



# REKAYASA PERANGKAT LUNAK

Edy, M.Kom.

# REKAYASA PERANGKAT LUNAK

Edy, M.Kom.



Universitas Buddhi Dharma

# REKAYASA PERANGKAT LUNAK

ISBN:

Hak Cipta 2024 pada Penulis

Hak penerbitan pada UNIVERSITAS BUDDHI DHARMA. Bagi mereka yang ingin memperbanyak sebagian isi buku ini dalam bentuk atau cara apapun harus mendapatkan izin tertulis dari penulis dan penerbit UNIVERSITAS BUDDHI DHARMA.

**Penulis:**

Edy, M.Kom.

**Editor:**

Benny Daniawan, M.Kom.

**Layout**

Yusuf Kurnia, M.Kom.

**Desain sampul:**

Lidya Lunardi, S.Kom.



**Penerbit:**

UNIVERSITAS BUDDHI DHARMA

Gedung Vipassi Lt. 1 Universitas Buddhi Dharma

Jl. Imam Bonjol No 41 Karawaci Ilir, Tangerang 15115

Telp. (021) 5517853

E-Mail: lp3m@buddhidharma.ac.id

Hak Cipta dilindungi Undang-undang

All Right Reserved

Cetakan I, Desember 2024

## KATA PENGANTAR

Puji syukur ke hadirat Tuhan Yang Maha Esa atas rahmat dan karunia-Nya sehingga buku ajar berjudul *Rekayasa Perangkat Lunak* ini dapat diselesaikan. Buku ini disusun sebagai referensi pembelajaran bagi mahasiswa, praktisi, maupun siapa saja yang ingin mendalami konsep, metode, dan praktik pengembangan perangkat lunak.

Dalam era digital yang berkembang pesat, kebutuhan perangkat lunak berkualitas tinggi menjadi semakin penting. *Rekayasa Perangkat Lunak* menawarkan pendekatan sistematis untuk merancang, membangun, menguji, dan memelihara perangkat lunak yang memenuhi kebutuhan pengguna. Buku ini dirancang untuk menjembatani teori dan praktik, sehingga memberikan wawasan yang komprehensif kepada pembaca.

Materi buku mencakup pengantar dasar, siklus hidup perangkat lunak, analisis kebutuhan, desain perangkat lunak, implementasi, pengujian, hingga pemeliharaan, serta dilengkapi studi kasus dan latihan soal.

Terima kasih kami sampaikan kepada semua pihak yang mendukung penyusunan buku ini. Kritik dan saran yang membangun sangat kami harapkan untuk penyempurnaan di masa mendatang.

Semoga buku ini bermanfaat bagi pembaca dan menjadi kontribusi bagi pengembangan ilmu pengetahuan dan teknologi.

Tangerang, Desember 2024

Edy, M.Kom.

## DAFTAR ISI

KATA PENGANTAR.....	3
DAFTAR ISI.....	4
BAB 1 Pengantar Rekayasa Perangkat Lunak.....	7
A. Tujuan Pembelajaran.....	7
B. Definisi <i>Software</i> .....	8
C. Perangkat Lunak ( <i>Software</i> ) di Sekitar Kita .....	10
D. Jenis-Jenis <i>Software</i> .....	14
E. Sejarah dan Evolusi Rekayasa <i>Software</i> .....	17
F. Definisi dan Lingkup Kerja dalam Rekayasa <i>Software</i> .....	18
BAB 2 Rekayasa <i>Software</i> dan Perannya.....	21
A. Tujuan Pembelajaran.....	21
B. Peran dan Manfaat <i>Software</i> .....	22
C. Perbedaan antara <i>Hardware</i> dan <i>Software</i> .....	23
D. Mengapa Rekayasa <i>Software</i> ? .....	27
E. Soal Latihan.....	28
BAB 3 Tantangan dalam Rekayasa <i>Software</i> dan Metrik Kualitas....	29
A. Tujuan Pembelajaran.....	29
B. Masalah dalam Mitos dalam Rekayasa <i>Software</i> .....	30
C. Metrik Kualitas <i>Software</i> .....	34
D. Tantangan dalam Rekayasa <i>Software</i> .....	37
E. Soal Latihan.....	40
BAB 4 Siklus Hidup Pengembangan Perangkat Lunak (SDLC) .....	41
A. Tujuan Pembelajaran.....	41
B. <i>Software Development Life Cycle</i> (SDLC) .....	42
C. Tahap Perencanaan (Planning) .....	50
D. Tahap Kelayakan ( <i>Feasibility Analysis</i> ) .....	57
E. Studi Kasus Kelayakan.....	67
F. Soal Latihan.....	72
BAB 5 Analisis Kebutuhan.....	74
A. Tujuan Pembelajaran.....	74
B. Analisis Kebutuhan.....	75

C.	Paradigma dan Diagram dalam Analisis.....	79
BAB 6	Unified Modeling Language (UML).....	83
A.	Tujuan Pembelajaran.....	83
B.	Unified Modeling Language (UML).....	84
C.	Diagram-diagram UML.....	85
D.	Artefak dalam SDLC.....	88
BAB 7	<i>Use Case, Activity, dan Sequence Diagram</i> .....	91
A.	Tujuan Pembelajaran.....	91
B.	<i>Use Case Diagram</i> .....	92
C.	<i>Activity Diagram</i> .....	97
D.	<i>Sequence Diagram</i> .....	106
E.	Evaluasi / Soal Latihan.....	113
BAB 8	<i>Design</i> .....	115
A.	Tujuan Pembelajaran.....	115
B.	Pengantar Desain.....	116
C.	Elemen Diagram Kelas.....	120
D.	Atribut dalam Diagram Kelas.....	124
E.	Hubungan Multiplikasi.....	127
F.	Contoh Diagram Kelas: Studi Kasus ATM.....	129
BAB 9	Desain ( <i>User Interface dan Deployment Diagram</i> ).....	134
A.	Tujuan Pembelajaran.....	134
B.	Desain Antar Muka.....	135
C.	Diagram Implementasi.....	140
D.	Model Data.....	145
E.	Soal Latihan.....	149
BAB 10	Implementasi.....	153
A.	Tujuan Pembelajaran.....	153
B.	Pengantar Implementasi.....	154
C.	Konstruksi ( <i>Construction</i> ).....	157
D.	Pengujian ( <i>Testing</i> ).....	160
E.	Dokumentasi.....	164
F.	Instalasi.....	170
G.	Studi Kasus Implementasi.....	173

DAFTAR PUSTAKA.....	177
GLOSARIUM.....	180
INDEKS.....	186
HASIL SCANNING SIMILARITY .....	189
BIOGRAFI PENULIS.....	190

## GLOSARIUM

- Activity Diagram* : Diagram UML yang memodelkan alur kerja atau logika bisnis dalam sistem, menunjukkan urutan aktivitas, keputusan, dan aliran kontrol.
- Agile* : Pendekatan pengembangan *Software* yang fleksibel, berfokus pada iterasi pendek dan kolaborasi tim.
- Agregasi  
(Aggregation) : Relasi antar kelas yang menunjukkan bahwa satu kelas adalah bagian dari kelas lain, tetapi keduanya dapat berdiri sendiri.
- Aksesibilitas  
(*Accessibility*) : Kemampuan antarmuka untuk dapat diakses oleh pengguna dengan berbagai keterbatasan fisik atau sensorik, seperti melalui penggunaan teks alternatif, navigasi keyboard, atau ukuran teks yang dapat disesuaikan.
- Analisis Kebutuhan : Proses untuk mengidentifikasi dan mendefinisikan kebutuhan pengguna, baik fungsional maupun non-fungsional, untuk memastikan sistem yang dikembangkan sesuai dengan harapan pengguna.
- Antarmuka  
Pengguna (*User Interface/UI*) : Bagian dari sistem perangkat lunak yang berfungsi sebagai penghubung antara pengguna dan sistem, mencakup elemen-elemen visual dan interaktif untuk mempermudah penggunaan sistem.
- Atribut : Properti atau karakteristik dari suatu

- kelas yang digunakan untuk menyimpan informasi tentang objek yang dibuat dari kelas tersebut.
- Automasi : Proses menggunakan teknologi untuk mengurangi intervensi manusia dalam menyelesaikan tugas berulang.
- Basis Data (*Database*) : Sistem yang digunakan untuk menyimpan, mengelola, dan mengakses data secara efisien, biasanya terdiri dari tabel-tabel yang saling terkait.
- BEP (*Break-Even Point*) : Titik di mana total pendapatan proyek setara dengan total biaya yang diinvestasikan, menunjukkan kapan proyek mulai menghasilkan keuntungan.
- Bug* : Kesalahan atau kekurangan dalam kode *Software* yang dapat menyebabkan malfungsi atau hasil tak terduga.
- CAD (*Computer-Aided Design*) : *Software* yang digunakan untuk membuat desain 2D dan 3D, seperti AutoCAD.
- CMMI (*Capability Maturity Model Integration*) : Model kerangka kerja yang digunakan untuk mengukur dan meningkatkan proses pengembangan *Software* secara sistematis.
- DevOps* : Praktik yang mengintegrasikan pengembangan dan operasi untuk meningkatkan efisiensi melalui otomatisasi.
- Diagram Alir (Flowchart)* : Representasi grafis dari langkah-langkah dalam suatu proses atau sistem, menggunakan simbol seperti

- oval, persegi panjang, dan belah ketupat untuk menggambarkan tindakan dan keputusan.
- Dokumentasi : Kumpulan informasi yang menjelaskan cara kerja dan penggunaan perangkat lunak, seperti manual pengguna atau komentar kode.
- Entitas (*Entity*) : Unit atau objek utama dalam model data yang mewakili sesuatu dengan keberadaan unik, seperti "User," "Product," atau "Transaction."
- Entity-Relationship Diagram* (ERD) : Diagram yang digunakan untuk menggambarkan hubungan antar entitas dalam sebuah model data, mencakup atribut, relasi, dan aturan integritas.
- Firmware* : *Software* tertanam dalam perangkat keras yang dirancang untuk menjalankan fungsi tertentu.
- Hardware* : Komponen fisik dari sistem komputer yang mendukung operasional *Software*.
- Open-Source Software* : *Software* dengan kode sumber yang dapat diakses, dimodifikasi, dan didistribusikan secara bebas oleh publik.
- Proprietary Software* : *Software* dengan kode sumber tertutup yang penggunaannya diatur oleh lisensi tertentu.
- Prototyping* : Metode pengembangan *Software* yang menciptakan model awal untuk diuji dan ditingkatkan sebelum produk akhir dikembangkan sepenuhnya.
- Rekayasa *Software* : Disiplin ilmu yang mencakup semua

- proses yang terlibat dalam pengembangan, pengujian, dan pemeliharaan perangkat lunak.
- ROI (*Return on Investment*) : Rasio yang mengukur keuntungan proyek dibandingkan dengan biaya yang diinvestasikan, digunakan untuk mengevaluasi kelayakan ekonomi proyek.
- RPA (*Robotic Process Automation*) : Teknologi yang digunakan untuk mengotomatisasi proses bisnis menggunakan robot perangkat lunak.
- SDLC (*Software Development Life Cycle*) : Kerangka kerja sistematis dan terstruktur yang digunakan untuk mengelola seluruh proses pengembangan perangkat lunak dari perencanaan hingga pemeliharaan.
- Sequence Diagram* : Diagram UML yang memodelkan interaksi antara objek dalam urutan waktu, menunjukkan pesan-pesan yang dikirim untuk mencapai fungsi tertentu.
- Sistem Operasi : *Software* sistem yang mengelola sumber daya *hardware* dan menyediakan layanan dasar untuk *Software* aplikasi, seperti Windows atau Linux.
- Software* : Kumpulan instruksi yang dirancang untuk menjalankan tugas tertentu pada komputer atau perangkat lainnya.
- Standar ISO 9126 : Standar internasional yang mengukur kualitas *Software* berdasarkan enam karakteristik utama: fungsionalitas, keandalan, kegunaan, efisiensi,

- Testing : pemeliharaan, dan portabilitas.  
: Proses evaluasi perangkat lunak untuk mendeteksi kesalahan dan memastikan bahwa perangkat lunak berfungsi sesuai dengan kebutuhan pengguna.
- UML (*Unified Modeling Language*) : Bahasa pemodelan standar untuk memvisualisasikan, menspesifikasikan, membangun, dan mendokumentasikan artefak perangkat lunak, mencakup berbagai diagram untuk menggambarkan aspek statis dan dinamis sistem.
- Usability (*Kegunaan*) : Seberapa mudah dan efisien pengguna dapat memanfaatkan *Software* dalam memenuhi kebutuhan mereka.
- Use Case Diagram* : Diagram UML yang menggambarkan fungsionalitas sistem dari perspektif pengguna, menunjukkan aktor, *use case*, dan interaksi antara aktor dan sistem.
- Waterfall* : Model pengembangan *Software* yang berurutan, mulai dari analisis kebutuhan hingga pemeliharaan.

## INDEKS

---

### A

Activity · 80, 81, 83, 86, 88,  
89, 90, 96, 98, 112

Agile · 18, 38, 171

Agregasi · 121, 130

Analisis · 19, 42, 44, 45, 51,  
54, 56, 57, 59, 62, 64, 66,  
67, 68, 69, 70, 72, 73, 74,  
75, 76, 77, 78, 87, 134, 144,  
148, 149, 154, 174

Aplikasi · 15, 16, 23, 58, 67,  
142, 147

Atribut · 118, 119, 122, 123,  
124, 125, 128, 129, 144,  
146

---

### D

Data · 9, 13, 33, 47, 78, 79, 88,  
127, 133, 137, 143, 145,  
146, 147, 148, 149, 161,  
174, 175

Desain · 2, 12, 20, 39, 40, 47,  
73, 75, 77, 88, 113, 114,  
115, 116, 117, 119, 120,  
127, 132, 133, 135, 136,  
137, 138, 148, 152, 153

Design · 23, 42, 44, 46, 47, 88,  
113, 137

Diagram · 46, 78, 79, 80, 81,  
82, 83, 84, 85, 86, 87, 88,

89, 90, 91, 92, 93, 95, 96,  
98, 99, 100, 101, 102, 103,  
104, 105, 107, 108, 109,  
110, 111, 112, 114, 118,  
119, 120, 121, 122, 128,  
130, 131, 132, 133, 138,  
139, 140, 141, 142, 145,  
148, 149, 154, 161, 162, 163

Dokumentasi · 10, 20, 31, 34,  
37, 40, 48, 50, 74, 76, 78,  
88, 151, 152, 153, 157, 160,  
161, 162, 163, 164, 165,  
167, 171

---

### F

Feasibility · 43, 45, 57, 58, 59,  
62, 67, 68, 69, 72, 87

---

### H

Hardware · 13, 14, 16, 21, 22,  
24, 25, 26, 27, 28, 33, 175

---

### I

Implementasi · 19, 20, 37, 44,  
60, 73, 75, 127, 135, 138,  
139, 140, 141, 142, 148,  
150, 151, 152, 153, 158,  
159, 160, 169, 171, 172,  
175

ISO · 8, 36, 37, 165

---

**K**

Kebutuhan · 19, 33, 38, 45, 46,  
51, 52, 53, 55, 57, 58, 67,  
73, 74, 75, 76, 77, 117, 134,  
139, 144, 153, 163  
Kelayakan · 45, 57, 58, 59, 62,  
67, 68, 69, 72

---

**L**

Linux · 14, 16, 17  
Lunak · 7, 8, 10, 42, 43, 48, 49,  
59, 82, 113, 117, 150, 151,  
152, 158, 161, 169, 174,  
175, 184, 185

---

**M**

Metrik · 19, 30, 34  
Microsoft · 15, 16, 17, 23  
Mitos · 30, 31, 32, 33, 34  
Model · 18, 35, 36, 47, 88, 133,  
143, 145, 146, 147, 148,  
149, 161, 174

---

**O**

Oriented · 78, 79, 80, 115,  
174, 175

---

**P**

Perangkat · 7, 8, 10, 11, 17,  
42, 43, 48, 49, 59, 82, 113,  
117, 150, 151, 152, 158,  
161, 169, 174, 175, 184,  
185

---

**R**

Rekayasa · 7, 16, 17, 18, 21,  
22, 27, 28, 30, 31, 34, 38,  
82, 161, 169, 175, 184, 185

---

**S**

SDLC · 42, 43, 44, 48, 49, 50,  
83, 87, 113, 114, 116, 150,  
151, 153  
Sistem · 11, 12, 13, 15, 16, 17,  
25, 45, 47, 48, 51, 52, 53,  
54, 55, 60, 67, 70, 73, 74,  
76, 82, 89, 90, 95, 98, 100,  
101, 103, 104, 107, 108,  
109, 110, 111, 127, 132,  
133, 137, 139, 141, 142,  
143, 146, 147, 152, 154,  
155, 157, 158, 161, 162,  
165, 166, 168, 169, 172,  
174, 175  
Software · 8, 9, 10, 11, 12, 13,  
14, 15, 16, 17, 18, 19, 20,  
21, 22, 23, 24, 25, 26, 27,  
28, 29, 30, 31, 32, 33, 34,  
35, 36, 37, 38, 39, 40, 41,  
42, 43, 44, 48, 50, 83, 87,  
113, 114, 150, 151, 174,  
175, 185  
Standar · 8, 37, 155, 161, 162,  
163

---

***T***

Testing · 48, 88, 135, 157,  
158, 159, 160, 174

---

***W***

Windows · 11, 14, 16, 17

## HASIL SCANNING SIMILARITY

Berdasarkan penilaian teks pada Buku Ajar yang diajukan di bawah ini:

Penulis : Edy, M.Kom.  
Fakultas : Sains dan Teknologi/Teknik Perangkat Lunak  
Judul : Rekayasa Perangkat Lunak  
Tipe : Buku

Turnitin mencatat adanya tingkat kesamaan antara dokumen ini dengan dokumen yang ada dalam aplikasi seperti yang tertera di bawah ini:

Word Count : 29.371  
Character Count : 301.083  
**Similarity Index : 11%**  
Internet Source : 10%  
Publication : 3%  
Student paper : 3%  
Exclude Quotes : Off  
Exclude Bibliography : On  
Exclude Matches : Off



## BIOGRAFI PENULIS



Penulis lahir pada tanggal 28 Desember 1982 di Pematang Siantar, Sumatera Utara. Penulis menempuh Pendidikan SMA Negeri 51 Jakarta, lalu melanjutkan studinya di Program Studi Teknik Informatika di Universitas Gunadarma pada tahun 2000 - 2004, dan pada tahun 2012 - 2012 penulis melanjutkan studinya di STMIK Eresha dengan konsentrasi di bidang *Software Engineering*. Selama menjadi mahasiswa penulis aktif dalam pengembangan *Software* selama 6 tahun di perusahaan Berca dan sudah menjadi minat penulis sejak awal lulus perkuliahan Sarjana. Penulis juga pernah bekerja di PT Axa Indonesia pada tahun 2010 dibidang *Database Administrator*. Kemudian berbekal pengalaman bekerja di dunia *industry*, penulis memutuskan untuk mengabdikan diri sepenuhnya pada dunia Pendidikan sejak 2010 hingga sekarang di Universitas Buddhi Dharma menjadi dosen tetap di Program Studi Teknik Perangkat Lunak dengan konsentrasi pengajaran di bidang Rekayasa Perangkat Lunak dan Penjaminan Kualitas Perangkat Lunak.

# REKAYASA PERANGKAT LUNAK



Buku Rekayasa Perangkat Lunak menghadirkan panduan terstruktur untuk memahami konsep dasar hingga lanjutan dalam pengembangan perangkat lunak. Mulai dari pengenalan perangkat lunak, sejarah, dan jenis-jenisnya, hingga pembahasan siklus hidup pengembangan perangkat lunak (SDLC), buku ini mengajak pembaca mengeksplorasi setiap tahapan penting, seperti perencanaan, analisis kebutuhan, desain, hingga implementasi. Topik-topik seperti Unified Modeling Language (UML), diagram use case, dan desain antarmuka juga dibahas secara detail untuk membantu pembaca memahami teknik perancangan modern.

Dilengkapi dengan studi kasus dan soal latihan di setiap bab, buku ini tidak hanya menawarkan teori tetapi juga penerapan praktis untuk mempersiapkan pembaca menghadapi tantangan nyata di dunia rekayasa perangkat lunak. Cocok untuk mahasiswa dan profesional, buku ini menjadi rujukan yang relevan bagi siapa saja yang ingin mendalami pengembangan perangkat lunak dengan pendekatan yang efektif dan terarah.



**UNIVERSITAS BUDDHI DHARMA**  
Gedung Vipassi Lt. 1  
Jl. Imam Bonjol No. 41 Karawaci Ilir  
Tangerang 15115  
Telp. (021) 5517853  
E-mail: lp3m@buddhidharma.ac.id





# REKAYASA PERANGKAT LUNAK

Edy, M.Kom.

# REKAYASA PERANGKAT LUNAK



Buku Rekayasa Perangkat Lunak menghadirkan panduan terstruktur untuk memahami konsep dasar hingga lanjutan dalam pengembangan perangkat lunak. Mulai dari pengenalan perangkat lunak, sejarah, dan jenis-jenisnya, hingga pembahasan siklus hidup pengembangan perangkat lunak (SDLC), buku ini mengajak pembaca mengeksplorasi setiap tahapan penting, seperti perencanaan, analisis kebutuhan, desain, hingga implementasi. Topik-topik seperti Unified Modeling Language (UML), diagram use case, dan desain antarmuka juga dibahas secara detail untuk membantu pembaca memahami teknik perancangan modern.

Dilengkapi dengan studi kasus dan soal latihan di setiap bab, buku ini tidak hanya menawarkan teori tetapi juga penerapan praktis untuk mempersiapkan pembaca menghadapi tantangan nyata di dunia rekayasa perangkat lunak. Cocok untuk mahasiswa dan profesional, buku ini menjadi rujukan yang relevan bagi siapa saja yang ingin mendalami pengembangan perangkat lunak dengan pendekatan yang efektif dan terarah.



UNIVERSITAS BUDDHI DHARMA  
Gedung Vipassi Lt. 1  
Jl. Imam Bonjol No. 41 Karawaci Ilir  
Tangerang 15115  
Telp. (021) 5517853  
E-mail: lp3m@buddhidharma.ac.id



REKAYASA PERANGKAT LUNAK

Edy, M.Kom.



# REKAYASA PERANGKAT LUNAK

Edy, M.Kom.