran

Tahap	Kegiatan Mengajar	Kegiatan Mahasiswa	Media dan Alat Pengajar
1	2	3	4
Pendahuluan	1. Menjelaskan cakupan materi dalam pertemuan yang sedang berlangsung. 2. Menjelaskan manfaat pengembangan berorientasi objek. 3. Menjelaskan kompetensi dalam TIU dan TIK untuk pertemuan yang sedang berlangsung.	<ul style="list-style-type: none"> • Memperhatikan. • Memperhatikan. • Memperhatikan. 	<ul style="list-style-type: none"> • OHP • OHP • OHP
Penyajian	4. Konsep Berorientasi Objek: a. Menjelaskan konsep enkapsulasi / abstraksi data.	<ul style="list-style-type: none"> • Memperhatikan, dan 	<ul style="list-style-type: none"> • OHP, papan tulis.

Isian	Kegiatan mengajar	Kegiatan mahasiswa	Pengajar
1	2	3	4
	<ul style="list-style-type: none"> b. Menjelaskan konsep inheritances. c. Menjelaskan konsep polymorphism. d. Menjelaskan tools yang digunakan untuk pengembangan software dengan metoda berorientasi objek. 5. Metoda-metoda Berorientasi Objek: <ul style="list-style-type: none"> a. Menjelaskan perbedaan mendasar metoda berorientasi objek dengan metoda konvensional. b. Menjelaskan OOA, OOD, OOAD, OMT, OOSE, Fusion, dan Objectory. 6. Use Case Diagram <ul style="list-style-type: none"> a. Menjelaskan actor. b. Menjelaskan use case. c. Menjelaskan include dan extend. 7. Sequences Diagram: <ul style="list-style-type: none"> a. Menjelaskan dinamika objek. b. Menjelaskan penyusunan sequences diagram. 8. Class Diagram: <ul style="list-style-type: none"> a. Menjelaskan bentuk statik sistem. b. Menjelaskan class. c. Menjelaskan penyusunan class diagram. 	<ul style="list-style-type: none"> mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. • Memperhatikan, dan mencatat. 	<ul style="list-style-type: none"> • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis. • OHP, papan tulis.
Penutup	<ul style="list-style-type: none"> 9. Menutup pertemuan: <ul style="list-style-type: none"> a. Memberikan pertanyaan-pertanyaan kepada mahasiswa serta melakukan diskusi dalam rangka memberikan 	<ul style="list-style-type: none"> • Melaksanakan, dan mengemukakan pendapat. 	<ul style="list-style-type: none"> • OHP, papan tulis

Isianap	Kegiatan mengajar	Kegiatan mahasiswa	Pengajar
1	2	3	4
	pemahaman secara lebih baik tentang materi yang diajarkan. b. Memberikan beberapa penguatan bagi mahasiswa. c. Menjelaskan ulang tentang hal-hal yang masih belum dimengerti oleh mahasiswa. d. Merangkum secara ringkas dan membahas keterkaitan dengan materi kuliah mendatang.		

E. Evaluasi

1. Memberikan tugas kepada mahasiswa dengan materi berupa penyusunan dokumen yang dihasilkan pada pengembangan perangkat lunak dengan menggunakan metoda berorientasi objek.
2. Penilaian dilakukan minggu berikutnya setelah mahasiswa menyerahkan tugas yang diberikan.

F. Referensi

1. Bahrami, *Object Oriented System Development*, McGraw-Hill, 2000.
2. Booch, G., J., Rumbaugh, Jacobson, *The Unified Modeling Language*, User Guide, Rational Software Corporation, Addison-Wesley, Massachusetts, USA, 1999.
3. Davis, M., Alan, *Software Requirement Object, Functions, and States*, Fifth Edition, Prentice Hall, 1993.
4. Ken Lunn, *Information Modeling and Development*, Prentice Hall, 2001.
5. Pressman, Roger, S., *Software Engineering A Practitioner's Approach*, Fifth Edition, Mac Graw-Hill, 2000.
6. Samorville, Ian, *Software Engineering*, Third Edition, Prentice Hall, 1996.