

PERENCANAAN PROYEK PERANGKAT LUNAK

3 Langkah Perencanaan :

- I. Pendefinisian masalah,
- II. Pengembangan strategi solusi,
- III. Rencana proses pengembangan.

I. Pendefinisian Masalah

1. Nyatakan masalah yang akan diselesaikan secara tegas
 - pernyataan masalah dalam sudut pelanggan
 - pernyataan masalah dalam sudut pengembang
 - Teknik-teknik yang digunakan untuk mendapatkan informasi kebutuhan pelanggan

2. Rancang sebuah strategi solusi berbasis komputer

- Solusi harus ekonomis dan dapat diterima secara sosial maupun secara politik.
- Solusi yang ekonomis adalah sistem komputerisasi yang memberikan pelayanan dan informasi yang sama dengan sistem lama tetapi membutuhkan waktu dan personal yang lebih sedikit dalam pengoperasiannya.
- Sistem baru mungkin akan mengurangi keterlibatan personal; hal ini mungkin akan menimbulkan dampak sosial, bahkan politik.

3. Identifikasi sumber daya yang tersedia.

Tiga subsistem dalam sistem komputerisasi adalah : perangkat keras, perangkat lunak, dan personal. Identifikasi juga keterkaitan antar ketiga subsistem tersebut.

4. Tetapkan sasaran dan persyaratan, baik untuk proses pengembangan maupun produk.

Sasaran adalah tujuan yang ingin dicapai. Sasaran digunakan sebagai dasar bagi kerangka kerja proyek pengembangan perangkat lunak, baik dalam proses pengembangan maupun untuk produk kerja.

- Sasaran dapat dinyatakan secara kualitatif maupun kuantitatif.
Contoh :
- ◆ proses (kualitatif) : harus meningkatkan keterampilan personal
- ◆ proses (kuantitatif) : sistem harus selesai dalam 12 bulan
- ◆ produk (kualitatif) : sistem harus membuat pekerjaan *user* *maikin menarik*
- ◆ produk (kuantitatif) : sistem harus mengurangi biaya transaksi sebesar 25%.

Persyaratan menetapkan spesifikasi kemampuan sistem dalam menyelesaikan masalah.

- Persyaratan mencakup aspek-aspek : fungsional, kinerja, perangkat keras, perangkat lunak, personal, dan pengantarmukaan.
- Kalau memungkinkan, nyatakan persyaratan secara kuantitatif untuk menghindari ketidakjelasan dan perselisihan antara pengembang dengan pelanggan.

Contoh persyaratan kuantitatif :

- Tanggapan maksimum terhadap *interrup* adalah 0.25 detik
- *Space* maksimum yang digunakan sistem adalah 50 KB memori utama, tidak termasuk *space* untuk *file-file* buffer
- sistem harus dapat beroperasi dengan kemampuan 95% ketika dioperasikan selama 24 jam penuh

Contoh persyaratan kualitatif :

- ◆ sistem harus mempunyai tanggapan yang baik
- ◆ sistem harus hemat dalam penggunaan memori utama
- ◆ keandalan sistem harus 99%

- Sasaran dan persyaratan dapat juga ditetapkan melalui atribut-atribut kualitas yang harus dimiliki sistem, di antaranya : *portability* (*S/W tidak bergantung mesin*), *reliability* (*kemampuan program melakukan fungsi yang diinginkan*), *efficiency* (*menggunakan sumber daya minimal*), *accuracy* (*ukuran besarnya error*), *robustness/integrity* (*kemampuan bekerja dengan baik walau mendapat input yang tidak benar*), *correctness* (*hasil sesuai dengan yang diharapkan*).

5. Tetapkan kriteria penerimaan sebuah sistem

Kriteria harus dinyatakan sedemikian rupa sehingga tidak akan menimbulkan perselisihan antara pengembang dan pelanggan. Kriteria harus dapat diverifikasi dengan suatu metoda baku seperti : peninjauan langsung, analisa, atau serangkaian uji, terhadap produk yang dihasilkan.

II. Pengembangan Strategi Solusi

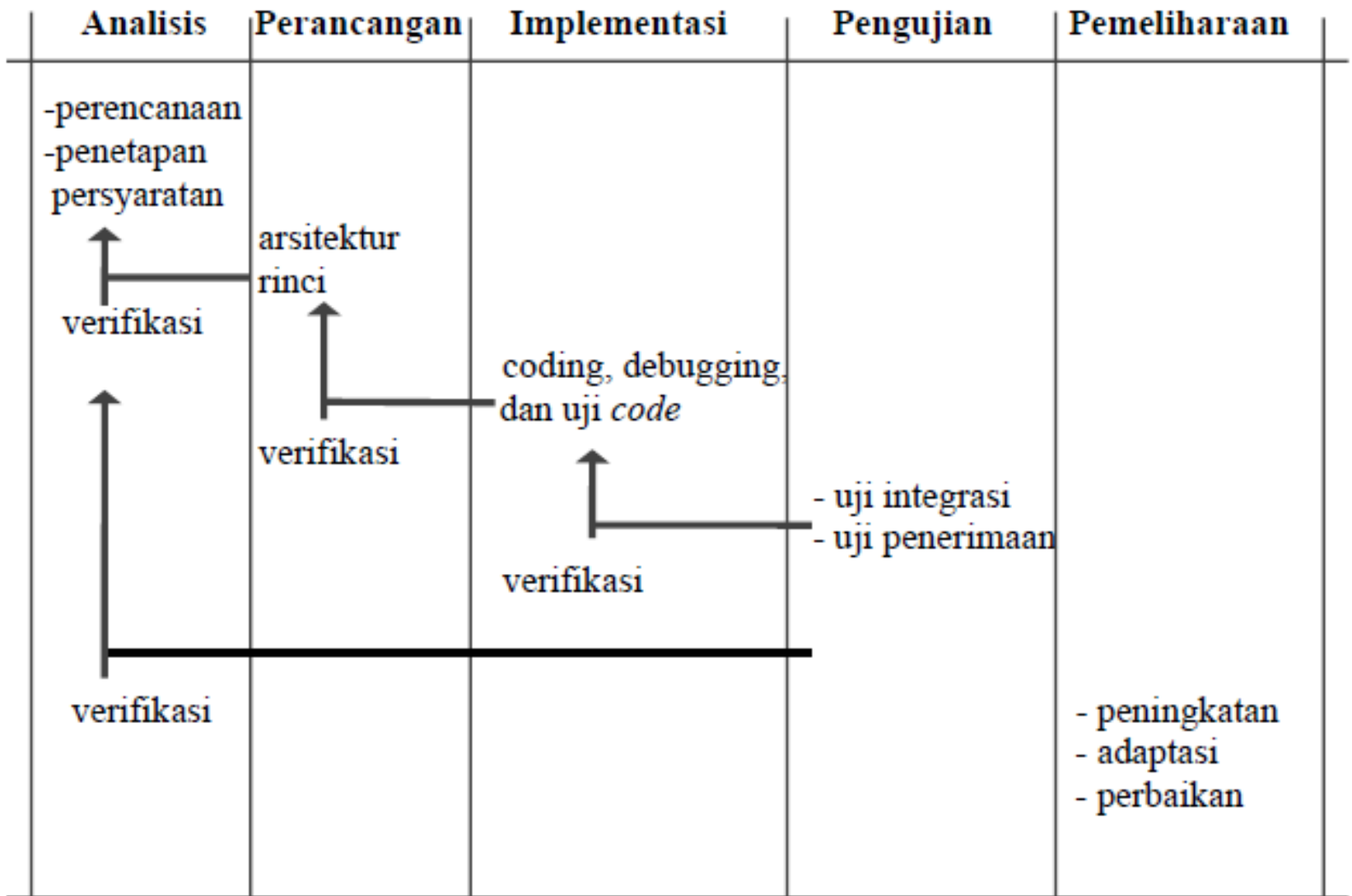
- Kecenderungan untuk menerima begitu saja solusi pertama yang terlintas di benak kita adalah masalah utama dalam perancangan perangkat lunak. Ini tidak memberi peluang terhadap solusi lain yang sebenarnya masih mungkin untuk dipertimbangkan.
- Kembangkan strategi solusi. Strategi bukan merupakan solusi rinci tetapi pernyataan umum tentang sifat-sifat dari solusi yang mungkin.

- Langkah-langkah pengembangan strategi solusi adalah sebagai berikut :

1. Uraikan beberapa strategi solusi tanpa memperhatikan batasan-batasan apapun
2. Adakan studi kelayakan terhadap setiap strategi. Perhatikan bahwa *an unreasonable idea will lead to other ideas, some of which may be very reasonable.*
3. Rekomendasikan sebuah strategi solusi, beri catatan mengapa solusi lain ditolak
4. Buat sebuah daftar prioritas karakteristik produk. Daftar ini penting jika kondisi tidak memungkinkan untuk mengimplementasikan seluruh kemampuan produk yang diinginkan seperti yang telah ditentukan sebelumnya.

III. Perencanaan Proses Pengembangan

1. Tentukan sebuah model *life-cycle* dan *struktur organisasi proyek*.
2. Rencanakan konfigurasi manajemen, jaminan kualitas, dan kegiatan validasi
3. Tentukan tools setiap fase proyek, serta teknik-teknik dan notasi yang digunakan
4. Tetapkan perkiraan biaya untuk pengembangan sistem
5. Tetapkan jadwal pengembangan
6. Tetapkan perkiraan susunan personalia proyek
7. Tetapkan perkiraan sumber daya sistem komputaerisasi yang diperlukan untuk mengoperasikan dan memelihara sistem
8. Siapkan daftar istilah
9. Identifikasi sumber-sumber informasi dan jadikan sebagai acuan proyek



Life cycle mode fase dari sebuah perangkat lunak

Subfase perencanaan menghasilkan dua produk :
System Definition dan Project Plan.

Format kedua produk adalah sebagai berikut :

Format *System Definition*

- Bab 1 : Pendefinisian masalah
- Bab 2 : Justifikasi sistem
- Bab 3 : Sasaran sistem dan proyek
- Bab 4 : Batasan sistem dan proyek
- Bab 5 : Fungsi yang harus disiapkan
(H/W, S/W, personal)
- Bab 6 : Karakteristik pengguna
- Bab 7 : Lingkungan pengembangan/
operasi/pemeliharaan
- Bab 8 : Strategi solusi
- Bab 9 : Prioritas gambaran sistem
- Bab 10: Kriteria penerimaan sistem
- Bab 11: Sumber informasi
- Bab 12: Daftar istilah

Format *Project Plan*

- Bab 1 : Model *life cycle* : terminologi, tonggak penting,
produk kerja
- Bab 2 : Struktur organisasi : struktur manajemen/
struktur team, gambaran kerja
- Bab 3 : Perkiraan personal & persyaratan sumber daya
- Bab 4 : Jadwal awal pengembangan
- Bab 5 : Perkiraan awal biaya
- Bab 6 : Pengawasan proyek dan mekanisme kontrol
- Bab 7 : Alat bantu dan teknik yang digunakan
- Bab 8 : Bahasa pemrograman
- Bab 9 : Persyaratan pengujian
- Bab 10: Dokumen pendukung yang diperlukan
- Bab 11: Cara demonstrasi
- Bab 12: Jadwal dan materi pelatihan
- Bab 13: Rencana pemasangan (instalasi)
- Bab 14: Pokok perhatian dalam pemeliharaan
- Bab 15: Metoda dan waktu pengantaran
- Bab 16: Metoda dan waktu pembayaran
- Bab 17: Sumber informasi

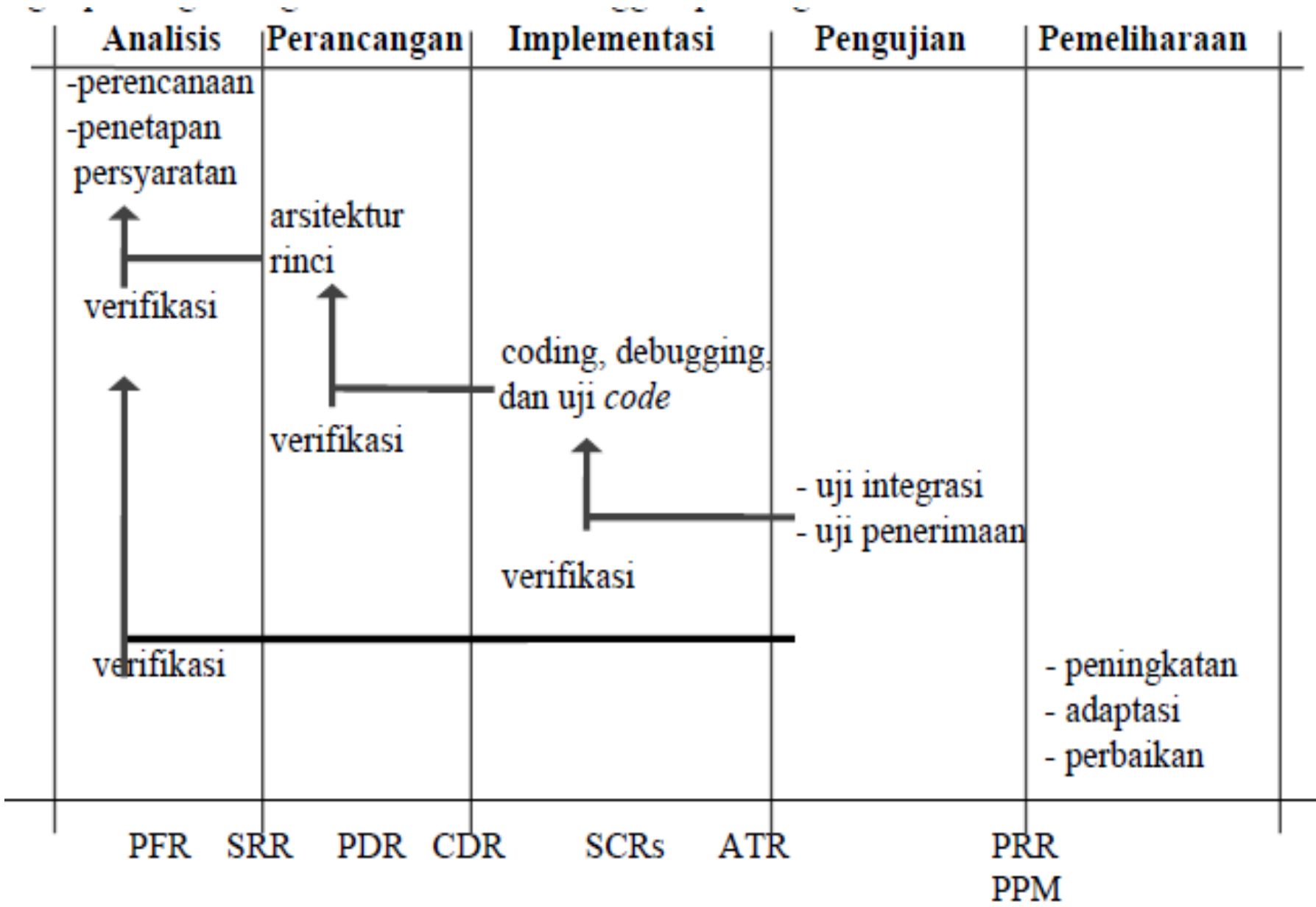
Subfase penetapan persyaratan menghasilkan sebuah produk : *Software Requirements Specifications*. Format produk ini adalah sbb :

Format *Software Requirements Specifications*

- Bab 1 : Gambaran dan penjelasan ringkasan produk
- Bab 2 : Lingkungan pengembangan, pengoperasian, dan pemeliharaan
- Bab 3 : Pengantarmukaan eksternal dan aliran data : format tampilan, *user command*, DFD, kamus data
- Bab 4 : Persyaratan fungsional : fungsi-fungsi yang diinginkan
- Bab 5 : Persyaratan kinerja : tanggapan, waktu proses
- Bab 6 : Penanganan kesalahan : aksi dan pesan yang harus dilakukan sebagai tanggapan atas input atau situasi yang tidak dikehendaki produk
- Bab 7 : Subset permulaan dan prioritas implementasi : 'versi' awal produk
- Bab 8 : Perkiraan modifikasi dan peningkatan
- Bab 9 : Kriteria penerimaan
- Bab 10 : Petunjuk dan panduan perancangan
- Bab 11 : Index acuan
- Bab 12 : Daftar istilah

- Fase perancangan melakukan identifikasi terhadap komponen perangkat lunak (fungsi, arus data, penyimpanan data), hubungan antar komponen, struktur perangkat lunak (dekomposisi menjadi modul-modul dan pengatarmukaannya). Fase ini menghasilkan arsitektur rinci, terutama dalam bentuk algoritma-algoritma.
- Fase implementasi adalah terjemahan langsung arsitektur rinci ke dalam bahasa pemrograman tertentu.

- Subfase uji integrasi melakukan pengujian terhadap semua modul dan pengantarmukaan sehingga pada level sistem dapat beroperasi dengan benar
- Subfase uji penerimaan melakukan baerbagi pengujian mengacu kepada berbagai persyaratan yang telah ditentukan.
- Kegiatan yang meliputi fase pemeliharaan adalah : peningkatan kemampuan, adaptasi terhadap lingkungan pemrosesan, dan melakukan berbagai koreksi atas kesalahan yang terjadi
- Penilaian kemajuan proyek akan lebih mudah jika pada model fase tersebut ditetapkan beberapa tonggak penting (*milestone*) yang pada setiap fase atau antar setiap dua fase yang berurutan. Berikut ini *Life cycle mode fase* dari sebuah perangkat lunak yang dilengkapi dengan kegiatan *review dan tonggak penting* :



Review	Produk Kerja yang direview
PFR (<i>Product Feasibility Review</i>)	<i>System Definition</i>
	<i>Project Plan</i>
SRR (<i>Software Requirements Review</i>)	Spesifikasi persyaratan perangkat lunak
	<i>User's Manual</i> awal
	Rencana awal verifikasi
PDR (<i>Preliminary Design Review</i>)	Dokumen disain arsitektur
CDR (<i>Critical Design Review</i>)	Spesifikasi disain rinci
SCR (<i>Source Code Review</i>)	Penelusuran dan Pemeriksaan <i>source code</i>
ATR (<i>Acceptance Test Review</i>)	Rencana uji penerimaan
PRR (<i>Product Release Review</i>)	Semua produk kerja sebelumnya
PPM (<i>Project Post-Mortem</i>)	Catatan umum pelaksanaan proyek

Kasus :

- Dalam dunia yang makin kompetitif ini penggunaan model-model pemasaran baru dipercaya mampu mendorong minat orang untuk membeli produk yang ditawarkan. Penggunaan *voucher* belanja telah menjadi salah satu cara bagi perusahaan untuk mempromosikan produk-produknya dan menarik minat calon pembeli untuk membeli. Salah satu kendala yang dihadapi dalam penggunaan *voucher* belanja ini adalah pada distribusi *voucher* kepada calon pembeli dan pengelolaan *voucher* oleh calon pembeli tersebut. *Mobile voucher* mencoba menjawab permasalahan tersebut dengan menghadirkan *voucher* belanja elektronik yang dapat didistribusikan melalui jaringan telepon seluler dan disimpan oleh calon pembeli sebagai teks SMS.

- Sistem *mobile voucher* terdiri dari 2 bagian, yaitu *voucher management System* dan *customer wallet*. *Voucher management System* adalah sistem komputer yang mengelola *voucher*, terdiri dari sebuah server yang terhubung dengan SMS gateway, bertugas untuk mendistribusikan *voucher* belanja. *Customer Wallet* adalah aplikasi Java SIM Card yang berfungsi untuk memvalidasi dan memverifikasi *voucher* yang dikirimkan melalui SMS oleh *Voucher Management System* berdasarkan kode keamanan tertentu.
- Proyek ini bertujuan untuk membuat aplikasi *Customer Wallet*, sebagai bagian dari keseluruhan aplikasi *Mobile Voucher*. Produk yang dihasilkan adalah perangkat lunak *Customer Wallet* berbasis *Java Smart Card API*.
- Proyek ini dilaksanakan dengan tahapan-tahapan: pendefinisian kebutuhan, desain, koding, dan pengujian, dengan jangka waktu pengerjaan selama 2 bulan dan biaya sebesar Rp. 99.999.999.