



TIF08 - Rekayasa Perangkat Lunak

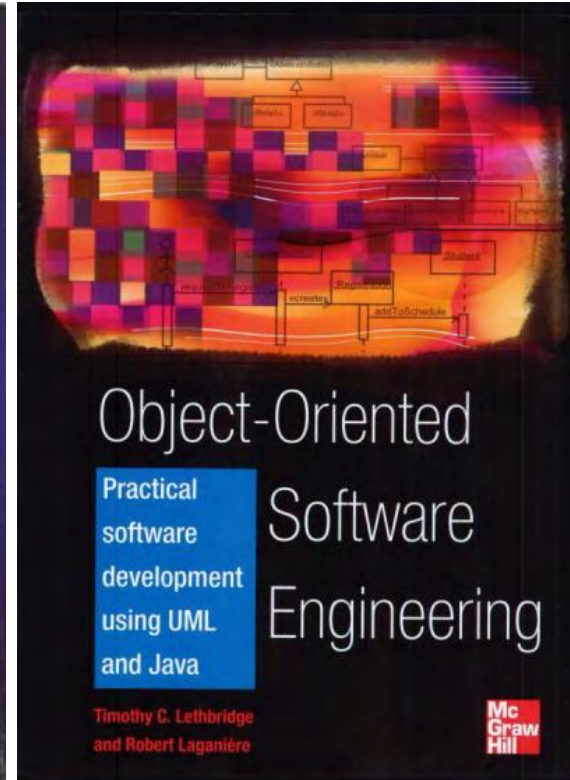
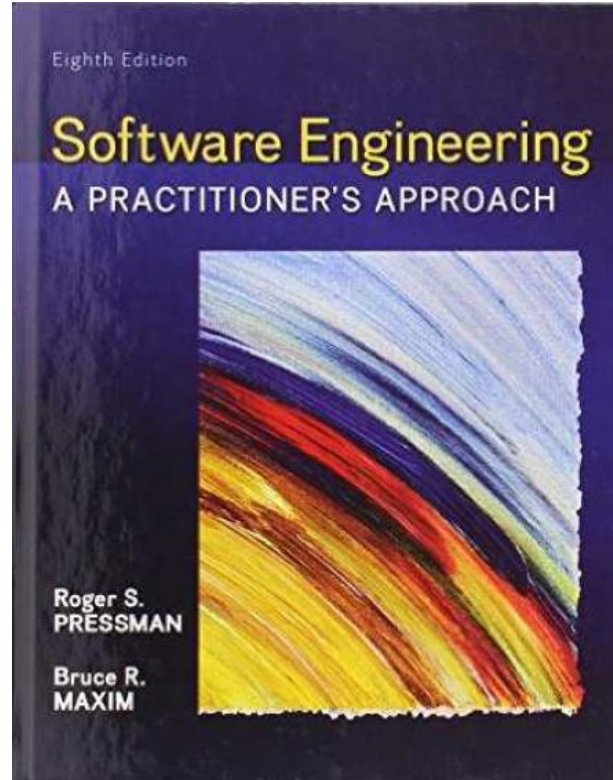
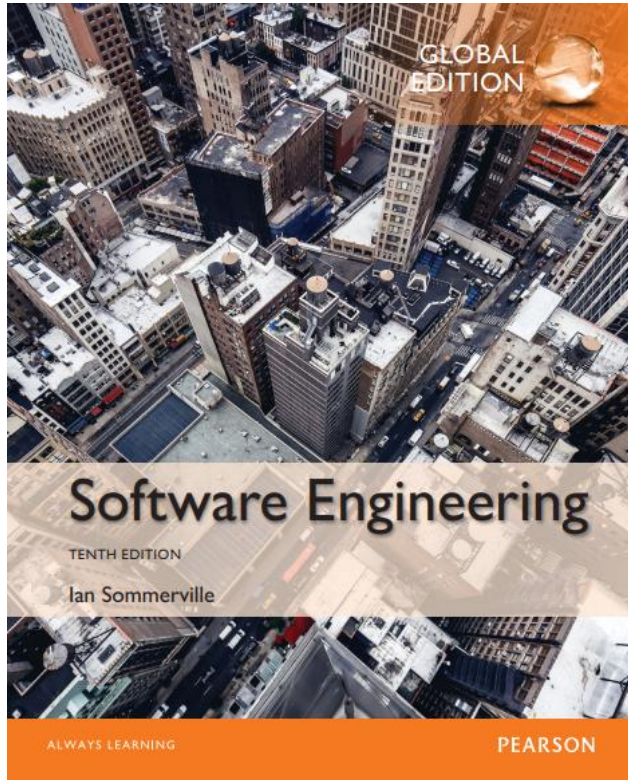
Capaian Pembelajaran Mata Kuliah

- Mahasiswa dapat menjelaskan definisi dan prinsip-prinsip rekayasa perangkat lunak (C2,A2).
- Mahasiswa dapat menentukan kebutuhan dan persyaratan dalam pengembangan perangkat lunak (C3,A3).
- Mahasiswa dapat memilih metode pengembangan Sistem yang sesuai dengan karakteristik perangkat lunak (C3,A3).
- Mahasiswa dapat merencanakan dan menggambarkan pemodelan sistem yang sesuai dengan kebutuhan sistem (C4,A3).



GAMBARAN UMUM REKAYASA PERANGKAT LUNAK

Pertemuan ke: 1



Sub-CPMK

- Mahasiswa mampu menjelaskan kembali definisi dari Rekayasa Perangkat Lunak (C2, A2).

Materi

1. Pengenalan dan Penjelasan Sistem Perkuliahan
2. Definisi Perangkat Lunak
3. Definisi Rekayasa Perangkat Lunak
4. Etika Rekayasa Perangkat Lunak



1. Pengenalan dan Penjelasan Sistem Perkuliahan

1.1 RPL

- RPL (Rekayasa Perangkat Lunak) / *Software Engineering* merupakan yang mempelajari dibidang kereyasaan dalam pengembangan perangkat lunak, sehingga mendapatkan perangkat lunak yang berkualitas sesuai degan kebutuhan pengguna (*user*).

1.1 RPL (Lanj.)

- Sehingga RPL merupakan suatu matakuliah yang diajarkan pada bidang studi Informatika, Sistem Informasi, Ilmu Komputer, Bisnis Digital, dan yang berkaitan dengan bidang ilmu kerekyasaan perangkat lunak.

1.1 RPL (Lanj.)

- Bidang kerekayasaan perangkat lunak berkembang sesuai dengan perkembangan jaman, dan kebutuhan perangkat lunak.
- Sehingga alat pemodelan dan tahapan pengembangan (metodologi/*framework*) juga mengalami perkembangan.

1.2 Sistem Perkuliahan

- Pada materi perkuliahan ini sepenuhnya adalah teori dengan beban SKS yaitu 4 SKS.
- Matakuliah ini juga berkaitan dengan analisis dan desain sistem seperti: Analisis dan Desain Sistem baik yang non objek maupun yang berorientasi objek.

1.2 Sistem Perkuliahan (Lanj.)

- Juga berkaitan dengan pemrograman, sehingga perpaduan antara analisis dan desain sistem yang dikombinasikan dengan RPL dapat menghasilkan perangkat lunak yang berkualitas sesuai dengan konsep (teori) dibidang rekayasa perangkat lunak.



2. Definisi Perangkat Lunak

Definisi Perangkat Lunak

- Pada materi ini merupakan pendahuluan sebelum menjelaskan inti dari Matakuliah (Bahan Ajar) RPL.
- Setelah mempelajari konten materi ini mahasiswa memiliki pengetahuan dibidang kerekayasaan perangkat lunak.

Definisi Perangkat Lunak (Lanj..)

- Sehingga pada konten ini membahas:
 - Pengertian perangkat lunak.
 - Pengertian rekayasa perangkat lunak.
 - Proses rekayasa perangkat lunak.
 - Etika rekayasa perangkat lunak.

Definisi Perangkat Lunak (Lanj..)

- Perangkat lunak juga sering disebut dengan sistem perangkat lunak.
- Sistem berarti kumpulan komponen yang saling terkait dan mempunyai satu tujuan yang ingin dicapai.

Definisi Perangkat Lunak (Lanj..)

- **Perangkat Lunak** (*software*) adalah: “program komputer yang terasosiasi dengan dokumentasi perangkat lunak (*user manual*). Sebuah program komputer tanpa terasosiasi dengan dokumentasinya maka belum dapat disebut perangkat lunak (*software*)” [Rosa].

Definisi Perangkat Lunak (Lanj..)

Karakteristik perangkat lunak yaitu:

- Perangkat lunak dibangun dengan rekayasa (*software engineering*) bukan diproduksi secara manufaktur atau pabrikasi.
- Perangkat lunak tidak pernah usang, karena kecacatan perangkat lunak dapat diperbaiki.

Definisi Perangkat Lunak (Lanj..)

- Barang produksi pabrikan biasanya komponen barunya akan terus diproduksi, sedangkan perangkat lunak biasanya terus diperbaiki seiring bertambahnya kebutuhan.

Definisi Perangkat Lunak (Lanj..)

Aplikasi dari perangkat lunak adalah:

- **Perangkat lunak sistem** (*system software*): adalah kumpulan program dalam hal ini program yang dibuat untuk memenuhi kebutuhan program lainnya.

Definisi Perangkat Lunak (Lanj..)

- **Perangkat lunak waktu nyata (*real-time software*):** merupakan perangkat lunak yang memonitor, menganalisis, mengontrol sesuatu secara waktu nyata. Reaksi yang dibutuhkan pada perangkat lunak harus langsung menghasilkan respon yang diinginkan.

Definisi Perangkat Lunak (Lanj..)

- **Perangkat lunak bisnis** (*business software*): merupakan perangkat lunak pengelola informasi bisnis (seperti akuntansi, penjualan, pembayaran, penyimpanan / *inventory*, dan lain sebagainya).

Definisi Perangkat Lunak (Lanj..)

- **Perangkat lunak untuk rekayasa & keilmuan** (*engineering and scientific software*): merupakan perangkat lunak yang mengimplementasikan algoritma yang terkait dengan keilmuan ataupun perangkat lunak yang membantu keilmuan.

Definisi Perangkat Lunak (Lanj..)

- **Perangkat lunak tambahan (*embedded software*):** merupakan perangkat lunak tambahan untuk membantu mengerjakan suatu fungsi dari perangkat lunak yang lain.

Definisi Perangkat Lunak (Lanj..)

- **Perangkat lunak personal** (*personal computer software*): merupakan perangkat lunak tambahan untuk PC misalkan pengolah kata, pengolah data, pengolah tabel (*worksheet*), untuk presentasi, dll.

Definisi Perangkat Lunak (Lanj..)

- **Perangkat lunak berbasis web** (*web base software*): merupakan perangkat lunak yang dapat diakses dengan menggunakan *browser*.

Definisi Perangkat Lunak (Lanj..)

- **Perangkat lunak berintelijensia buatan** (*artificial intelligence software*): merupakan perangkat lunak yang menggunakan algoritma tertentu untuk mengelola data sehingga seakan-akan memiliki intelijensia seiring bertambahnya data yang diproses.



3. Definisi Rekayasa Perangkat Lunak

Definisi Rekayasa Perangkat Lunak

- Pekerjaan yang terkait dengan rekayasa perangkat lunak dapat dikategorikan menjadi 3, yaitu:
 1. Fase pendefinisian fokus pada “***what***”
 2. Fase pengembangan fokus pada “***how***”
 3. Fase pendukung fokus pada perbaikan “***error***”

Definisi Rekayasa Perangkat Lunak (Lanj..)

- Fase pendukung terdiri dari empat perubahan antara lain:
 1. Koreksi (*correction*).
 2. Adaptasi (*adaptation*).
 3. Perbaikan (*enhancement*).
 4. Pencegahan (*prevention*).

Definisi Rekayasa Perangkat Lunak (Lanj..)

- Pengembang perangkat lunak peduli dengan pengembangan produk perangkat lunak, yaitu perangkat lunak yang dapat dijual ke pelanggan.
- Ada dua macam produk perangkat lunak, yaitu:

Definisi Rekayasa Perangkat Lunak (Lanj..)

1. ***Generic products*** / Produk generik Ini adalah sistem yang berdiri sendiri yang diproduksi oleh organisasi pengembangan dan dijual di pasar terbuka kepada setiap pelanggan yang mampu membelinya. Contoh: jenis produk ini mencakup aplikasi untuk perangkat seluler, perangkat lunak untuk PC, pengolah kata, paket gambar, dan alat manajemen proyek.

Definisi Rekayasa Perangkat Lunak (Lanj..)

2. **Customized** / Perangkat lunak yang disesuaikan (atau dipesan lebih dahulu) Ini adalah sistem yang ditugaskan oleh dan dikembangkan untuk pelanggan tertentu. Pengembang perangkat lunak merancang dan mengimplementasikan perangkat lunak tersebut terutama untuk pelanggan tersebut. Contoh: perangkat lunak jenis ini antara lain sistem kendali untuk perangkat elektronik, sistem yang ditulis untuk mendukung proses bisnis tertentu, dll.

3.1 Rekayasa Perangkat Lunak

- Rekayasa perangkat lunak (*software engineering*) merupakan pembangunan dengan menggunakan prinsip atau konsep rekayasa dengan tujuan menghasilkan perangkat lunak yang bernilai ekonomis yang dipercaya dan berkerja secara efisien menggunakan mesin. [Rosa].

3.1 Rekayasa Perangkat Lunak (Lanj.)

- Rekayasa perangkat lunak adalah: “disiplin teknik yang berkaitan dengan semua aspek produksi perangkat lunak dari tahap awal spesifikasi sistem hingga pemeliharaan sistem setelah digunakan” [Sommerville].
- Dalam definisi ini, ada dua frase kunci, yaitu:

3.1 Rekayasa Perangkat Lunak (Lanj.)

1. *Engineering* (Kerekayasaan) membuat sesuatu bekerja, mereka menerapkan teori, metode, dan alat jika sesuai. Mereka menggunakannya secara selektif & selalu berusaha menemukan solusi untuk masalah meskipun tidak ada teori & metode yang berlaku. *Engineering* juga menyadari bahwa mereka harus bekerja dlm batasan organisasi & keuangan, dan mereka harus mencari solusi dalam batasan ini.

3.1 Rekayasa Perangkat Lunak (Lanj.)

2. Semua aspek produksi rekayasa perangkat lunak tidak hanya berkaitan dengan proses teknis pengembangan perangkat lunak. Ini juga mencakup kegiatan seperti manajemen proyek perangkat lunak dan pengembangan alat, metode, dan teori untuk mendukung pengembangan perangkat lunak.

3.1 Rekayasa Perangkat Lunak (Lanj.)

Karakteristik produk	Penjelasan
Akseptabilitas	Perangkat lunak harus dapat diterima oleh jenis pengguna yang dirancang untuk itu. Ini berarti bahwa itu harus dapat dimengerti, dapat digunakan, dan kompatibel dengan sistem lain yang mereka gunakan.
Ketergantungan dan keamanan	Ketergantungan perangkat lunak mencakup berbagai karakteristik termasuk keandalan, keamanan, dan keselamatan. Perangkat lunak yang dapat diandalkan tidak boleh menyebabkan kerusakan fisik atau ekonomi jika terjadi kegagalan sistem. Perangkat lunak harus aman sehingga pengguna jahat tidak dapat mengakses atau merusak sistem.

3.1 Rekayasa Perangkat Lunak (Lanj.)

Karakteristik produk	Penjelasan
Efisiensi	Perangkat lunak tidak boleh menggunakan sumber daya sistem seperti siklus memori dan prosesor secara boros. Oleh karena itu, efisiensi mencakup daya tanggap, waktu pemrosesan, pemanfaatan sumber daya, dll.
Pemeliharaan	Perangkat lunak harus ditulis sedemikian rupa sehingga dapat berkembang untuk memenuhi kebutuhan pelanggan yang terus berubah. Ini adalah atribut penting karena perubahan perangkat lunak merupakan persyaratan yang tak terhindarkan dari lingkungan bisnis yang berubah.

3.1 Rekayasa Perangkat Lunak (Lanj.)

- *Engineering* (Kerekayasaan) adalah: tentang mendapatkan hasil dari kualitas yang dibutuhkan sesuai jadwal dan anggaran.
- Ini sering kali melibatkan kompromi—*engineers* tidak bisa menjadi perfeksionis.

3.1 Rekayasa Perangkat Lunak (Lanj.)

- Orang yang menulis program untuk diri mereka sendiri, bagaimanapun, dapat menghabiskan waktu sebanyak yang mereka inginkan untuk pengembangan program.

3.1 Rekayasa Perangkat Lunak (Lanj.)

- Secara umum, *engineers* perangkat lunak mengadopsi pendekatan sistematis dan terorganisir untuk pekerjaan mereka, karena ini seringkali merupakan cara paling efektif untuk menghasilkan perangkat lunak berkualitas tinggi.

3.1 Rekayasa Perangkat Lunak (Lanj.)

- Namun, teknik adalah tentang memilih metode yang paling tepat untuk serangkaian keadaan, jadi pendekatan pengembangan yang lebih kreatif dan kurang formal mungkin yang tepat untuk beberapa jenis perangkat lunak.

3.1 Rekayasa Perangkat Lunak (Lanj.)

- Teknologi perangkat lunak yang lebih fleksibel yang mengakomodasi perubahan cepat sangat sesuai untuk pengembangan sistem berbasis web interaktif dan aplikasi seluler, yang memerlukan perpaduan keterampilan perangkat lunak dan desain grafis.

3.1 Rekayasa Perangkat Lunak (Lanj.)

RPL penting karena dua alasan:

1. Semakin banyak, individu dan masyarakat bergantung pada sistem perangkat lunak yang canggih. Kita harus mampu menghasilkan sistem yang andal dan dapat dipercaya secara ekonomis dan cepat.

3.1 Rekayasa Perangkat Lunak (Lanj.)

2. Biasanya lebih murah, dalam jangka panjang, menggunakan metode dan teknik rekayasa perangkat lunak untuk sistem perangkat lunak profesional daripada hanya menulis program sebagai proyek pemrograman pribadi. Kegagalan menggunakan metode rekayasa perangkat lunak menyebabkan biaya yang lebih tinggi untuk pengujian, jaminan kualitas, dan pemeliharaan jangka panjang.

3.1 Rekayasa Perangkat Lunak (Lanj.)

- Pendekatan sistematis yang digunakan dalam rekayasa perangkat lunak kadang-kadang disebut proses perangkat lunak.
- Proses perangkat lunak adalah urutan aktivitas yang mengarah pada produksi produk perangkat lunak.
- Empat aktivitas dasar umum untuk semua proses perangkat lunak, **yaitu:**

3.1 Rekayasa Perangkat Lunak (Lanj.)

1. **Spesifikasi perangkat lunak:** pelanggan dan *engineers* menentukan perangkat lunak yang akan diproduksi dan batasan pengoperasiannya.
2. **Pengembangan perangkat lunak:** perangkat lunak dirancang dan diprogram.

3.1 Rekayasa Perangkat Lunak (Lanj.)

3. **Validasi perangkat lunak:** perangkat lunak diperiksa untuk memastikan bahwa itu adalah apa yang dibutuhkan pelanggan.
4. **Evolusi perangkat lunak:** perangkat lunak dimodifikasi untuk mencerminkan kebutuhan pelanggan dan pasar yang berubah.

3.1 Rekayasa Perangkat Lunak (Lanj.)

- Jenis sistem yang berbeda memerlukan proses pengembangan yang berbeda.
- Sebagai contoh, perangkat lunak waktu nyata di pesawat terbang harus ditentukan secara lengkap sebelum pengembangan dimulai.

3.1 Rekayasa Perangkat Lunak (Lanj.)

- Dalam sistem *e-commerce*, spesifikasi dan program biasanya dikembangkan bersama. Akibatnya, aktivitas umum ini dapat diatur dengan cara yang berbeda dan dijelaskan pada tingkat detail yang berbeda, tergantung pada jenis perangkat lunak yang dikembangkan.

3.1 Rekayasa Perangkat Lunak (Lanj.)

Rekayasa perangkat lunak terkait dengan ilmu komputer dan rekayasa sistem.

1. Ilmu komputer berkaitan dengan teori dan metode yang mendasari komputer dan sistem perangkat lunak, sedangkan rekayasa perangkat lunak berkaitan dengan masalah praktis dalam menghasilkan perangkat lunak.

3.1 Rekayasa Perangkat Lunak (Lanj.)

Beberapa pengetahuan tentang ilmu komputer sangat penting bagi pengembang perangkat lunak dengan cara yang sama seperti beberapa pengetahuan fisika penting untuk insinyur listrik. Teori ilmu komputer, bagaimanapun, seringkali paling dapat diterapkan pada program yang relatif kecil. Teori ilmu komputer yang elegan jarang sekali relevan dengan masalah besar dan kompleks yang memerlukan solusi perangkat lunak.

3.1 Rekayasa Perangkat Lunak (Lanj.)

2. Rekayasa sistem berkaitan dengan semua aspek pengembangan dan evolusi sistem yang kompleks di mana perangkat lunak memainkan peran utama. Oleh karena itu, rekayasa sistem berkaitan dengan pengembangan perangkat keras, desain kebijakan dan proses, dan penerapan sistem, serta rekayasa perangkat lunak.

3.1 Rekayasa Perangkat Lunak (Lanj.)

Insinyur (pengembang) sistem dilibatkan dalam menentukan sistem, mendefinisikan arsitektur keseluruhannya, dan kemudian mengintegrasikan bagian-bagian yang berbeda untuk membuat sistem yang sudah jadi.

3.1 Rekayasa Perangkat Lunak (Lanj.)

- Tidak ada metode atau teknik rekayasa perangkat lunak universal yang dapat digunakan.
- Namun, ada empat masalah terkait yang memengaruhi berbagai jenis perangkat lunak:

3.1 Rekayasa Perangkat Lunak (Lanj.)

1. Heterogenitas Semakin banyak sistem yang diperlukan untuk beroperasi sebagai sistem terdistribusi di seluruh jaringan yang mencakup berbagai jenis komputer dan perangkat seluler. Selain berjalan di komputer tujuan umum, perangkat lunak mungkin juga harus dijalankan di ponsel,

3.1 Rekayasa Perangkat Lunak (Lanj.)

Kita sering kali harus mengintegrasikan perangkat lunak baru dengan sistem lama yang ditulis dalam bahasa pemrograman berbeda. Tantangannya di sini adalah mengembangkan teknik untuk membangun perangkat lunak yang dapat diandalkan yang cukup fleksibel untuk mengatasi heterogenitas ini.

3.1 Rekayasa Perangkat Lunak (Lanj.)

2. Perubahan bisnis dan sosial Bisnis dan masyarakat berubah sangat cepat seiring berkembangnya ekonomi dan teknologi baru tersedia. Mereka harus dapat mengubah perangkat lunak yang ada dan mengembangkan perangkat lunak baru dengan cepat.

3.1 Rekayasa Perangkat Lunak (Lanj.)

Banyak teknik rekayasa perangkat lunak tradisional memakan waktu, dan pengiriman sistem baru seringkali memakan waktu lebih lama dari yang direncanakan. Mereka perlu berevolusi sehingga waktu yang dibutuhkan perangkat lunak untuk memberikan nilai kepada pelanggannya berkurang.

3.1 Rekayasa Perangkat Lunak (Lanj.)

3. Keamanan dan kepercayaan Karena perangkat lunak terkait dengan semua aspek kehidupan kita, sangatlah penting bagi kita untuk mempercayai perangkat lunak itu. Hal ini terutama berlaku untuk sistem perangkat lunak jarak jauh yang diakses melalui halaman web atau antarmuka layanan web.

3.1 Rekayasa Perangkat Lunak (Lanj.)

Kami harus memastikan bahwa pengguna jahat tidak berhasil menyerang perangkat lunak kami dan keamanan informasi tetap terjaga.

3.1 Rekayasa Perangkat Lunak (Lanj.)

4. *Scale Software* harus dikembangkan di berbagai skala yang sangat luas, dari sistem tertanam yang sangat kecil di perangkat portabel atau perangkat yang dapat dikenakan hingga sistem berskala *Internet* berbasis *cloud* yang melayani komunitas global.

3.1 Rekayasa Perangkat Lunak (Lanj.)

- Untuk mengatasi tantangan ini, kita memerlukan alat dan teknik baru serta cara inovatif untuk menggabungkan dan menggunakan metode rekayasa perangkat lunak yang ada.

3.2 Keragaman Rekayasa Perangkat Lunak

- Rekayasa perangkat lunak adalah pendekatan sistematis untuk produksi perangkat lunak yang memperhitungkan masalah biaya praktis, jadwal, dan ketergantungan, serta kebutuhan pelanggan dan produsen perangkat lunak.

3.2 Keragaman Rekayasa Perangkat Lunak (Lanj.)

- Metode, alat, dan teknik khusus yang digunakan bergantung pada organisasi yang mengembangkan perangkat lunak, jenis perangkat lunak, dan orang-orang yang terlibat dalam proses pengembangan.

3.2 Keragaman Rekayasa Perangkat Lunak (Lanj.)

- Tidak ada metode rekayasa perangkat lunak universal yang cocok untuk semua sistem dan semua perusahaan.
- Sebaliknya, beragam metode dan alat rekayasa perangkat lunak telah berkembang selama 50 tahun terakhir.

3.2 Keragaman Rekayasa Perangkat Lunak (Lanj.)

- Mungkin faktor yang paling signifikan dalam menentukan metode dan teknik rekayasa perangkat lunak mana yang paling penting adalah jenis aplikasi yang dikembangkan.
- Ada banyak jenis aplikasi, termasuk:

3.2 Keragaman Rekayasa Perangkat Lunak (Lanj.)

1. **Stand-Alone:** adalah aplikasi yang berjalan di komputer pribadi atau aplikasi yang berjalan di perangkat seluler. Mereka mencakup semua fungsionalitas yang diperlukan dan mungkin tidak perlu disambungkan ke jaringan. Contoh: CAD, editing foto, dan sebagainya.

3.2 Keragaman Rekayasa Perangkat Lunak (Lanj.)

2. **Interactive transaction-based applications:** aplikasi dijalankan di komputer jarak jauh dan diakses oleh pengguna dari komputer, telepon, tablet mereka sendiri. ini termasuk aplikasi web seperti aplikasi *e-commerce* tempat berinteraksi dengan sistem jarak jauh untuk membeli barang dan jasa.

3.2 Keragaman Rekayasa Perangkat Lunak (Lanj.)

3. **Embedded control systems:** adalah sistem kontrol perangkat lunak yang mengontrol dan mengelola perangkat perangkat keras. Contoh sistem tertanam perangkat lunak di ponsel (seluler), perangkat lunak yang mengontrol pengereman antipenguncian di mobil, dll.

3.2 Keragaman Rekayasa Perangkat Lunak (Lanj.)

4. **Sistem pemrosesan batch:** adalah sistem bisnis yang dirancang untuk memproses data dalam batch besar. Mereka memproses *input* individu dalam jumlah besar untuk membuat *output* yang sesuai. Contoh sistem *batch* adalah sistem penagihan berkala, seperti sistem penagihan telepon, dan sistem pembayaran gaji.

3.2 Keragaman Rekayasa Perangkat Lunak (Lanj.)

- 5. Entertainment systems:** adalah sistem untuk penggunaan pribadi yang dimaksudkan untuk menghibur pengguna. Sebagian besar sistem ini adalah permainan dari satu jenis atau lainnya, yang dapat berjalan pada perangkat keras konsol tujuan khusus.

3.2 Keragaman Rekayasa Perangkat Lunak (Lanj.)

- 6. Systems for modeling and simulation:** adalah sistem yang dikembangkan oleh para ilmuwan dan insinyur untuk memodelkan proses atau situasi fisik, yang mencakup banyak objek yang terpisah dan berinteraksi. Ini sering kali intensif secara komputasi dan membutuhkan sistem paralel berkinerja tinggi untuk dieksekusi.

3.2 Keragaman Rekayasa Perangkat Lunak (Lanj.)

7. **Sistem pengumpulan dan analisis data:** adalah sistem yang mengumpulkan data dari lingkungannya dan mengirimkan data tersebut ke sistem lain untuk diproses. Analisis "***big data***" mungkin melibatkan sistem berbasis *cloud* yang melakukan analisis statistik dan mencari hubungan dalam data yang dikumpulkan.

3.2 Keragaman Rekayasa Perangkat Lunak (Lanj.)

8. **Systems of systems:** adalah sistem yang digunakan di perusahaan dan organisasi besar lainnya, yang terdiri dari sejumlah sistem perangkat lunak lain. Beberapa di antaranya mungkin merupakan produk perangkat lunak umum, seperti sistem ERP.

3.2 Keragaman Rekayasa Perangkat Lunak (Lanj.)

- Namun demikian, ada dasar-dasar rekayasa perangkat lunak yang berlaku untuk semua jenis sistem perangkat lunak.
- Adapun dasar-dasar rekayasa perangkat lunak yang berlaku untuk semua jenis sistem perangkat lunak adalah:

3.2 Keragaman Rekayasa Perangkat Lunak (Lanj.)

1. Mereka harus dikembangkan menggunakan proses pembangunan yang dikelola dan dipahami. Organisasi yang mengembangkan perangkat lunak harus merencanakan proses pengembangan dan memiliki gagasan yang jelas tentang apa yang akan diproduksi dan kapan akan selesai.

3.2 Keragaman Rekayasa Perangkat Lunak (Lanj.)

2. Ketergantungan dan kinerja penting untuk semua jenis sistem. Perangkat lunak harus berperilaku seperti yang diharapkan, tanpa kegagalan, dan harus tersedia untuk digunakan saat diperlukan. Itu harus aman dalam operasinya dan, sejauh mungkin, harus aman dari serangan eksternal. Sistem harus bekerja secara efisien dan tidak membuang sumber daya.

3.2 Keragaman Rekayasa Perangkat Lunak (Lanj.)

3. Memahami dan mengelola spesifikasi dan persyaratan perangkat lunak (apa yang harus dilakukan perangkat lunak) adalah penting. Anda harus mengetahui apa yang diharapkan oleh pelanggan dan pengguna sistem yang berbeda darinya, dan Anda harus mengelola ekspektasi mereka sehingga sistem yang berguna dapat disampaikan sesuai anggaran dan jadwal.

3.2 Keragaman Rekayasa Perangkat Lunak (Lanj.)

4. Anda harus memanfaatkan sumber daya yang ada secara efektif. Artinya, jika sesuai, Anda harus menggunakan kembali perangkat lunak yang telah dikembangkan daripada membuat perangkat lunak baru.

3.3 Rekayasa Perangkat Lunak Internet

- Perkembangan Internet dan World Wide Web telah memberikan pengaruh yang sangat besar pada semua kehidupan kita. Awalnya, web pada dasarnya adalah penyimpanan informasi yang dapat diakses secara universal, dan tidak banyak berpengaruh pada sistem perangkat lunak.

3.3 Rekayasa Perangkat Lunak Internet (Lanj.)

- Perkembangan browser web yang dapat menjalankan program kecil dan melakukan beberapa pemrosesan lokal menyebabkan evolusi dalam bisnis dan perangkat lunak organisasi.

3.3 Rekayasa Perangkat Lunak Internet (Lanj.)

- Sekarang ini telah menjadi pendekatan standar untuk pengiriman produk sistem berbasis web seperti Google Apps, Microsoft Office 365, dan Adobe Creative Suite. Semakin banyak perangkat lunak yang berjalan di "*cloud*" jarak jauh daripada server lokal dan diakses melalui Internet.

3.3 Rekayasa Perangkat Lunak Internet (Lanj.)

- Perubahan dalam organisasi perangkat lunak ini berdampak besar pada rekayasa perangkat lunak untuk sistem berbasis web.
- Sebagai contoh:

3.3 Rekayasa Perangkat Lunak Internet (Lanj.)

1. Penggunaan kembali perangkat lunak telah menjadi pendekatan dominan untuk membangun sistem berbasis web. Saat membangun sistem ini, Anda memikirkan tentang bagaimana Anda dapat merakitnya dari komponen dan sistem perangkat lunak yang sudah ada sebelumnya, sering kali digabungkan dalam satu kerangka kerja.

3.3 Rekayasa Perangkat Lunak Internet (Lanj.)

2. Sekarang secara umum diakui bahwa tidak praktis untuk menentukan semua persyaratan untuk sistem seperti itu sebelumnya. Sistem berbasis web selalu dikembangkan dan disampaikan secara bertahap.

3.3 Rekayasa Perangkat Lunak Internet (Lanj.)

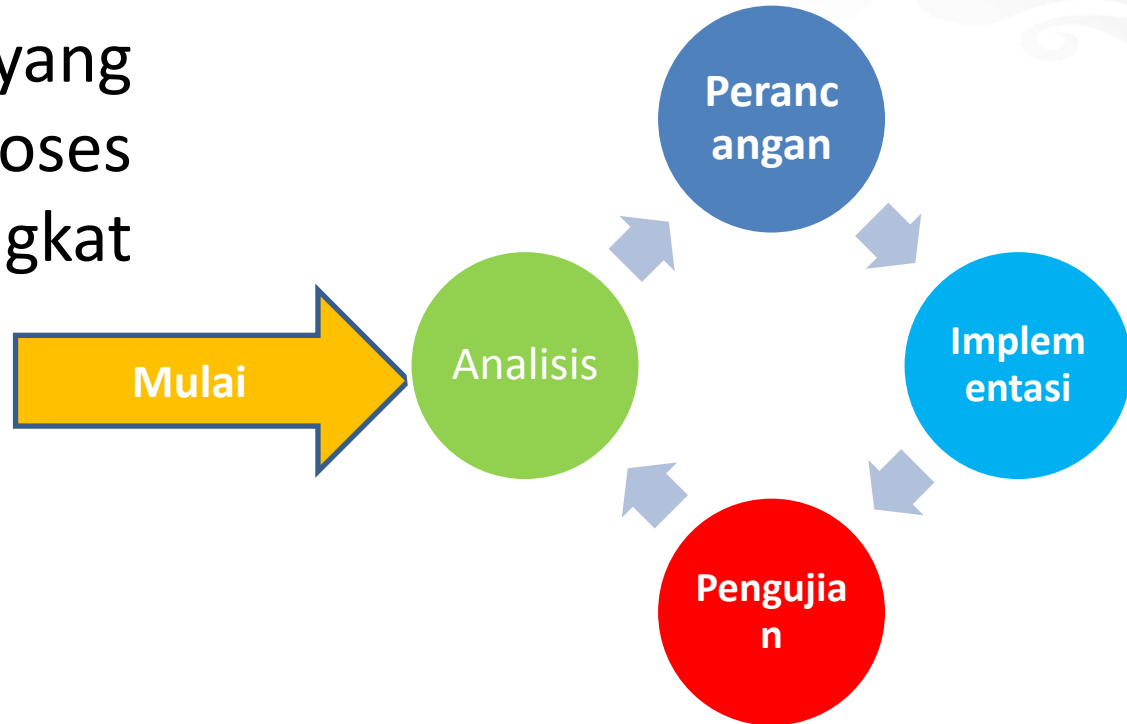
3. Perangkat lunak dapat diimplementasikan menggunakan rekayasa perangkat lunak berorientasi layanan, di mana komponen perangkat lunak adalah layanan web yang berdiri sendiri.

3.3 Rekayasa Perangkat Lunak Internet (Lanj.)

4. Teknologi pengembangan antarmuka seperti AJAX (Holdener 2008) dan HTML5 (Freeman 2011) telah muncul yang mendukung pembuatan antarmuka yang kaya dalam browser web.

3.4 Proses RPL

Garis besar yang dilakukan dalam proses RPL (Rekayasa Perangkat Lunak):



3.4 Proses RPL (Lanj.)

Proses perangkat lunak (*software process*) adalah sekumpulan aktifitas yang memiliki tujuan untuk mengembangkan atau mengubah perangkat lunak.

Secara umum proses perangkat lunak terdiri dari:

3.4 Proses RPL (Lanj.)

1. Pengumpulan spesifikasi (*specification*): mengetahui apa saja yang harus dapat dikerjakan sistem perangkat lunak dan batasan pengembangan perangkat lunak.
2. Pengembangan (*development*): pengembangan perangkat lunak untuk menghasilkan sistem perangkat lunak.

3.4 Proses RPL (Lanj.)

3. Validasi (*validation*): memeriksa apakah perangkat lunak sudah memenuhi keinginan pelanggan (*customer*).
4. Evolusi (*evolution*): mengubah perangkat lunak untuk memenuhi perubahan kebutuhan pelanggan (*customer*).

3.5 PERUBAHAN SIFAT PERANGKAT LUNAK

- Empat kategori besar perangkat lunak berkembang untuk mendominasi industri.
- Namun, kategori-kategori ini masih dalam tahap awal lebih dari satu dekade yang lalu.
 - WebApps.
 - Mobile Applications.
 - Cloud Computing.
 - Product Line Software.

3.5 PERUBAHAN SIFAT PERANGKAT LUNAK (Lanj.)

1. **WebApps:** saat ini WebApps telah berkembang menjadi alat komputasi canggih yang tidak hanya menyediakan fungsi yang berdiri sendiri untuk pengguna akhir, tetapi juga telah terintegrasi dengan database perusahaan dan aplikasi bisnis.

3.5 PERUBAHAN SIFAT PERANGKAT LUNAK (Lanj.)

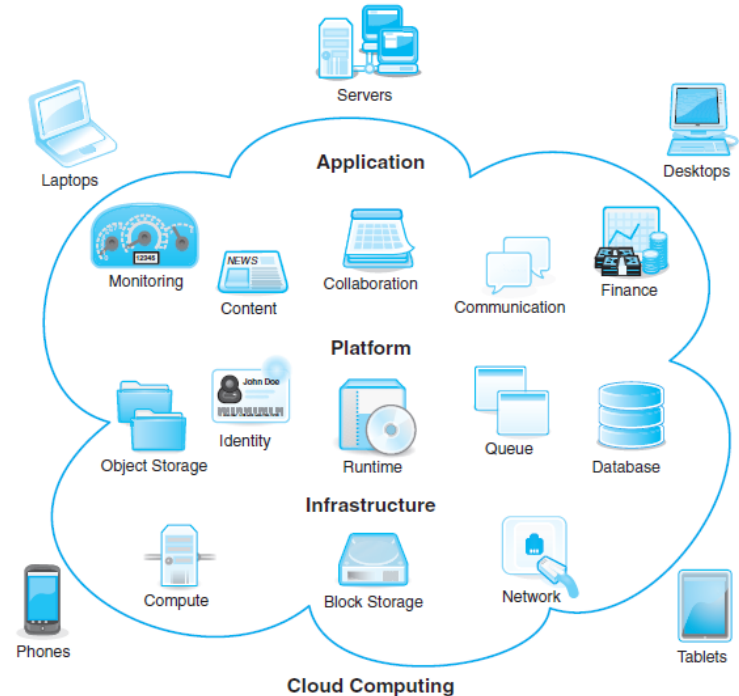
2. **Mobile Applications:** Istilah aplikasi telah berevolusi menjadi perangkat lunak yang secara khusus dirancang untuk berada pada platform seluler (mis., IOS, Android, atau Windows Mobile).

3.5 PERUBAHAN SIFAT PERANGKAT LUNAK (Lanj.)

- 3. Cloud Computing:** mencakup infrastruktur atau "ekosistem" yang memungkinkan setiap pengguna, di mana saja, menggunakan perangkat komputasi untuk berbagi sumber daya komputasi dalam skala luas.

3.5 PERUBAHAN SIFAT PERANGKAT LUNAK (Lanj.)

Arsitektur logis keseluruhan dari komputasi cloud diwakili pada Gambar [Pressman].



3.5 PERUBAHAN SIFAT PERANGKAT LUNAK (Lanj.)

4. **Product Line Software:** Product Line Software menghasilkan pengembangan banyak produk yang direkayasa dengan memanfaatkan kesamaan di antara semua produk dalam line produk tersebut.



4. Etika Rekayasa Perangkat Lunak

Etika Rekayasa Perangkat Lunak

- Seperti disiplin ilmu teknik lainnya, RPL dilakukan dalam kerangka sosial dan hukum yang membatasi kebebasan orang yang bekerja di bidang tersebut.
- Pengembang perangkat lunak, harus menerima bahwa pekerjaan Anda melibatkan tanggung jawab yang lebih luas daripada sekadar penerapan keterampilan teknis.

Etika Rekayasa Perangkat Lunak (Lanj..)

- Anda juga harus berperilaku etis dan bertanggung jawab secara moral jika Anda ingin dihormati sebagai pengembang profesional.
- Sudah jelas bahwa Anda harus menjunjung standar normal tentang kejujuran dan integritas.

Etika Rekayasa Perangkat Lunak (Lanj..)

- Anda tidak boleh menggunakan keterampilan dan kemampuan Anda untuk berperilaku tidak jujur atau dengan cara yang akan merusak reputasi profesi rekayasa perangkat lunak.
- Namun, ada area di mana standar perilaku yang dapat diterima tidak terikat oleh hukum tetapi oleh gagasan tanggung jawab profesional yang lebih lemah.

Etika Rekayasa Perangkat Lunak (Lanj..)

- Beberapa di antaranya adalah:
 1. Kerahasiaan Biasanya Anda harus menghormati kerahasiaan pemberi kerja atau klien Anda terlepas dari apakah perjanjian kerahasiaan formal telah ditandatangani atau belum.

Etika Rekayasa Perangkat Lunak (Lanj..)

2. Kompetensi Anda tidak boleh salah menggambarkan tingkat kompetensi Anda. Anda tidak boleh dengan sengaja menerima pekerjaan di luar kompetensi Anda.

Etika Rekayasa Perangkat Lunak (Lanj..)

3. Hak kekayaan intelektual Anda harus mengetahui undang-undang setempat yang mengatur penggunaan kekayaan intelektual seperti paten dan hak cipta. Anda harus berhati-hati untuk memastikan bahwa kekayaan intelektual pemberi kerja dan klien dilindungi.

Etika Rekayasa Perangkat Lunak (Lanj..)

4. Penyalahgunaan komputer Anda tidak boleh menggunakan keahlian teknis Anda untuk menyalahgunakan komputer orang lain. Penyalahgunaan komputer berkisar dari yang relatif sepele (permainan yang dimainkan di mesin perusahaan) hingga sangat serius (penyebaran virus atau malware lainnya).

Ringkasan

- Rekayasa perangkat lunak adalah disiplin teknik yang berkaitan dengan semua aspek pengembangan (produksi) perangkat lunak.
- *Software* bukan hanya sekedar program tetapi juga mencakup semua dokumentasi yang dibutuhkan oleh pengguna sistem, staf jaminan kualitas, dan pengembang.

Ringkasan (Lanj.)

- Atribut produk perangkat lunak penting seperti: perawatan, keamanan, efisiensi, dan akseptabilitas.
- Proses perangkat lunak mencakup semua aktivitas yang terlibat dalam pengembangan perangkat lunak.

Ringkasan (Lanj.)

- Aktivitas tingkat tinggi seperti: spesifikasi, pengembangan, validasi, dan evolusi adalah bagian dari semua proses perangkat lunak.
- Ada banyak jenis sistem yang berbeda, dan masing-masing memerlukan alat dan teknik rekayasa perangkat lunak yang sesuai untuk pengembangannya.

Ringkasan (Lanj.)

- Ada sedikit (jika ada) desain khusus dan teknik implementasi yang dapat diterapkan pada semua jenis sistem.
- Ide dasar dari rekayasa perangkat lunak dapat diterapkan pada semua jenis sistem perangkat lunak. Dasar-dasar ini mencakup proses perangkat lunak yang dikelola, keandalan dan keamanan perangkat lunak, rekayasa persyaratan, dan penggunaan kembali perangkat lunak.

Ringkasan (Lanj.)

- Insinyur (Pengembang) perangkat lunak memiliki tanggung jawab untuk profesi teknik dan masyarakat. Mereka seharusnya tidak hanya peduli dengan masalah teknis tetapi harus menyadari masalah etika yang mempengaruhi pekerjaan mereka.

Ringkasan (Lanj.)

- Perkumpulan (asosiasi) profesional menerbitkan kode etik yang menanamkan standar etika dan profesional. Ini menetapkan standar perilaku yang diharapkan dari anggotanya.

Latihan

- Diskusikan manfaat mengapa rekayasa perangkat lunak sangat penting dalam pengembangan perangkat lunak.
- Berikan pandangan bagaimana seandainya pengembangan perangkat lunak tidak menerapkan Kerekayasaan dalam membuat perangkat lunak.

Referensi

- Presman, Maxim, Software Engineering, USA.
- Sommerville Ian, 2016, Software Engineering, USA, PEARSON.
- Timoty C, Robert, Object Oriented Software Engineering, USA, Ms Graw Hill.
- Rosa A.S, Rekayasa Perangkat Lunak, Bandung, Penerbit Informatika.



TERIMA KASIH

U N I V E R S I T A S B U N D A M U L I A