



Rekayasa Perangkat Lunak

Widiyawati, Nazaruddin Ahmad, Eka Hartati, Erly Krisnanik, Yuniansyah, Yunita Ardilla, Iin Ernawati, Guntoro, Kraugusteeliana, I Wayan Widi Pradnyana, Titus Kristanto, D Tri Octafian, Irwanto.

Rekayasa **Perangkat Lunak**

Widiyawati, Nazaruddin Ahmad, Eka Hartati, Erly Krisnanik, Yuniansyah,
Yunita Ardilla, Iin Ernawati, Guntoro, Kraugusteeliana, I Wayan Widi Pradnyana,
Titus Kristanto, D Tri Octafian, Irwanto.

Editor: Nazaruddin Ahmad



REKAYASA PERANGKAT LUNAK

Tim Penulis:

**Widiyawati, Nazaruddin Ahmad, Eka Hartati, Erly Krisnanik, Yuniansyah,
Yunita Ardilla, Iin Ernawati, Guntoro, Kraugusteeliana, I Wayan Widi Pradnyana,
Titus Kristanto, D Tri Octafian, Irwanto.**

Desain Cover:

Fawwaz Abyan

Tata Letak:

Handarini Rohana

Editor:

Nazaruddin Ahmad

Proofreader:

Aas Masruroh

ISBN:

978-623-459-115-6

Cetakan Pertama:

Juni, 2022

Hak Cipta 2022, Pada Penulis

Hak Cipta Dilindungi Oleh Undang-Undang

Copyright © 2022

by Penerbit Widina Bhakti Persada Bandung

All Right Reserved

Dilarang keras menerjemahkan, memfotokopi, atau memperbanyak sebagian atau seluruh isi buku ini tanpa izin tertulis dari Penerbit.

PENERBIT:

WIDINA BHAKTI PERSADA BANDUNG

(Grup CV. Widina Media Utama)

Komplek Puri Melia Asri Blok C3 No. 17 Desa Bojong Emas
Kec. Solokan Jeruk Kabupaten Bandung, Provinsi Jawa Barat

Anggota IKAPI No. 360/JBA/2020

Website: www.penerbitwidina.com

Instagram: [@penerbitwidina](https://www.instagram.com/penerbitwidina)

PRAKATA EDITOR

Rasa syukur yang teramat dalam dan tiada kata lain yang patut kami ucapkan selain mengucapkan kata syukur. Karena berkat rahmat dan karunia Tuhan Yang Maha Esa, buku yang berjudul “*Rekayasa Perangkat Lunak*” telah selesai disusun dan berhasil diterbitkan, semoga buku ini dapat memberikan sumbangsih keilmuan dan penambah wawasan bagi siapa saja yang memiliki minat terhadap pembahasan tentang *Rekayasa Perangkat Lunak*.

Akan tetapi pada akhirnya kami mengakui bahwa tulisan ini terdapat beberapa kekurangan dan jauh dari kata sempurna, sebagaimana pepatah menyebutkan “*tiada gading yang tak retak*” dan sejatinya kesempurnaan hanyalah milik Tuhan semata. Maka dari itu, kami dengan senang hati secara terbuka untuk menerima berbagai kritik dan saran dari para pembaca sekalian, hal tersebut tentu sangat diperlukan sebagai bagian dari upaya kami untuk terus melakukan perbaikan dan penyempurnaan karya selanjutnya di masa yang akan datang.

Terakhir, ucapan terima kasih kami sampaikan kepada seluruh pihak yang telah mendukung dan turut andil dalam seluruh rangkaian proses penyusunan dan penerbitan buku ini, sehingga buku ini bisa hadir di hadapan sidang pembaca. Semoga buku ini bermanfaat bagi semua pihak dan dapat memberikan kontribusi bagi pengembangan ilmu pengetahuan di Indonesia.

Juni, 2022

Tim Penulis

DAFTAR ISI

| | |
|--|------------|
| PRAKATA EDITOR | iii |
| DAFTAR ISI | iv |
| BAB 1 KONSEP DASAR REKAYASA PERANGKAT LUNAK | 1 |
| A. Pendahuluan..... | 2 |
| B. Pengertian Rekayasa Perangkat Lunak | 3 |
| C. Sejarah Rekayasa Perangkat Lunak | 7 |
| D. Sifat dan Karakteristik Perangkat Lunak | 11 |
| E. Tipe-Tipe Perangkat Lunak | 15 |
| F. Peranan Perangkat Lunak | 17 |
| G. Mitos Perangkat Lunak | 18 |
| H. Rangkuman Materi | 21 |
| BAB 2 MODEL PROSES PENGEMBANGAN PERANGKAT LUNAK | 25 |
| A. Pendahuluan..... | 26 |
| B. Proses Rekayasa Perangkat Lunak..... | 27 |
| C. Karakteristik Proses Perangkat Lunak | 29 |
| D. Model Proses Pengembangan Perangkat Lunak | 29 |
| E. Rangkuman Materi | 40 |
| BAB 3 KONSEP DAN PRINSIP DESAIN PERANGKAT LUNAK | 45 |
| A. Pendahuluan..... | 46 |
| B. Desain Perangkat Lunak | 47 |
| C. Konsep Desain Perangkat Lunak..... | 50 |
| D. Prinsip Desain Perangkat Lunak | 56 |
| E. Rangkuman Materi | 57 |
| BAB 4 ANALISIS KEBUTUHAN PENGGUNA (USER REQUIREMENT ANALYSIS) | 61 |
| A. Pendahuluan..... | 62 |
| B. Teknik Pengumpulan Data | 63 |
| C. Studi Kelayakan | 67 |
| D. Rangkuman Materi | 79 |
| BAB 5 ANALISA DAN REKAYASA KEBUTUHAN PERANGKAT LUNAK | 83 |
| A. Pendahuluan..... | 84 |

| | | |
|---|---|------------|
| B. | Perangkat Lunak | 85 |
| C. | Analisa dan Rekayasa Kebutuhan Perangkat Lunak | 89 |
| D. | Tahapan Analisis Kebutuhan | 90 |
| E. | Metode Metode Analisis | 91 |
| F. | Kegiatan Pada Rekayasa Kebutuhan Perangkat Lunak..... | 93 |
| G. | Rangkuman Materi | 95 |
| BAB 6 SPESIFIKASI KEBUTUHAN PERANGKAT LUNAK | | 99 |
| A. | Pendahuluan..... | 100 |
| B. | Pengantar | 106 |
| C. | Deskripsi | 107 |
| D. | Persyaratan..... | 108 |
| E. | Indeks dan Lampiran | 109 |
| F. | Rangkuman Materi | 110 |
| BAB 7 PEMODELAN ANALISIS (DATA FLOW DIAGRAM DAN DATA DICTIONARY) | | 113 |
| A. | Pendahuluan..... | 114 |
| B. | Pemodelan (Konsep Dasar dan Pengertian)..... | 114 |
| C. | Bentuk Pemodelan (Model Vs Diagram) | 116 |
| D. | Pemodelan Kebutuhan (Terstruktur & Orientasi Objek)..... | 119 |
| E. | Data Flow Diagram (DFD) | 121 |
| F. | Data Dictionary (Kamus Data) | 124 |
| G. | Rangkuman Materi | 128 |
| BAB 8 METODE STRUCTURED SYSTEM ANALYSIS AND DESIGN | | 131 |
| A. | Pendahuluan..... | 132 |
| B. | Sejarah | 133 |
| C. | Elemen Desain Terstruktur | 133 |
| D. | Ciri-Ciri Analisis dan Desain Terstruktur | 134 |
| E. | Metodologi Desain Terstruktur | 135 |
| F. | Kelebihan Analisis dan Desain Terstruktur | 136 |
| G. | Kekurangan Teknik Analisis dan Desain Terstruktur (ADT) | 136 |
| H. | Alat Pengembangan Grafik ADT | 138 |
| I. | Bagan Analisis dan Desain Terstruktur | 139 |
| J. | Teknik Analisis dan Desain Terstruktur | 140 |
| K. | Metode Structured Systems Analysis and Design | 142 |

| | |
|---|------------|
| L. Teknik SSAD | 142 |
| M. Tahapan | 143 |
| N. Rangkuman Materi | 147 |
| BAB 9 METODE PERANCANGAN BERBASIS OBJECT (OOAD) | 149 |
| A. Pendahuluan..... | 150 |
| B. Konsep OOAD (Object Oriented Analysis and Design) | 152 |
| C. UML dan Simbol | 153 |
| D. Perbedaan Terstruktur Vs Objek Oriented | 163 |
| E. Kelebihan dan Kekurangan Ooad | 165 |
| F. Software Yang Digunakan – UML | 166 |
| G. Rangkuman Materi | 166 |
| BAB 10 METODE PEMODELAN UNIFIED MODELING LANGUAGE | 169 |
| A. Pendahuluan..... | 170 |
| B. Mengenai Unified Modeling Language (UML)..... | 170 |
| C. Spesifikasi UML..... | 173 |
| D. Diagram Dengan UML..... | 174 |
| E. Perangkat Lunak Untuk Pemodelan Dengan UML | 184 |
| F. Siklus Tahapan Pemodelan Dengan UML | 184 |
| G. Rangkuman Materi | 188 |
| BAB 11 ANTARMUKA PENGGUNA (USER INTERFACE)..... | 191 |
| A. Pendahuluan..... | 192 |
| B. Desain Interaksi (<i>Interaction Design</i>) | 192 |
| C. Antarmuka Pengguna (<i>User Interface</i>) | 196 |
| D. <i>User Experience</i> | 201 |
| E. Rangkuman Materi | 206 |
| BAB 12 METODE PENGUJIAN PERANGKAT LUNAK..... | 211 |
| A. Pendahuluan..... | 212 |
| B. Pengenalan Software Development Life Cycle..... | 213 |
| C. Pengenalan Software Testing (Penguji-an Perangkat Lunak) | 213 |
| D. Proses Pengujian Perangkat Lunak..... | 215 |
| E. Tingkat Pengujian | 220 |
| F. Jenis Pengujian | 222 |
| G. Menulis Skenario Uji..... | 223 |
| H. Menulis Kasus Uji..... | 226 |

| | |
|--|------------|
| I. Teknik Uji Black Box | 228 |
| J. Teknik Uji White Box | 231 |
| K. Rangkuman Materi | 232 |
| BAB 13 METODE PEMELIHARAAN PERANGKAT LUNAK..... | 235 |
| A. Pendahuluan..... | 236 |
| B. Teknik Pemeliharaan Perangkat Lunak | 242 |
| C. Siklus Hidup Pemeliharaan Sistem (SMLC) | 248 |
| D. Kategori Pemeliharaan Perangkat Lunak | 253 |
| E. Prosedur Untuk Memelihara Sistem | 256 |
| F. Permasalahan Pemeliharaan Perangkat Lunak | 256 |
| G. Model Pemeliharaan Perangkat Lunak..... | 258 |
| H. Perawatan Dan Konfigurasi | 260 |
| I. Karakteristik Perangkat Lunak Yang Mudah Dalam Pemeliharaan..... | 261 |
| J. Proses Pemeliharaan Perangkat Lunak | 264 |
| K. Manajemen Pemeliharaan Perangkat Lunak..... | 267 |
| L. Rangkuman Materi | 270 |
| BAB 14 PENJAMINAN MUTU PERANGKAT LUNAK | 275 |
| A. Pendahuluan..... | 276 |
| B. Review (Ulasan) | 276 |
| C. Standar dan Model | 280 |
| D. Rangkuman Materi | 286 |
| GLOSARIUM | 288 |
| PROFIL PENULIS | 296 |



REKAYASA PERANGKAT LUNAK

BAB 1: KONSEP DASAR

REKAYASA PERANGKAT LUNAK

Widiyawati, S.Kom., M.Kom

STMIK BANI SALEH Bekasi

BAB 1

KONSEP DASAR REKAYASA PERANGKAT LUNAK

A. PENDAHULUAN

Teknologi pembuatan program perangkat lunak berkembang dengan pesatnya, seiring dengan berkembangnya infrastruktur dan perangkat keras yang tersedia pada saat ini. Rekayasa perangkat atau software engineering merupakan pembangunan dengan menggunakan prinsip atau konsep rekayasa dengan tujuan menghasilkan perangkat lunak yang bernilai ekonomi yang dipercaya dan bekerja secara efisiensi dengan menggunakan mesin komputer. Rekayasa perangkat lunak dibutuhkan agar perangkat lunak yang dibuat tidak hanya menjadi perangkat lunak yang tidak terpakai. Perangkat lunak yang baik adalah perangkat lunak yang fokus pada penggunaan dan pelanggan. Rekayasa perangkat lunak lebih difokuskan pada pengembangan perangkat lunak dan mengirimkan perangkat lunak yang bermanfaat kepada pelanggan dan pengguna.

Perangkat lunak atau software adalah program komputer yang terasosiasi dengan dokumentasi perangkat lunak seperti dokumentasi kebutuhan, model desain dan cara penggunaan user manual. Sistem perangkat lunak berarti adalah sebuah sistem yang memiliki komponen berupa perangkat lunak yang memiliki hubungan satu sama lainnya untuk dapat memenuhi kebutuhan pelanggan (customer). Pelanggan adalah orang atau organisasi yang memesan atau membeli perangkat lunak dari pengembang perangkat lunak. User atau pemakai perangkat lunak adalah

DAFTAR PUSTAKA

- EBookPressman__Software_Engineering 5th [BRO75] Brooks, F., The Mythical Man-Month, Addison-Wesley, 1975. [DEJ98] De Jager, P. et al., Countdown Y2K: Business Survival Planning for the Year 2000, Wiley, 1998.
- [GLA98] Glass, R.L., "Is There Really a Software Crisis?" IEEE Software, vol. 15, no. 1, January 1998, pp. 104–105.
- Rekayasa perangkat lunak , Pendekatan Praktisi edisi 7, Roger S, Pressman,Ph.D
- Judul asli SOFTWARE ENGINEERING : A practitioner Approach,Seven Edition ,tahun 2010 penerbit ANDI Yogyakarta .
- Rekayasa Perangkat Lunak ber orientasi Objek menggunakan PHP, oleh Prof. Dr. Ir. Riri Fitri Sari, MM, M.Sc, DTM, SMIEEE, Ardiati Utami S , 2021 ,PENERBIT ANDI (Anggota IKAPI) Yogyakarta



REKAYASA PERANGKAT LUNAK

BAB 2: MODEL PROSES

PENGEMBANGAN PERANGKAT LUNAK

Nazaruddin Ahmad, M.T.

Universitas Islam Negeri Ar-Raniry Banda Aceh

BAB 2

MODEL PROSES PENGEMBANGAN PERANGKAT LUNAK

A. PENDAHULUAN

Kegiatan rekayasa perangkat lunak sangat membutuhkan proses dokumentasi yang baik, sehingga setiap hal yang akan dikerjakan pada proses pengembangan perangkat lunak dapat dilakukan dengan baik dan teratur. Hasil yang ingin diharapkan adalah adanya produk perangkat lunak yang baik, bebas dari kesalahan, dapat menyelesaikan pekerjaan secara komputerisasi dengan baik. Untuk mendapatkan hasil produk perangkat lunak yang baik maka pengembang perangkat lunak harus memahami model proses pengembangan perangkat lunak dengan baik. Sebelum melakukan pengembangan perangkat lunak maka harus didahului dengan proses perencanaan pengembangan perangkat lunak dengan lengkap sehingga spesifikasi perangkat lunak yang diinginkan oleh pelanggan akan dapat tergambarkan dengan jelas. Proses perencanaan merupakan langkah paling awal yang harus dilakukan oleh pengembang perangkat lunak.

Secara umum proses pengembangan perangkat lunak itu terdiri dari tahapan pengumpulan spesifikasi (*specification*), pengembangan (*development*), validasi (*validation*), dan evolusi (*evolution*). Empat tahapan umum ini dapat dikembangkan menjadi tahapan-tahapan lainnya untuk melengkapi proses model pengembangan perangkat lunak yang disesuaikan dengan model-model pengembangan perangkat lunak yang dikembangkan dan digunakan.

DAFTAR PUSTAKA

- A.S, R., & Shalahuddin, M. (2019). *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*. Penerbit INFORMATIKA.
- Bolung, M., Ronald, H., & Tampangela, K. (2017). Analisa penggunaan metodologi pengembangan perangkat lunak. *Jurnal ELTIKOM*, 1(1), 1–10.
- Budi, D. S., Siswa, T. A. Y., & Abijono, H. (2016). Analisis Pemilihan Penerapan Proyek Metodologi Pengembangan Rekayasa Perangkat Lunak. *Teknika*, 5(1), 24–31. <https://doi.org/10.34148/teknika.v5i1.48>
- Dewi, A. O. P. (2018). Daur Hidup Pengembangan Sistem untuk Otomasi Perpustakaan. *Anuva*, 2(4), 347. <https://doi.org/10.14710/anuva.2.4.347-354>
- Fadli, S., Putra, S. A., & Saleh, M. (2019). Rancang Bangun Sistem Pengolahan Data Obat Menggunakan Model Spiral. *Jurnal Manajemen Informatika Dan Sistem Informasi*, 2(1), 36. <https://doi.org/10.36595/misi.v2i1.77>
- Parlika, R., Hidayat, M. F., Putra, H. R., Satria, V. H., Lesmana, H. F., & Pralas, F. H. (2018). Studi Komparatif Model Proses Perangkat Lunak Terhadap Karakteristik Sistem ERP. *Jurnal IPTEK Media Komunikasi Teknologi*, 22(2), 1–8. <https://doi.org/10.31284/j.iptek.2018.v22i2.252>
- Purnomo, D. (2017). Model Prototyping Pada Pengembangan Sistem Informasi. *J I M P - Jurnal Informatika Merdeka Pasuruan*, 2(2), 54–61. <https://doi.org/10.37438/jimp.v2i2.67>
- Sagala, J. R. (2018). Model Rapid Application Development (RAD) Dalam Pengembangan Sistem Informasi Penjadwalan Belajar Mengajar. *Jurnal Mantik Penusa*, 2(1), 87–90.
- Sudarma, M. (2012). Manajemen Proyek Teknologi Informasi. In J. Atmaja (Ed.), *Laboratorium Penelitian dan Pengembangan FARMAKA TROPIS Fakultas Farmasi Universitas Muallawarman, Samarinda, Kalimantan Timur* (Issue April). Udayana University Press.

- Taqwiym, A. (2020). Penerapan Metode Iterative Pada Perancangan Sistem Pembukuan Penjualan Pt.Xyz. *Jurnal Sisfokom (Sistem Informasi Dan Komputer)*, 9(2), 254–263. <https://doi.org/10.32736/sisfokom.v9i2.895>
- Trimiharta, K. A., Dharma, E. M., & Pudyanti, A. A. A. R. (2021). Rancang Bangun Sistem Penggajian Berbasis WEB Dengan Model Sekuensial Linier. *Jurnal TEKINKOM*, 4, 318–327.
- Utomo, A. N., & Alfaridzi, M. (2018). Perancangan Sistem Informasi Pada Percetakan CV Citra Kencana Jakarta Timur Berbasis WEB. *Jurnal Rekayasa Informasi*, 7(1), 43–47.



REKAYASA PERANGKAT LUNAK

BAB 3: KONSEP DAN PRINSIP DESAIN PERANGKAT LUNAK

Eka Hartati, M.Kom

Institut Teknologi dan Bisnis PalComTech Palembang

BAB 3

KONSEP DAN PRINSIP DESAIN PERANGKAT LUNAK

A. PENDAHULUAN

Saat ini perangkat lunak banyak digunakan, dikarenakan perangkat lunak merupakan salah satu usaha untuk menyelesaikan masalah di dunia nyata. Desain perangkat lunak merupakan hal yang penting dalam pembuatan sistem. Desain perangkat lunak ini sendiri pada dasarnya berada pada inti teknik dari proses rekayasa perangkat lunak dan diaplikasikan tanpa harus memperhatikan model proses yang akan digunakan. Model desain terdiri dari desain data, desain arsitektur, desain interface, dan desain prosedural. Dapat disimpulkan bahwa desain berfungsi sebagai hal yang mendasar bagi semua rekayasa perangkat lunak dan langkah-langkah dalam pemeliharaan. Pada dasarnya konsep desain sangat berkembang sampai saat ini, meskipun untuk saat ini tingkat minat konsep desain bervariasi, masing-masing minat mengalami seiring berjalannya waktu. Masing-masing desainer mengaplikasikan konsep desain untuk membuat fondasi yang memadai. Dari berbagai konsep desain sangat membantu para perekayasa perangkat lunak. Konsep perangkat lunak mulai dari abstraksi, penyaringan, modularitas, arsitektur perangkat lunak, hierarki kontrol, partisi struktural, struktur data, prosedur perangkat lunak, dan penyembunyian informasi. Suatu desain perangkat lunak itu sendiri berupa proses dan data. pada dasarnya bisa disimpulkan bahwa proses desain itu tidak mudah dan tidak sederhana. Adapun faktor-faktor sukses dalam desain yang kompeten dipengaruhi

7. Jelaskan prinsip-prinsip desain perangkat lunak?

Tugas diskusi kelompok

Buatlah kelompok belajar dan diskusikan materi bab 3 konsep dan desain perangkat lunak. Buat juga Power point presentasi dan makalah, lakukan sharing dan tanya jawab dengan kelompok diskusi yang lain, yang di bimbingan oleh dosen.

DAFTAR PUSTAKA

Davis, S., 201 Principles of Software Development, Mc-Graw Hill, 1995.

Jackson, M. A., Principles of Programm Design, Academic Perss. 1975.

Kaiser S.H., The Design of Operating System for Small Computer Syste, Wiley-Interscience, 1983.

Meyer, B., Object-Oriented Software Construction, Printince - Hall, 1988.

Pernas, D.L., On Criteris To Be Used in Decomposing System into Modules, CACM, vol. 14, No. 4, 1972.

Pressman, Roger S. Software Engineering : A practitioner's Approach, Mc-Graw Hill, 1995.

Pressman, Roger S. Rekayasa Perangkat Lunak : Pendekatan Praktis Buku 1. Yogyakarta: Andi., 2002.

Wirth, N., Program Development by Stepwise Refinement, CACM, vol. 14, No. 4, 1971.



REKAYASA PERANGKAT LUNAK

BAB 4: ANALISIS KEBUTUHAN PENGGUNA (*USER REQUIREMENT ANALYSIS*)

Erly Krisnanik, S. Kom, M.M

Universitas Pembangunan Nasional Veteran Jakarta

BAB 4

ANALISIS KEBUTUHAN PENGGUNA (USER REQUIREMENT ANALYSIS)

A. PENDAHULUAN

Analisa kebutuhan pengguna merupakan tahapan yang dilakukan oleh seorang analis sistem berinteraksi dengan para pemakai sistem (end user), di mana tim pengembang sistem menunjukkan kepakaran mereka untuk mendapatkan data dan informasi serta kepercayaan pengguna sistem untuk mendapatkan partisipasi yang baik. Seorang analis ketika ingin mendapatkan kepercayaan pengguna sistem, harus memahami permasalahan yang ada untuk mendapatkan kesepakatan dalam pengembangan sebuah sistem informasi. Pada bab ini akan dijelaskan tahapan dalam menganalisis kebutuhan pengguna (*User Requirement Analysis*) terdiri dari : 1) Teknik Pengumpulan Data, 2) Studi Kelayakan, 3) Tujuan Studi Kelayakan, 4) Bentuk Kelayakan, dan 5) Dampak perkembangan Perangkat Lunak

ANALISA KEBUTUHAN PENGGUNA (USER REQUIREMENT ANALYSIS)

Setiap proyek perangkat lunak melewati fase yang disebut *Requirement Gathering*. Sebuah proyek yang sukses dimulai dengan serangkaian diskusi/pertemuan yang alot tentang apa yang harus dilakukan. Hal ini adalah tanggung jawab utama dari Tim Analis untuk dapat mengumpulkan data persyaratan dari klien. Mendapatkan data persyaratan yang benar dari klien sering menjadi salah satu rintangan terbesar dalam proyek perangkat lunak apa pun. Jika tim analis

DAFTAR PUSTAKA

- Adi Nugroho, *Rekayasa Perangkat Lunak Berorientasi Objek dengan Metode USDP*, Andi : Yogyakarta., 2012
- Feri Hari Utami dan Asnawati, *Rekayasa Perangkat Lunak*, Deepublish, 2015
- Hasanah, F. N., & Untari, R. S. (2021). Buku Ajar Rekayasa Perangkat Lunak. *Umsida Press*, 1-119. <https://doi.org/10.21070/2020/978-623-6833-89-6>
- Lila Setiyani, *Rekayasa Perangkat Lunak [Software Engineering]*, Jatayu Catra Internusa, 2019
- Pressman, Roger S. P.hD., *Rekayasa Perangkat Lunak*, Andi : Yogyakarta., 2012
- Rosa A.S dan M. Shalahuddin, *Rekayasa Perangkat Lunak untuk Terstruktur dan Berorientasi Objek*, Informatika Bandung, 2018
- Janner Simarmata, *REKAYASA PERANGKAT LUNAK*, Andi Publisher , 2010



REKAYASA PERANGKAT LUNAK

BAB 5: ANALISA DAN REKAYASA KEBUTUHAN PERANGKAT LUNAK

Yuniansyah, S.Kom., M.Kom

STMIK Palcomtech Palembang

BAB 5

ANALISA DAN REKAYASA KEBUTUHAN PERANGLAT LUNAK

A. PENDAHULUAN

Bagian ini akan membahas tentang Analisa dan rekayasa kebutuhan perangkat lunak. Bagian ini terdiri dari tiga bagian utama , yaitu Analisa, rekayasa dan perangkat lunak. Menurut Kamus Besar Bahasa Indonesia Analisa atau analisis berarti penyelidikan terhadap suatu masalah atau peristiwa untuk mengetahui keadaan yang sebenarnya. Analisis sangat dibutuhkan untuk mengetahui Tindakan yang akan dilakukan selanjutnya . Pada kebutuhan perangkat lunak, analisis sangat dibutuhkan untuk mengetahui peralatan atau perangkat keras yang akan digunakan serta sumber daya manusia yang akan menggunakannya. Bagian kedua adalah rekayasa. Rekayasa adalah penerapan ilmu dan teknologi yang digunakan untuk menyelesaikan suatu permasalahan atau pekerjaan. Rekayasa biasanya dilakukan oleh para ahli atau para praktisi di bidangnya.. Sedangkan rekayasa perangkat lunak adalah suatu pengembangan perangkat lunak yang meliputi berbagai tahapan seperti analisis, desain pengkodean, pengujian dan pemeliharaan, di mana semua tahapan ini dilakukan oleh para ahli. Bagian ketiga adalah perangkat lunak. Perangkat lunak atau dikenal dengan software adalah sekumpulan perintah atau sekumpulan data elektronik yang tersimpan dan di proses oleh computer. Perangkat lunak juga sebagai bagian dari system computer. Perangkat lunak terdiri dari beberapa jenis berdasarkan pengembangan dan fungsinya.

DAFTAR PUSTAKA

- Setiyani, Lila . 2018. *Rekayasa Perangkat Lunak Software Engineering*, Karawang, Jawa Barat, Jatayu Catra Internusa.
- Pressman, S, R. (2012). *Rekayasa Perangkat Lunak*. Andi. Yogyakarta.
- Rosa A.S, & M. Shalahuddin. 2014. *Rekayasa Perangkat Lunak*. Bandung: Informatika Bandung.



REKAYASA PERANGKAT LUNAK

BAB 6: SPESIFIKASI KEBUTUHAN PERANGKAT LUNAK

Yunita Ardilla S.Kom., M.M.T.

UIN Sunan Ampel Surabaya

BAB 6

SPEKIFIKASI KEBUTUHAN PERANGKAT LUNAK

A. PENDAHULUAN

Spesifikasi Kebutuhan Perangkat Lunak (SKPL) adalah deskripsi yang komprehensif mengenai tujuan dan lingkungan dari suatu perangkat lunak yang akan dikembangkan. Dalam dokumen spesifikasi kebutuhan perangkat lunak menjelaskan mengenai segala bentuk fitur yang akan dimiliki perangkat lunak dan bagaimana kinerja yang diharapkan nantinya. Dengan membuat dokumen Spesifikasi kebutuhan perangkat lunak diharapkan akan meminimalkan waktu dan upaya yang dibutuhkan oleh pengembang untuk mencapai tujuan yang diinginkan dan juga meminimalkan biaya pengembangan perangkat lunak (Demirel & Das, 2018).

Dokumen spesifikasi kebutuhan perangkat lunak yang baik akan mendefinisikan bagaimana aplikasi akan berinteraksi dengan perangkat keras sistem, program/aplikasi lain, dan pengguna/user dalam berbagai situasi di dunia nyata. Parameter seperti kecepatan operasi, waktu respon, portabilitas, maintenance, keamanan dan kecepatan pemulihan dari kejadian buruk akan dievaluasi.

Adapun komponen kunci atau inti dari spesifikasi kebutuhan perangkat lunak adalah:

- 1) *Business Drivers* : bagian ini akan menjelaskan mengenai alasan yang mendasari pelanggan ingin membangun sistem tersebut, termasuk

DAFTAR PUSTAKA

- Barcelos, L. V., & Penteado, R. D. (2017). Elaboration of software requirements documents by means of patterns instantiation. *Journal of Software Engineering Research and Development*, 5(1). <https://doi.org/10.1186/s40411-017-0038-9>
- Demirel, S. T., & Das, R. (2018). Software requirement analysis: Research challenges and technical approaches. *6th International Symposium on Digital Forensic and Security, ISDFS 2018 - Proceeding, 2018-Janua*, 1–6. <https://doi.org/10.1109/ISDFS.2018.8355322>
- Geogy, M., & Dharani, A. (2016). A Scrutiny of the Software Requirement Engineering Process. *Procedia Technology*, 25, 405–410. <https://doi.org/10.1016/j.protcy.2016.08.125>
- Kebutuhan, S., Lunak, P., Pembangunan, I. F., Lunak, P., Zulen, A. A., & Bachtiar, A. M. (2011). *Jurusan Teknik Informatika - Institut Teknologi Bandung GL01-ATM*.
- Marasco, J. (2007). Software requirements. In *Dr. Dobb's Journal* (Vol. 32, Issue 9). <https://doi.org/10.1201/b12149-5>



REKAYASA PERANGKAT LUNAK

BAB 7: PEMODELAN ANALISIS (DATA FLOW DIAGRAM DAN DATA DICTIONARY)

Iin Ernawati, S.Kom, M.Si

UPN Veteran Jakarta

BAB 7

PEMODELAN ANALISIS (DATA FLOW DIAGRAM DAN DATA DICTIONARY)

A. PENDAHULUAN

Pemodelan perangkat lunak selain berguna untuk melakukan dokumentasi perangkat lunak juga berguna untuk memudahkan komunikasi dan koordinasi dengan pihak-pihak yang terlibat dalam pengembangan perangkat lunak. Pihak-pihak tersebut adalah *stakeholder* (pengguna langsung), klien maupun tim internal developer itu sendiri mulai dari analis, desainer, *Programmer* dan tester.

Dalam bab ini selanjutnya akan dibahas beberapa hal terkait pemodelan analisis, yaitu :

1. Pemodelan (konsep dasar dan pengertian)
2. Pentingnya pemodelan
3. Bentuk pemodelan (model vs diagram)
4. Pemodelan perangkat lunak (kualitas dalam pemodelan)
5. Pemodelan Kebutuhan (terstruktur dan orientasi objek)
6. Data dictionary

B. PEMODELAN (KONSEP DASAR DAN PENGERTIAN)

Model merupakan representasi abstrak perangkat lunak yang akan dibuat oleh tim developer, yang tentunya representasi ini bertujuan memudahkan pemahaman dari semua stakeholder yang terlibat di dalam pengembangan perangkat lunak. Dari keseluruhan fase pengembangan

DAFTAR PUSTAKA

Ian Sommerville, Engineering Software Products An Introduction to Modern Software Engineering, © 2020 Pearson Education, Inc.
Roger S. Pressman. Software Engineering A Practitioner's Approach - Seventh Ed. Copyright © 2010 by The McGraw-Hill Companies, Inc.
Edward Ali. Rekayasa Perangkat Lunak. CV MFA – 2019.
www.tutorialpoints.com



REKAYASA PERANGKAT LUNAK

BAB 8: METODE STRUCTURED SYSTEM ANALYSIS AND DESIGN

Guntoro, S.T., M.Kom

Universitas Lancang Kuning

BAB 8

METODE STRUCTURED SYSTEM ANALYSIS AND DESIGN

A. PENDAHULUAN

Sebelum membahas tentang Metode *Structured System Analysis and Design*, kita akan membahas terkait dengan Teknik *Structured Analysis and Design* (SADT). Teknik *Structured Analysis and Design* (SADT) merupakan rekayasa sistem dan metodologi rekayasa perangkat lunak. Metode ini dikembangkan pada akhir 1960-an oleh Douglas T. Ross dan diresmikan sebagai IDEF0 pada tahun 1981. Metode *Structured Analysis and Design* (SADT) dikembangkan sebagai akibat munculnya pemrograman terstruktur pada tahun 1970-an. Pendekatan pengembangan yang sistematis ini selalu dikembangkan hingga dapat digunakan dalam kehidupan sehari-hari. Teknik terstruktur adalah cara formal untuk memecah tantangan dalam kegiatan bisnis menjadi bagian-bagian kecil yang dapat dikelompokkan dan terkait sebelum dipasang kembali menjadi satu kesatuan yang dapat digunakan untuk mengatasi masalah.

Desain terstruktur adalah proses mengubah temuan penyelidikan menjadi rencana yang bisa diterapkan. Metode terstruktur dilengkapi dengan alat dan prosedur yang diperlukan untuk pengembangan sistem, sehingga menghasilkan sistem dengan struktur yang terdefinisi dengan baik dan jelas sebagai produk akhir. Tantangan kompleks di perusahaan dapat diselesaikan dengan menggunakan metode terstruktur, dan hasilnya mudah digunakan, fleksibel, lebih ramah pengguna, memiliki dokumentasi

DAFTAR PUSTAKA

- D. Marca, C. McGowan, Structured Analysis and Design Technique, McGraw-Hill, 1987
- D. T. Ross: Structured Analysis (SA): A Language for Communicating Ideas. IEEE Transactions on Software Engineering, SE-3(1), pp. 16-34
- Ross, Douglas T., and Kenneth E. Schoman Jr. "Structured analysis for requirements definition." Software Engineering, IEEE Transactions on 1 (1977): 6-15.



REKAYASA PERANGKAT LUNAK

BAB 9: METODE PERANCANGAN BERBASIS OBJECT (OOAD)

Kraugusteeliana, M.Kom, M.M

Universitas Pembangunan Nasional Veteran Jakarta

BAB 9

METODE PERANCANGAN BERBASIS OBJECT (OOAD)

A. PENDAHULUAN

Pendekatan berorientasi objek pada buku ini akan membahas bagaimana menganalisa dan mendesain atas kebutuhan sistem berdasarkan hasil pemeriksaan (*requirement*) terhadap syarat atau kebutuhan sistem yang akan dibangun atau dikembangkan. Konsep OOAD menjelaskan kebutuhan actor, objek, kelas dalam lingkup sistem yang akan dibangun maupun dianalisa. Saat ini, analisis desain yang paling intuitif dan efektif mengadopsi pola pikir berorientasi objek yaitu pendekatan Orientasi Objek. Kegiatan analisis dan desain terjadi relatif lebih awal dalam proses perangkat lunak di mana "objek" sering terlihat dari semua elemen masalah. Keindahan paradigma berorientasi objek ini karena sangat elastis atau fleksibel. Pengembang dapat menggambarkan objek yang mudah dipahami oleh pelaku bisnis di tingkat konseptual, dan mudah untuk melalui proses iterasi dan elaborasi, perbaiki objek-objek itu menjadi tingkat abstraksi yang jauh lebih mudah untuk pengembang/designer teknis dapat menggunakan secara efektif. Penggunaan UML sebagai bentuk notasinya, dimulai dengan proses analisis yang baik di awal tahapan yang dilakukan. UML merupakan tools atau alat untuk memahami masalah melalui proses elisitasi, elaborasi, dan representasi. UML menekankan pada penggunaan (*Use Case*) dan membantu untuk pemahaman masalah dan sebagai mekanisme yang dapat sangat membantu perencanaan proyek dengan baik.

DAFTAR PUSTAKA

- Adi Nugroho. (2010). Rekayasa perangkat lunak menggunakan UML dan Java. Penerbit andi . Yogyakartaarta
- Dennis, Wixom And Tegarden (2009). *Systems Analysis & Design An Object-Oriented Approach with UML*. 5 Edition . Wiley . Usa
- Liang Huang1 , Yucong Duan , et al (2016). Enhancing UML Class Diagram Abstraction with Knowledge Graph. Spring internasional publishing.
- Nazaruddin,dkk (2022) .Analisa Perancangan Sistem Informasi Berbasis Object. PT. Widina.Bandung
- Quatrani,Terry. 2002. *Visual Modeling With Rational Rose*, Addison Wesley. (Q).
- R.B., Kim, D.-K., Ghosh, S., Song, E. (2004). *A UML-based pattern specification technique*. IEEE Trans. Softw. Eng. 30(3), 193–206
- Raul Sidnei Wazlawick,. 2014. Object-Oriented Analysis and Design for Information Systems Modeling with UML, OCL, and IFML. Elsevier
- Simon Bennet, Steve McRobb and Ray Farmer, Object Oriented Systems Analysis and Design Using UML, Edisi 3. ; McGraw Hill, 2006
- Storrie, H. (2011) *On the impact of layout quality to understanding UML diagrams*. In: 2011 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC), pp. 135–142. IEEE (2011)
- Tom, B.D.M.: Techniques of event history modeling: new approaches to causal analysis. Mahwah N. J. Lawrence Erlbaum Assoc. 52(2), 236–238 (1995)
- Wendy Boggs, Michael Boggs,Mastering UML with Rational Rose 2002, Sybex Inc
- Yuni Widiastiwi , dkk. (2020). *Pengantar teknologi informasi*. Widina. Bandung



REKAYASA PERANGKAT LUNAK

BAB 10: METODE PEMODELAN UNIFIED MODELING LANGUAGE

I Wayan Widi Pradnyana, S.Kom, MTI

Universitas Pembangunan Nasional Veteran Jakarta

BAB 10

METODE PEMODELAN UNIFIED MODELING LANGUAGE

A. PENDAHULUAN

Penggunaan teknologi pemodelan perangkat lunak adalah sangat dibutuhkan dalam rangkaian pengelolaan sistem informasi pada suatu organisasi. Penggunaan model dapat mempermudah pemahaman dan kesepakatan pemangku kepentingan dan pelaksana mengenai rancangan produk yang akan dibuat dan diterima. Tata bahasa atau notasi *Unified Modeling Language* (UML) sebagai aturan pemodelan yang disepakati umum dapat mengurangi kesalahpahaman pemahaman rancangan, dan menjembatani beberapa tahapan pengembangan perangkat lunak yang diterapkan. Pada bagian ini akan dibahas kebutuhan, konsep, elemen, dan cara penggunaan UML sebagai sarana pendukung dalam pengembangan perangkat lunak.

B. MENGENAI UNIFIED MODELING LANGUAGE (UML)

1. Sejarah UML

Dalam pembahasan yang diberikan oleh (Unhelkar,2017) menyebutkan mengenai sejarah UML. Pemodelan perangkat lunak diawali dengan penggunaan flowchartm dilengkapi dengan pola model berbasis MVC sebagai basis struktur program. Dikuti dengan metode SSADM (*structured system analysis and design method*) yang berfokus pada analysis, namun tetap menggunakan pendekatan prosedural. Pada 1980 mulai dikenali pemodelan data pada ERD (*Entity Relationship Diagram*)

DAFTAR PUSTAKA

- Boehm, B., Lane, J. A., Koolmanojwong, S., & Turner, R. (2014). *The Incremental Commitment Spiral Model: Principles and Practices for Successful Systems and Software*. Addison-Wesley Professional.
- Booch. (1994). *Object Oriented Analysis and Design with Applications, 3e*. Benjamin/Cummings Publishing Company.
- Chonoles, M. J. (2017). *OCUP 2 Certification Guide: Preparing for the OMG Certified UML 2.5 Professional 2 Foundation Exam*. Elsevier Science.
- Gomaa, H. (2011). *Software Modeling and Design: UML, Use Cases, Patterns, and Software Architectures*. Cambridge University Press.
- Jacobson, I. (1993). *Object-oriented Software Engineering: A Use Case Driven Approach*. Pearson Education.
- Jacobson, I., & Bylund, S. (2000). *The Road to the Unified Software Development Process*. Cambridge University Press.
- OMG MOF 2.5.1. (2016). <https://www.omg.org/spec/MOF/2.5.1/>
- OMG UML 2.0. (2005). <https://www.omg.org/spec/UML/2.0/>
- OMG UML 2.5.1. (2017). <https://www.omg.org/spec/UML/2.5.1/>
- Papyrus Documentation. (2022). <https://www.eclipse.org/papyrus/documentation.html>
- Rumbaugh, J., Rumbaugh, J., Blaha, S., Blaha, M., Blaha, M. R., Eddy, F., Premerlani, W., Rumbaugh, J. R., Lorensen, W., & Lorensen, B. (1991). *Object-oriented Modeling and Design*. Prentice Hall.
- Unhelkar, B. (2017). *Software Engineering with UML*. CRC Press.



REKAYASA PERANGKAT LUNAK

BAB 11: ANTARMUKA PENGGUNA (*USER INTERFACE*)

Titus Kristanto, S.Kom., M.Kom

Institut Teknologi Telkom Surabaya

BAB 11

ANTARMUKA PENGGUNA (*USER INTERFACE*)

A. PENDAHULUAN

Desain antarmuka merupakan hal terpenting pada kunci keberhasilan dalam pengembangan perangkat lunak (Mustafa, 2014). Masing-masing perangkat lunak dikembangkan oleh developer yang mempunyai keahlian tersendiri dalam membuat desain antarmuka, sesuai dengan kebutuhan user (pelanggan). Antarmuka merupakan bagian dari computer yang dapat didengar, disentuh, dilihat, dan diajak bicara secara langsung maupun tidak langsung. Tujuan desain adalah sederhana sehingga dalam bekerja pada komputer membuat lebih mudah, nyaman, dan menyenangkan.

Antarmuka yang baik adalah membuat pengguna lebih konsentrasi pada tugas yang dikerjakan, dan bukan pada konsentrasi mekanisme yang digunakan dalam menampilkan informasi dan mengerjakan tugas (Galitz, 2007). Tujuan utama desain antarmuka adalah menghasilkan sistem komputer yang dapat digunakan dengan baik oleh pengguna (*good usability*). Selain desain antarmuka, karakteristik manusia sangat berpengaruh pada interaksi manusia dan komputer.

B. DESAIN INTERAKSI (*INTERACTION DESIGN*)

Desain Interaksi adalah merancang atau mendesain produk-produk interaktif untuk mendukung cara orang berkomunikasi dan berinteraksi dalam kehidupan pekerjaan dan kehidupan sehari-hari (Preece et al.,

DAFTAR PUSTAKA

- Adrian, M. (2013). *Konsep Desain Interaksi*. Politeknik Telkom.
- Basuki, A. (2020). *Graphic User Interfaces (GUI)*. Politeknik Elektronika Negeri Surabaya (PENS).
- Defi Poppyna Putri. (2019). Analisis User Experience dan Usability Terhadap Sistem Pembelajaran Berbasis Teknologi Informasi Menggunakan Usability Testing. In *Universitas Islam Negeri Sultan Syarif Kasim Riau*.
- Galitz, W. O. (2007). The Essential Guide to User Interface Design : An Introduction to GUI Design Principles and Techniques (Thrid Edition). In *Nutrition & Food Science* (Vol. 40, Issue 5). Wiley Publishing, Inc.
- Garett, J. J. (2011). The Elements of User Experience : User Centered Design for the Web and Beyond (Second Edition). In *New Rides2*. New Riders. <https://doi.org/10.1145/889692.889709>
- Ghiffary, M. N. El, Susanto, T. D., & Herdiyanti, A. (2018). Analisis Komponen Desain Layout, Warna, dan Kontrol Pada Antarmuka Pengguna Aplikasi Mobile Berdasarkan Kemudahan Penggunaan (Studi Kasus: Aplikasi Olride). *Jurnal Teknik ITS*, 7(1), A143-A 148.
- Griffin, B., & Baston, L. (2014). *Interfaces*. Girton Colege University of Cambridge.
- Hasibuan, D. H. (2021). *Perancangan Desain Interaksi Website Project Collaboration*. Universitas Komputer Indonesia.
- International Standard Organization. (2010). *ISO 9241-210*. <https://doi.org/10.5100/jje.30.1>
- Mayhew, D. J. (1992). *Principles and Guidelines in Software User Interface Design*. Prentice Hall PTR.
- Mustafa, F. (2014). *Rancang Bangun Antarmuka Aplikasi Pepak Bahasa Jawa Berbasis Android*. Sekolah Tinggi Manajemen Informatika dan Teknik Komputer (STIKOM) Surabaya.

- Preece, J., Sharp, H., & Rogers, Y. (2015). *Interaction Design: Beyond Human-Computer Interaction*. John Wiley & Sons, Inc.
- Pressman, R. S. (2010). *Software Engineering : A Practitioner's Approach*. McGraw-Hill.
<https://doi.org/10.1145/336512.336521>
- Proboyekti, U. (2010). Pengantar Desain Interaksi. In *Pengantar Desain Interaksi*.
- Ramadhan, H. (2015). *BAB I CLI (Command Line Interface)*.
- Santi, R. C. N., & Fitriyah, A. (2016). Perancangan Interaksi Pengguna (User Interaction Design) Menggunakan Metode Prototyping. *Jurnal Teknik Informatika*, 9(2), 108–113.
- Santoso, H. B. (2018). *Desain Interaksi*. Universitas Indonesia.
- Schlatte, T., & Levinson, D. (2013). *Visual Usability: Principles and Practices for Designing Digital Applications*.
- Soegaard, M. (2002). The Basics of User Experience Design. In *Interaction Design Foundation*.
- Sukanto, T. S. (2014). *Interaksi Manusia & Komputer*. Universitas Dian Nuswantoro.
- Syahdatina, R., Tolle, H., & Az-zahra, H. M. (2020). Perancangan Desain Interaksi Aplikasi Malang Sehat Modul Pendataan dan Monitoring Kesehatan Masyarakat Kota Malang dengan menggunakan Metode Human-Centered Design. *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer (JPTIIK)*, 4(8), 2349–2357.
- Winograd, T., & Woods, D. D. (1997). *The Challenge of Human Centered Design*.
- Wiryawan, M. B. (2011). User Experience (UX) sebagai Bagian dari Pemikiran Desain dalam Pendidikan Tinggi Desain Komunikasi Visual. *Humaniora*, 2(2), 1158.
- Zhang, J., & Chen, C. H. (2007). Moving Objects Detection and Segmentation In Dynamic Video Backgrounds. *2007 IEEE Conference on Technologies for Homeland Security: Enhancing*

Critical Infrastructure Dependability,
<https://doi.org/10.1109/THS.2007.370021>

64–69.



REKAYASA PERANGKAT LUNAK

BAB 12: METODE PENGUJIAN PERANGKAT LUNAK

D Tri Octafian, M.Kom.

Institut Teknologi dan Bisnis PalComTech

BAB 12

METODE PENGUJIAN PERANGKAT LUNAK

A. PENDAHULUAN

Pengujian perangkat lunak adalah tahapan yang biasa ada pada sebuah metode pengembangan perangkat lunak (*Software Development Life Cycle Method*). Tahapan pengujian perangkat lunak hampir selalu tidak dibahas secara rinci dalam mata kuliah atau buku yang membahas tentang rekayasa perangkat lunak. Padahal pengujian perangkat lunak tidak kalah pentingnya berperan dalam menentukan keberhasilan pengembangan perangkat lunak. Sebagian besar literatur buku rekayasa perangkat lunak akan membahas secara rinci pada tahapan analisis dan desain perangkat lunak dengan pendekatan paradigma tertentu beserta tools-nya. Pada kenyataannya, pengujian perangkat lunak tidak bisa dilakukan oleh sembarangan orang dan membutuhkan kemampuan dan pengetahuan di bidang tersebut. Bahkan pengujian perangkat lunak menciptakan lapangan pekerjaan tersendiri. Pada pengujian perangkat lunak terdapat metode yang mana memiliki tahapan dan serangkaian aktivitas yang biasa dilakukan oleh tim penguji. Tahapan pengujian perangkat lunak seperti layaknya pengembangan perangkat lunak, secara umum dapat dikelompokkan ke dalam beberapa tahapan, yaitu: perencanaan, analisis, desain, implementasi dan evaluasi. Tahapan-tahapan ini membentuk siklus pengujian dengan kata lain dilakukan dengan iterasi tertentu sampai sebuah perangkat lunak dinyatakan lulus tahapan pengujian oleh tim penguji.

DAFTAR PUSTAKA

- Admin WBC. (2019). *Software Development Life Cycle (SDLC)*. <https://bigwater.consulting/2019/04/08/software-development-life-cycle-sdlc/>
- Aji, D. M. (2018). *Smoke Testing*. <https://medium.com/@damar.mustikoaji/smoke-testing-f841f9632b84>
- Arviana, G. N. (2021). *Mengenal Teknik White Box Testing untuk Menguji Struktur Software*. <https://glints.com/id/lowongan/white-box-testing-adalah/#.Yl7a4NpBzIU>
- Bose, S. (2021). *Functional Testing: A Detailed Guide*. <https://www.browserstack.com/guide/functional-testing>
- Hamilton, T. (2022a). *Difference Between Verification and Validation with Example*. <https://www.guru99.com/verification-v-s-validation-in-a-software-testing.html>
- Hamilton, T. (2022b). *Levels of Testing in Software Testing*. <https://www.guru99.com/levels-of-testing.html>
- Hamilton, T. (2022c). *What is Test Scenario? Template with Examples*. <https://www.guru99.com/test-scenario.html>
- Imperva. (n.d.). *White Box Testing*. Retrieved April 28, 2022, from <https://www.imperva.com/learn/application-security/white-box-testing/>
- katalon.com. (n.d.). *What is Regression Testing? Definition, Tools & How to Get Started*. Retrieved April 19, 2022, from <https://katalon.com/resources-center/blog/regression-testing>
- Kinsbruner, E. (2020). *What Is Non Functional Testing?* <https://www.perfecto.io/blog/what-is-non-functional-testing>
- McMullin, W. (2021). *How to Write Test Cases for Software: Examples & Tutorial*. <https://www.parasoft.com/blog/how-to-write-test-cases-for-software-examples-tutorial/>
- Olsen, K., Posthuma, M., & Ulrich, S. (2019). *Certified Tester Foundation Level Syllabus* (2018 V3.1; p. 93). International Software Testing Qualifications Board.

- ReqTest. (n.d.). *Black Box Testing – Understanding the Basics*. Retrieved April 26, 2022, from <https://reqtest.com/testing-blog/black-box-testing/>
- Sari, M. N. A. (2021). *Software Testing, Definisi dan Jenis-jenisnya*. <https://www.gamelab.id/news/667-software-testing-definisi-dan-jenis-jenisnya>
- Setiawan, R. (2021). *Black Box Testing Untuk Menguji Perangkat Lunak*. <https://www.dicoding.com/blog/black-box-testing/>
- Shobika. (2016). *Fundamental Test Process*. <https://www.egrovesys.com/software-testing/>
- softwaretestinghelp.com. (2022). *Types Of Software Testing: Different Testing Types With Details*. <https://www.softwaretestinghelp.com/types-of-software-testing/>
- Suhartono, J. (2016). *Software Testing*. <https://sis.binus.ac.id/2016/12/16/software-testing/>
- Susanto, E. (2015). *Static dan Dynamic Testing*. <https://www.kompasiana.com/edysusanto74/55ed4d094ef9fd57102b1fd5/static-dynamic-testing>
- Tutorialspoint. (n.d.). *Test Execution*. Retrieved March 27, 2022, from https://www.tutorialspoint.com/software_testing_dictionary/test_execution.htm



REKAYASA PERANGKAT LUNAK

BAB 13: METODE PEMELIHARAAN PERANGKAT LUNAK

Dr. Irwanto, S.Pd.T., M.T.

Universitas Sultan Ageng Tirtayasa

BAB 13

METODE PEMELIHARAAN PERANGKAT LUNAK

A. PENDAHULUAN

Pemeliharaan Software adalah proses umum perubahan atau pengembangan perangkat lunak setelah diserahkan ke konsumen. Perubahan mungkin berupa perubahan sederhana untuk membetulkan *error* coding atau perubahan yang lebih ekstensif untuk membetulkan error perancangan/perbaikan signifikan untuk membetulkan error spesifikasi atau akomodasi persyaratan baru. Pemeliharaan ini merupakan siklus terakhir dari SDLC yaitu dengan pemeriksaan periodik, audit dan permintaan pengguna akan menjadi source untuk melakukan perawatan system di seluruh masa hidup sistem. Istilah pemeliharaan perangkat lunak digunakan untuk menjabarkan aktivitas dari analisis sistem (*software engineering*) yang terjadi pada saat hasil produk perangkat lunak sudah dipergunakan oleh pemakai (*user*) (Sulistyoning, 2017). Biasanya pengembangan produk perangkat lunak memerlukan waktu antara 1 sampai dengan 2 tahun, tetapi pada fase pemeliharaan perangkat lunak menghabiskan 5 sampai dengan 10 tahun.

Aktivitas yang terjadi pada *fase* pemeliharaan antara lain: (1) Penambahan atau peningkatan atau juga perbaikan untuk produk perangkat lunak, (2) Adaptasi produk dengan lingkungan mesin yang baru, dan (3) Pembetulan permasalahan yang timbul. Pemeliharaan perangkat lunak adalah proses umum perubahan atau pengembangan perangkat lunak setelah diserahkan ke konsumen. Perubahan dapat berupa

DAFTAR PUSTAKA

- Anonim, 2013. *Analisis Terstruktur*. <http://soft-to-gine.blogspot.com/2011/09/analisis-terstruktur.html>. Diakses 30 Januari 2013.
- Darwiyanti, Sri dan Wahono Satrio Romi. Pengantar *Unified Modeling language*. <http://setia.staff.gunadarma.ac.id/Downloads/files/6039/MateriSuplemenUml.pdf>. Diakses 5 Februari 2013.
- Dekha Sundhawati. Tanpa Tahun. *KONSEP & TEKNIK PEMELIHARAAN PERANGKAT LUNAK*. JURUSAN SISTEM INFORMASI FAKULTAS ILMU KOMPUTER UNIVERSITAS MERCU BUANA.
- Fitria Nur Hasanah & Rahmania Sri Untari. 2020. *BUKU AJAR REKAYASA PERANGKAT LUNAK*. UNIVERSITAS MUHAMMADIYAH SIDOARJO. Diterbitkan oleh UMSIDA PRESS.
- Jogiyanto, H. 1990. *Analisis dan Design Sistem Informasi: Pendekatan Terstruktur Teori dan Praktek Aplikasi Bisnis*. Yogyakarta : ANDI Publisher.
- Kendall & Kendall. 2006. *Systems Analysis and Design, 6th Edition*. Prentice Hall.
- Kendall, J.E. & Kendall, K.E. 2010. *Analisis dan Perancangan Sistem*. Jakarta: Indeks.
- Kendall, Kenneth E. and Kendall, Julie E. 2011. *System Analysis and Design 8th ed*, Prentice Hall, New Jersey.
- Marakas, G.M. 2006. *System Analysis Design: an Active Approach*. New York: Mc.Graw-Hill.
- Mc.,Leod, R. Jr. 2002. *System Development: A Project Management Approach*. New York: Leigh Publishing LLC.
- Naik, K. and Tripathy, P. 2008. *Software Testing and Quality Assurance Theory and Practice*, John Wiley & Sons, New Jersey.
- Nugroho P E, Ratnasar K, Ramadhani N K, dan Putro L B. 2013. *Rekayasa Perangkat Lunak*
http://courseware.politekniktelkom.ac.id/BUKU_MI/Semester%201/IS213%20

- Pressman, Rouger. S. 1997. *Software Enigneering, 4th Edition*. Mc. Graw Hill.
- Pressman, Rouger. S. 2002. *Rekayasa Perangkat Lunak*. Andi. Yogyakarta.
- RAHMAT JAENURI.Tanpa Tahun. *Konsep dan Teknik Pemeliharaan Perangkat Lunak Pemeliharaan Korektif, Adaptif, Perfektif, Prefentif*. JURUSAN SISTEM INFORMASI FAKULTAS ILMU KOMPUTER UNIVERSITAS MERCU BUANA.
- Ratna Wardani. 2012. *Rekayasa Perangkat Lunak*. JURUSAN PENDIDIKAN TEKNIK ELEKTRONIKA. UNIVERSITAS NEGERI YOGYAKARTA. Program Pendidikan dan Pelatihan Profesi Guru Teknologi Informasi dan Komunikasi.
- Rekayasa%20Perangkat%20Lunak/Rekayasa%20Perangkat%20Lunak.pdf. Diakses 5 Februari 2021.
- Sommerville, Ian. 2004. *Software Engineering, 7th Edition*. Addison Wesley.
- Sulistiyoning. 2016. *Modul Praktikum Rekayasa Perangkat Lunak*. LABORATORIUM MULTIMEDIA DESIGN KEAHLIAN PROGRAMMING & INTERACTIVE MULTIMEDIA FAKULTAS ILMU TERAPAN UNIVERSITAS TELKOM.
- Verdi Yasin, *Rekayasa Perangkat Lunak Berorientasi Objek*, Mitra Wacana Media, Jakarta 2022.



REKAYASA PERANGKAT LUNAK

BAB 14: PENJAMINAN MUTU PERANGKAT LUNAK

Iin Ernawati, S.Kom., M.Si

UPN Veteran Jakarta

BAB 14

PENJAMINAN MUTU PERANGKAT LUNAK

A. PENDAHULUAN

Pada tahun 2005 seorang peneliti bernama Humphrey, mengumpulkan data bertahun-tahun dari ribuan insinyur perangkat lunak yang menunjukkan bahwa mereka secara tidak sengaja menyuntikkan 100 cacat perseribu baris kode. Humphrey juga menunjukkan bahwa perangkat lunak komersial biasanya mencakup satu sampai sepuluh kesalahan per seribu baris kode. Kesalahan ini seperti bom waktu tersembunyi yang akan meledak ketika kondisi tertentu terpenuhi. Oleh karena itu, perlu diterapkan praktik untuk mengidentifikasi dan memperbaiki kesalahan ini pada setiap tahap siklus pengembangan dan pemeliharaan.

B. REVIEW (ULASAN)

Salah satu teknik pendeteksian (verifikasi atau evaluasi) adalah dengan melakukan review. Teknik lain adalah dengan melakukan tes (pengujian), namun perlu diingat bahwa kualitas produk perangkat lunak dimulai pada tahap pertama dari proses pengembangan, yaitu ketika mendefinisikan persyaratan dan spesifikasi. Tinjauan akan mendeteksi dan memperbaiki kesalahan pada tahap awal pengembangan, sementara pengujian hanya akan digunakan jika *source code* tersedia. Jadi kita tidak perlu menunggu tahap pengujian untuk mulai mencari kesalahan. Selain itu, jauh lebih mudah untuk mendeteksi kesalahan dengan ulasan daripada dengan pengujian. Ini tidak berarti kita harus mengabaikan pengujian karena penting untuk mendeteksi kesalahan yang tidak dapat ditemukan

TUGAS DAN EVALUASI

1. Apa saja standar internasional yang digunakan dalam mengukur kualitas perangkat lunak?
2. Apa tujuan dari Standar IEEE 1028-2008 dalam penjaminan mutu perangkat lunak?
3. Apa tujuan dari Standar IEEE 730 dalam penjaminan mutu perangkat lunak?
4. Apa saja alasan dilaksanakannya proses *walk-through*?
5. Apa tujuan dari *walk-through* dalam mengevaluasi produk perangkat lunak?

DAFTAR PUSTAKA

- Pressman, R.S. (2015). *Software Engineering : A Practioner's Approach*. 8th ed. McGraww-Hill Companies. Inc, Americas, New York.
- Sommerville, I. (2018). *Software Engineering*. Hallbergmoos/Germany: Pearson.



PROFIL PENULIS

Widiyawati, S.Kom, M.Kom



Penulis Lahir di Jakarta 14 Juni ,menempuh pendidikan di SD Negeri Kebun Bawang 05 pagi Jakarta, SMPN 221 Jakarta, dan SMA Muhammadiyah 2 Jakarta, melanjutkan studi S1 di STMIK GUNADARMA lulus tahun 1995, kemudian melanjutkan kembali jenjang pendidikan S2 di STMIK ERESHA lulus tahun 2014 , penulis bekerja menjadi Dosen tetap di STMIK Bani Saleh bekasi jawa barat Sebagai dosen di prodik Manajemen Informatika dari tahun 2003 sampai saat ini

Nazaruddin Ahmad, M.T



Penulis Berasal dari kota Banda Aceh, Aceh, lahir di Banda Aceh 5 Juni 1982, merupakan anak keempat dari lima bersaudara dari pasangan Drs. H. Ahmad Habib Lubis dan Hj. Rostina. Saat ini penulis bekerja sebagai dosen tetap pada Prodi Teknologi Informasi, Fakultas Sains dan Teknologi, Universitas Islam Negeri Ar-Raniry Banda Aceh sejak tahun 2014. Penulis menyelesaikan pendidikan Strata Satu (S1) Teknik Informatika di Universitas Jabal Ghafur Sigli – Aceh Tahun 2008, kemudian melanjutkan pendidikan Strata Dua (S2) di Universitas Atma Jaya Yogyakarta pada program studi Magister Teknik Informatika dan mendapatkan gelar Magister Teknik pada tahun 2013. Penulis saat ini mengajar mata kuliah Basis Data, Manajemen Proyek Teknologi Informasi, Jaringan Syaraf Tiruan dan Data Mining. Penulis juga aktif dalam menulis buku *Bookchapter* yang berkolaborasi dengan dosen-dosen dari seluruh Indonesia dengan harapan dapat memberikan kontribusi untuk pengembangan ilmu dalam bidang Informatika.

Eka Hartati, M.Kom



Penulis lahir di Mendala, 26 November 1990, Putri dari (Alm) Syamiri dan Ibunda Azimah. Menyelesaikan pendidikan strata 1 Program Studi Sistem Informasi di STMIK PalComTech Palembang pada tahun 2014. Penulis kemudian melanjutkan dan menyelesaikan Pendidikan Strata 2 Program Studi Teknik Informatika di Universitas Bina Darma Palembang pada tahun 2016. Sejak tahun 2016 sampai saat ini penulis masih aktif mengajar. Penulis merupakan dosen tetap pada program studi Sistem Informasi di STMIK PalComTech Palembang sejak tahun 2016 sampai 19 April 2022. Fokus bidang keahlian penulis utamanya dibidang sistem informasi.

Erly Krisnanik, S. Kom, M.M



Penulis Lahir di Jakarta pada tanggal 8 September 1974. Pendidikan Sarjana Komputer di tempuh di Universitas Pembangunan Nasional Veteran Jakarta, pada tahun 1993 saat itu masih berstatus kedinasan lulus tahun 1997. Kemudian melanjutkan kuliah S2 Magister Manajemen bidang Sistem Informasi Manajemen di Universitas yang sama pada tahun 2006. Lulus Sarjana langsung bekerja di Universitas Pembangunan Nasional Veteran Jakarta sebagai Asisten Laboratorium. Pengalaman mengajar pada tahun 1998 ditawarkan magang menjadi dosen untuk mengampu mata kuliah Perangkat Lunak Aplikasi dan Pengantar Komputer. Kemudian diberi kepercayaan untuk mengajar matakuliah secara mandiri mulai tahun 2000 untuk matakuliah Perangkat Lunak Aplikasi, Aplikasi Berbasis Web, Sistem Informasi Manajemen, Manajemen Proyek dan Analisa dan Perancangan Sistem Informasi. Karya yang pernah diterbitkan adalah publikasi di jurnal internasional terindex Scopus dan nasional terakreditasi dan prosiding terindex scopus, google scholar dan WoS serta menulis book chapter di penerbit Widina dan Media Sains Indonesia. Email Penulis: erlykrisnanik@gmail.com

Yuniansyah, S.Kom., M.Kom



Penulis mengenal Ilmu Komputer Pada Tahun 1995 dimana penulis mendapatkan kesempatan melanjutkan pendidikan Strata satu di Universitas Bina Darma Palembang dan selesai pada tahun 1999. Pada saat kuliah juga selama 2 tahun (1997-1999) sempat menjadi Asisten Dosen di Laboratorium komputer Universitas Bina Darma untuk beberapa matakuliah pemrograman. Pada awal tahun 2002 melanjutkan pendidikan di Program Studi Magister Ilmu Komputer Fakultas Ilmu Matematika dan Ilmu Pengetahuan Alam (F-MIPA) Universitas Gadjah Mada – Yogyakarta dan selesai pada akhir tahun 2013. Selama di Yogyakarta juga penulis berkesempatan menjadi Dosen di salah satu Perguruan Tinggi yang ada di Yogyakarta. Selepas meraih gelar Magister Komputer penulis menjadi dosen di beberapa Perguruan Tinggi swasta dan Universitas Negeri di Kota Palembang. Penulis juga pernah menjadi Dosen di Kota Lubuk Linggau, Batam, dan Duri. Saat ini penulis menjadi dosen praktisi di salah satu perguruan tinggi ternama di Kota Palembang Penulis ini dapat dihubungi melalui email: yuniansyah.mr@gmail.com serta dan dapat juga melalui WA / Telegram : 0812 3516 8181

Yunita Ardilla S.Kom., M.MT



Penulis lahir di kota Sidoarjo pada tanggal 04 Juni 1992. Riwayat pendidikannya dimulai sejak lulus dari program sarjana Teknik Informatika pada tahun 2014 dari Institut Teknologi Sepuluh Nopember, yang kemudian dilanjutkan sampai jenjang pasca sarjana (S2) Manajemen Teknologi Informasi di Institut Teknologi Sepuluh Nopember. Ia juga menjadi dosen tetap (*faculty member*) Fakultas Dakwah dan Komunikasi di Universitas Islam Negeri Sunan Ampel Surabaya sebagai dosen sistem informasi manajemen. Informasi lengkap dapat dilihat di web pribadi www.yunita.net | surel: yunita.ardilla@uinsby.ac.id

lin Ernawati, S.Kom, M.Si



Penulis lahir di Madiun – Jawa Timur, pada tanggal 2 Januari 1976. Mengawali karir sebagai asisten laboratorium komputer pada Fakultas Ilmu Komputer – UPN Veteran Jakarta pada tahun 2001, khusus mata kuliah algoritma dan pemrograman, pemrograman C++, dan Basisdata. Sejak tahun 2001 hingga saat ini aktif sebagai dosen Fakultas Ilmu Komputer UPN Veteran Jakarta. Menyelesaikan pendidikan S1 Manajemen Informatika di UPN Veteran Jakarta pada tahun 2000, kemudian melanjutkan pendidikan ke jenjang (S2) Magister Ilmu Komputer di Departemen Ilmu Komputer – FMIPA IPB pada tahun 2008. Penelitian yang pernah dilakukan mayoritas berkaitan dengan bidang *Data Mining* dan *Machine Learning*, baik dilakukan secara mandiri maupun berkolaborasi dengan mahasiswa dan rekan sejawat. Pada tahun 2020 bergabung bersama beberapa rekan sejawat menulis sebuah bab terkait *Database Management System (DBMS)* dalam sebuah *book chapter* dengan judul “ Pengantar Teknologi Informasi “.

Guntoro, S.T., M.Kom



Penulis lahir di Bangkinang, 12 Januari 1988. Lulus S1 di Jurusan Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Sultan Syarif Kasim Riau tahun 2011. Lulus S2 di Program Studi Ilmu Komputer Fakultas Matematika dan Ilmu Pengetahuan Alam (FMIPA) IPB University tahun 2015. Saat ini adalah Dosen tetap di Program Studi Teknik Informatika Fakultas Ilmu Komputer, Universitas Lancang Kuning. Mengampu mata kuliah Forensika Digital, Sistem Operasi, Teknik Kompilasi, Metode Numerik dan Manajemen Proyek Perangkat Lunak. Saat ini diamanahkan sebagai Kepala Bagian Penelitian LPPM Universitas Lancang Kuning. Sebagai Editor in Chief Jurnal Dinamisia: Jurnal Pengabdian Kepada Masyarakat, Jurnal JITACS: Journal of Information Technology and Computer Science Universitas Lancang Kuning. Pernah menulis buku Pengantar Teknologi Informasi, Sistem

Informasi Manajemen, Data Mining dan Aplikasinya, buku Praktis Desain Dokumen dan Informasi dengan Canva Android : Membuat Dokumen dan Informasi Berbasis Digital dan Cetak. Ia bisa ditemua di www.guntoro.web.id

Kraugusteeliana, M.Kom, M.M



Penulis menghabiskan masa sekolah SD – SMAN 2 Cilegon (dh/ SMAKS) melanjutkan kuliah S1 di Univ Budi Luhur, Melanjutkan S2 bidang software engineering di Benarif dan mengambil S2 Magister Manajemen kosentrasi Human Capital (SDM) di UBL. Mengajar sejak 1999 diberbagai PT di Jakarta tahun 2014 pindah ke UPN Veteran Jakarta sebagai dosen

tetap sistem informasi hingga saat ini. Pengalaman mengajar mengampu matakuliah struktur data dan algoritma, Pengantar Teknologi Informasi, IMK, Metode Perancangan Program, Analisa Sistem Informasi Management Sains, Knowledge Management, SIM, SPK, Analisa Proses Bisnis, Rekayasa SI, RPL, SBD, Tata Kelola Teknologi Informasi, Manajemen Resiko TI, Audit Sistem Informasi, Sistem Enterprise, CRM, SCM dan Manajemen layanan TI. Adapun buku atau Book Chapter yang telah diterbitkan dibidang diantaranya PTI, SIM, Analisa perancangan sistem. RPL, Pengantar Manajemen SDM juga IMK. Focus penelitian pada eGovernance, LMS, SPK, SIM, eGovernment evaluasi atau Audit system Cobit, ITIIL, Webqual 4.0, UI/UX, HE VAL IT, IPA, UTAUT,TAM, ISO 27001, TOGAF 9.1, NIST SP 30-80 juga Octave Allegro dengan software olah data. Pengalaman dalam organisasi pernah menduduki posisi kabag perencanaan akademik dan statistik, auditor ISO 2015, Manager HR, auditor ISO 21001:2018, Plt. Dekan, Manager Humas dan Marketing, Sekretris LP3M, manager Quality Assurance, Wakil Dekan bidang Akademik, Ketua Komisi Senat, Komisi penjaminan mutu uji kompternsi Kepala Pusat Akreditasi Internasional dan Sertifikasi dan manager Kerjasama & IT pada LSP. Email Penulis: kraugusteeliana@upnvj.ac.id

I Wayan Widi Pradnyana, S.Kom, MTI



Penulis lahir di kota Denpasar, pada 25 Juni 1981. penulis menyelesaikan pendidikan dasar di Provinsi Bali. Pada tahun 1999 menempuh pendidikan Sarjana Teknik Informatika di Institut Teknologi Sepuluh Nopember Surabaya. Dilanjutkan dengan menempuh karir profesional sebagai konsultan teknologi informasi dan staf IT pada perusahaan di Jakarta hingga tahun 2014, sambil menempuh pendidikan Magister Teknologi Informasi di Fakultas Ilmu Komputer Universitas Indonesia pada 2009. Pada tahun 2015 berkarir sebagai pengajar di program Sarjana Sistem Informasi di Fakultas Ilmu Komputer Universitas Pembangunan Nasional Veteran Jakarta. Mata kuliah yang diajarkan dan penelitian yang dikerjakan terkait dengan topik Jaringan Komputer, Sistem Terdistribusi, Keamanan Informasi, Arsitektur Enterprise, dan Big Data.

Titus Kristanto, S.Kom., M.Kom



Penulis merupakan dosen tetap program studi Reayasa Perangkat Lunak (RPL) di Institut Teknologi Telkom Surabaya (ITTS). Penulis dilahirkan di Kota Surabaya tanggal 10 Februari 1986. Penulis awal sekolah di SDN Tandus Lor I/109 Surabaya, lalu di SMPN 33 Surabaya, dilanjutkan di SMA Ta'miriyah Surabaya. Pendidikan tinggi awal mula ditempuh pada Program Diploma 1 Sistem Informasi Universitas Airlangga, lanjut kuliah Diploma 3 Sistem Informasi Universitas Airlangga. Pendidikan S1 ditempuh pada Teknik Informatika Universitas 17 Agustus 1945 Surabaya dan dilanjutkan kuliah S2 di Teknik Informatika (Bidang Minat : Sistem Informasi) Institut Teknologi Sepuluh Nopember. Bidang keahlian penulis adalah Sistem Informasi dan Teknik Informatika. Saat ini penulis sudah menghasilkan 4 book chapter yaitu : Manajemen sumber daya manusia: urgensi, tren dan ruang lingkup Etika bisnis; Pengantar teknologi informasi; Sabda cinta : persembahan DPD ADRI Jawa Timur untuk khasanah tridharma di masa pandemi

D Tri Octafian, M.Kom



Penulis menyelesaikan Pendidikan D1 Analisis Komputer pada Lembaga LPPMK Pranata Mulya di Palembang tahun 2002, menyelesaikan Pendidikan Strata 1 Program Studi Teknik Informatika Konsentrasi Software Engineering pada Universitas Indo Global Mandiri di Palembang tahun 2009, menyelesaikan Pendidikan Strata 2 Program Studi Teknik Informatika Konsentrasi Software Engineering pada Universitas Bina Darma di Palembang tahun 2012. Penulis adalah dosen tetap Program Studi S1 Informatika Institut Teknologi dan Bisnis PalComTech di Palembang. Pengajaran dan penelitian penulis berfokus pada bidang rekayasa perangkat lunak. Penulis adalah *founder channel* Youtube Kidimi yang membahas seputar pemrograman komputer. Selain sebagai dosen, penulis adalah *programmer* lepas.

Dr. Irwanto, S.Pd.T., M.T



Penulis lahir di Jambu (Sulawesi-Selatan), 10 Oktober 1983 merupakan Dosen bidang Pendidikan Vokasional Teknik Elektro, Fakultas Keguruan dan Ilmu Pendidikan (FKIP) Universitas Sultan Ageng Tirtayasa (UNTIRTA), Serang-Banten. Semua Pendidikan mulai dari program Sarjana, Magister dan Doktor di selesaikan di Universitas Negeri Yogyakarta Dengan Jurusan Pendidikan Teknologi dan Kejuruan (PTK). Penyanggah gelar Doktor dengan predikat Sangat Memuaskan merupakan Doktor ke 108 Program Pascasarjana Universitas Negeri Yogyakarta dan merupakan Doktor ke 18 Program Studi Pendidikan Teknologi Kejuruan (PTK) Universitas Negeri Yogyakarta. Di luar itu, juga sebagai asisten pengajar Prof. Dr. Sugiyono, M.Pd. Untuk Mata Kuliah Statistik Pendidikan, Manajemen Pendidikan, Teknik Analisis Data Kuantitatif dan Metode Penelitian Pendidikan di Program Pascasarjana Universitas Negeri Yogyakarta. Dalam kesibukannya itu, Irwanto masih aktif sebagai tenaga pengajar di Universitas Sultan Ageng Tirtayasa, Serang-Banten, di program studi pendidikan vokasional teknik elektro

pada Fakultas Keguruan dan Ilmu Pendidikan (FKIP) Universitas Sultan Ageng Tirtayasa, Serang-Banten. Adapun mata kuliah yang diberikan adalah Matematika Teknik 1, Matematika Teknik 2, Elektronika Daya, Metodologi Penelitian Pendidikan Teknik Elektro, Telaah Kurikulum Pendidikan Kejuruan, Pengelolaan Pendidikan, Literasi ICT dan Media Pembelajaran, Pembinaan Kompetensi Pembelajaran, Mikrokontroler, Neuro Fuzzy, Praktik Elektronika Daya, Manajemen Pendidikan Kejuruan, Medan Elektromagnet, Struktur dan Organisasi Komputer, Kurikulum dan Pembelajaran, dan lain-lain. Juga, telah menulis puluhan artikel ilmiah dan ilmiah populer. Ia pernah melakukan studi banding bidang vokasional antara lain, Malaysia, Singapura untuk menambah wawasan studi dalam bidang pendidikan vokasional teknik elektro (PVTE) tersebut, sehingga keahlian yang dimiliki adalah manajemen pendidikan kejuruan yang ditekuni sampai sekarang ini.

Rekayasa Perangkat Lunak



Istilah *software engineering*, pertama kali digunakan pada akhir tahun 1950-an dan sekitar awal 1960-an. Pada tahun 1968, NATO menyelenggarakan konferensi tentang *software engineering* di Jerman dan kemudian dilanjutkan pada tahun 1969. Meski penggunaan kata *software engineering* masukan konferensi tersebut menimbulkan debat tajam tentang aspek *engineering* dari pengembangan perangkat lunak, banyak pihak yang menganggap konferensi tersebutlah yang menjadi awal tumbuhnya profesi rekayasa perangkat lunak. Rekayasa Perangkat Lunak (RPL, atau dalam bahasa Inggris: *Software Engineering* atau SE) adalah satu bidang profesi yang mendalami cara-cara pengembangan perangkat lunak termasuk pembuatan, pemeliharaan, manajemen organisasi pengembangan perangkat lunak dan manajemen kualitas. *IEEE Computer Society* mendefinisikan rekayasa perangkat lunak sebagai penerapan suatu pendekatan yang sistematis, disiplin dan terkuantifikasi atas pengembangan, penggunaan dan pemeliharaan perangkat lunak, serta studi atas pendekatan-pendekatan ini, yaitu penerapan pendekatan *engineering* atas perangkat lunak. Rekayasa perangkat lunak mengubah perangkat lunak itu sendiri guna mengembangkan, memelihara, dan membangun kembali dengan menggunakan prinsip rekayasa untuk menghasilkan perangkat lunak yang dapat bekerja lebih efisien dan efektif untuk pengguna. Rekayasa Perangkat Lunak di Indonesia dijadikan disiplin ilmu yang dipelajari mulai tingkat Sekolah Menengah Kejuruan sampai tingkat Perguruan Tinggi. Berdasarkan hal tersebut, maka buku ini menyajikan segala yang dibutuhkan oleh para pelaku pengguna atau pengelola Perangkat Lunak dalam menjalankan roda perputaran teknologinya agar dapat menciptakan kualitas dan kuantitas teknologi yang baik. Oleh sebab itu buku ini hadir dihadapan pembaca sebagai bagian dari upaya diskusi sekaligus dalam rangka melengkapi khazanah keilmuan di bidang Perangkat Lunak, sehingga buku ini sangat cocok untuk dijadikan bahan acuan bagi kalangan intelektual dilingkungan perguruan tinggi ataupun praktisi yang berkecimpung langsung di bidang Perangkat Lunak.