# **Pengertian Rekayasa Perangkat Lunak**

Rekayasa perangkat lunak telah berkembang sejak pertama kali diciptakan pada tahun 1940-an hingga kini. Focus utama pengembangannya adalah untuk mengembangkan praktek dan teknologi untuk meningkatkan produktivitas para praktisi pengembang perangkat lunak dan kualitas aplikasi yang dapat digunakan oleh pemakai.

[](http://4.bp.blogspot.com/-Y9AbLPPBLOc/VSwOTiPX-zI/AAAAAAAAArs/ENdo1rlSKH8/s1600/pengertian%2Brpl1.jpg)

**SEJARAH SOFTWARE ENGINEERING**

Istilah software engineering digunakan pertama kali pada akhir 1950-an dan awal 1960-an. Saat itu, masih terdapat perdebatan tajam mengenai aspek engineering dari pengembangan perangkat lunak. Pada tahun 1968 dan 1969, komite sains NATO mensponsori dua konferensi tentang rekayasa perangkat lunak, yang memberikan dampak kuat terhadap pengembangan rekayasa perangkat lunak. Banyak yang menganggap dua konferensi inilah yang menandai awal resmi profesi rekayasa perangkat lunak.

Pada tahun 1960-an hingga 1980-an, banyak masalah yang ditemukan para praktisi pengembangan perangkat lunak. Banyak project yang gagal, hingga masa ini disebut sebagai krisis perangkat lunak. Kasus kegagalan pengembangan perangkat lunak terjadi mulai dari project yang melebihi anggaran, hingga kasus yang mengakibatkan kerusakan fisik dan kematian. Salah satu kasus yang terkenal antara lain meledaknya roket Ariane akibat kegagalan perangkat lunak. Selama bertahun-tahun, para peneliti memfokuskan usahanya untuk menemukan teknik jitu untuk memecahkan masalah krisi perangkat lunak.

Berbagai teknik, metode, alat, proses diciptakan dan diklaim sebagai senjata pamungkas untuk memecahkan kasus ini. Mulai dari pemrograman terstruktur, pemrograman berorientasi objek, perangkat pembantu pengembangan perangkat lunak (CASE tools), berbagai standar, UML hingga metode formal diagung-agungkan sebagai senjata pamungkas untuk menghasilkan software yang benar, sesuai anggaran dan tepat waktu. Pada tahun 1987, Fred Brooks menulis artikel No Silver Bullet, yang berproposisi bahwa tidak ada satu teknologi atau praktek yang sanggup mencapai 10 kali lipat perbaikan dalam produktivitas pengembanan perngkat lunak dalam tempo 10 tahun.

**PENGERTIAN DASAR**

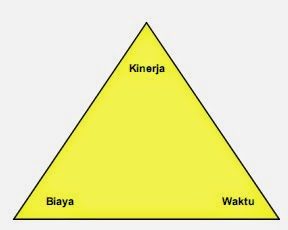
Istilah Reakayasa Perangkat Lunak (RPL) secara umum disepakati sebagai terjemahan dari istilah Software engineering. Istilah Software Engineering mulai dipopulerkan pada tahun 1968 pada software engineering Conference yang diselenggarakan oleh NATO. Sebagian orang mengartikan RPL hanya sebatas pada bagaimana membuat program komputer. Padahal ada perbedaan yang mendasar antara perangkat lunak (software) dan program komputer.

Perangkat lunak adalah seluruh perintah yang digunakan untuk memproses informasi. Perangkat lunak dapat berupa program atau prosedur. Program adalah kumpulan perintah yang dimengerti oleh komputer sedangkan prosedur adalah perintah yang dibutuhkan oleh pengguna dalam memproses informasi (O’Brien, 1999).

RPL sendiri adalah suatu disiplin ilmu yang membahas semua aspek produksi perangkat lunak, mulai dari tahap awal yaitu analisa kebutuhan pengguna, menentukan spesifikasi dari kebutuhan pengguna, disain, pengkodean, pengujian sampai pemeliharaan sistem setelah digunakan. Dari pengertian ini jelaslah bahwa RPL tidak hanya berhubungan dengan cara pembuatan program komputer. Pernyataan ”semua aspek produksi” pada pengertian di atas, mempunyai arti semua hal yang berhubungan dengan proses produksi seperti manajemen proyek, penentuan personil, anggaran biaya, metode, jadwal, kualitas sampai dengan pelatihan pengguna merupakan bagian dari RPL.

**TUJUAN REKAYASA PERANGKAT LUNAK**

Secara umum tujuan RPL tidak berbeda dengan bidang rekayasa yang lain. Hal ini dapat kita lihat pada Gambar di bawah ini.

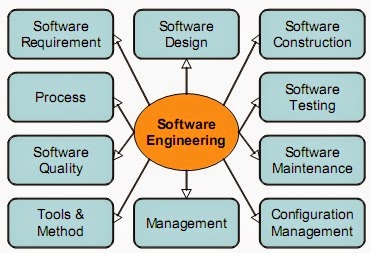
[](http://3.bp.blogspot.com/-nUjUE3O90TE/VSwOW01WHbI/AAAAAAAAAr0/7GFLxYHBGjI/s1600/pengertian%2BRPL.jpg)

Dari Gambar di atas dapat diartikan bahwa bidang rekayasa akan selalu berusaha menghasilkan output yang kinerjanya tinggi, biaya rendah dan waktu penyelesaian yang tepat. Secara lebih khusus kita dapat menyatakan tujuan RPL adalah:

* Memperoleh biaya produksi perangkat lunak yang rendah
* Menghasilkan perangkat lunak yang kinerjanya tinggi, andal dan tepat waktu
* Menghasilkan perangkat lunak yang dapat bekerja pada berbagai jenis platform
* Menghasilkan perangkat lunak yang biaya perawatannya rendah

**RUANG LINGKUP**

Sesuai dengan definisi yang telah disampaikan sebelumnya, maka ruang lingkup RPL dapat digambarkan sebagai berikut:

[](http://3.bp.blogspot.com/-essdJGhhRIE/VSwOXK_o72I/AAAAAAAAAr4/UaNdC9JD7R0/s1600/pengertian%2BRPL2.jpg)

* Software Requirements berhubungan dengan spesifikasi kebutuhan dan persyaratan perangkat lunak
* Software Desain mencakup proses penampilan arsitektur, komponen, antar muka, dan karakteristik lain dari perangkat lunak
* Software Construction berhubungan dengan detail pengembangan perangkat lunak, termasuk algoritma, pengkodean, pengujian dan pencarian kesalahan
* Software Testing meliputi pengujian pada keseluruhan perilaku perangkat lunak
* Software Maintenance mencakup upaya-upaya perawatan ketika perangkat lunak telah dioperasikan
* Software Configuration Management berhubungan dengan usaha perubahan konfigurasi perangkat lunak untuk memenuhi kebutuhan tertentu
* Software Engineering management berkaitan dengan pengelolaan dan pengukuran RPL, termasuk perencanaan proyek perangkat lunak
* Software Engineering Tools And Methods mencakup kajian teoritis tentang alat bantu dan metode RPL
* Software Engineering Process berhubungan dengan definisi, implementasi pengukuran, pengelolaan, perubahan dan perbaikan proses RPL
* Software Quality menitik beratkan pada kualitas dan daur hidup perangkat lunak