

## **PENERAPAN PEMELIHARAAN SYSTEMS MAINTENANCE LIFE CYCLE BANK SWASTA NASIONAL BERDASARKAN IEEE DAN ISO**

**Eko Rudi Suteja<sup>1</sup> , Lulu Chaerani Munggaran<sup>2</sup>**

Program Studi Magister Manajemen Sistem Informasi Universitas Gunadarma<sup>1</sup>

Program Studi Magister Manajemen Sistem Informasi Universitas Gunadarma<sup>2</sup>

rudi.invista@gmail.com , lulu.staff@gunadarma.ac.id

### **Abstrak**

Proses pemeliharaan sistem perangkat lunak di dalam organisasi keuangan yaitu perbankan menjadi bagian terpenting dalam mendukung kebutuhan bisnis perbankan yang selalu berubah dari waktu ke waktu. Bank sebagai organisasi keuangan tentunya memerlukan aturan khusus yang digunakan dalam mengawasi, mengatur dan mengendalikan seluruh kebutuhan pemeliharaan sistem perangkat lunak. Bank memerlukan aturan standard yang harus ditaati dan dipenuhi bagi semua pihak yang terlibat di dalam proses pemeliharaan sistem perangkat lunak. Systems Maintenance Life Cycle (SMLC) adalah aturan standard pada sebuah Bank Swasta Nasional yang harus dipatuhi dan dijalankan sesuai dengan aturan yang berlaku. Penetapan aturan pemeliharaan terhadap proses Systems Maintenance Life Cycle (SMLC) berdasarkan pada standard aturan ISO dan IEEE

**Kata Kunci:** *Pemeliharaan Perangkat Lunak, SMLC, ISO, IEEE, Rekayasa Perangkat Lunak.*

**Abstract:** *Process of maintenance software systems in financial organizations especially banking become the most important thing to support demands of banking business requirements which always changes several times depend on business requirements by the time. Bank as financial organization must have special policy that used to monitor, manage and control all needs in maintenance software systems. Bank needs standard policy that must be allowed to all employees who participates in process Systems Maintenance Life Cycles (SMLC). SMLC is set of standard organization policy in Private National Bank that must be allowed for all stakeholders who participate to do maintenance software systems in banking operations. Standard of make the policy of SMLC Process is ISO and IEEE*

**Keywords:** *Maintenance Software, CMMI, SMLC, ISO, IEEE, Software Engineering.*

### **I. PENDAHULUAN**

#### **Pengertian Pemeliharaan Systems Maintenance Life Cycle (SMLC)**

Kebutuhan untuk memodifikasi suatu potongan perangkat lunak setelah penyerahan telah menjadi bagian dari pemrograman sejak permulaan mesin hitung elektronik. Sistem perangkat lunak yang sukses, akan mengalami penurunan dengan berjalannya waktu. Suatu proporsi yang utama tentang perubahan digunakan untuk menemukan perubahan kebutuhan pengguna yang terus menerus. Suatu program yang digunakan dalam suatu lingkungan dunia nyata perlu diubah atau setidaknya semakin bermanfaat bagi lingkungan sedikit demi sedikit (Lehman, 1984). Perubahan yang penting juga

bersumber dari penyesuaian perangkat lunak untuk saling berhubungan dengan kesatuan eksternal, mencakup pengguna-pengguna, organisasi, dan sistem tiruan. Sesungguhnya, perangkat lunak tidak terbatas dengan bagian yang mudah. Oleh karena itu, bagian tersebut sering diubah dalam suatu sistem (Brooks, 1987). Pemeliharaan perangkat lunak atau Systems Maintenance Life Cycle (SMLC) adalah suatu aktivitas yang sangat luas yang sering digambarkan mencakup semua pekerjaan yang dibuat di suatu sistem perangkat lunak setelah perangkat lunak beroperasi, hal ini meliputi pengoreksi kesalahan, peningkatan, penambahan, dan penghapusan kemampuan, adaptasi dalam perubahan kebutuhan data di lingkungan operasi, peningkatan

dari pencapaian, kemampuan, atau kualitas yang lain (Martin, 1983). Definisi SMLC dari IEEE adalah sebagai berikut (IEEE Std. 610.12, 1990): Pemeliharaan perangkat lunak adalah proses dari pemodifikasian suatu komponen atau sistem perangkat lunak setelah penyerahan kepada suatu kesalahan yang benar, peningkatan pencapaian atau atribut yang lain, atau penyesuaian kepada suatu lingkungan yang berubah. Definisi ini mencerminkan pandangan umum bahwa pemeliharaan perangkat lunak adalah suatu aktivitas penyerahan. Hal itu bermula ketika suatu sistem dirilis ke pengguna atau pelanggan dan meliputi semua aktivitas yang menjaga agar sistem tetap beroperasi dan memenuhi kebutuhan pengguna. Organisasi sangat memerlukan pendekatan siklus hidup perangkat lunak untuk menangani dan mengubah sistem perangkat lunak, suatu pendekatan untuk meneliti semua aspek dari pengembangan proses yang mengarah pada pemeliharaan (Osborne dan Chikofsky, 1990). Pemeliharaan perangkat lunak mencakup keseluruhan dari aktivitas yang diperlukan yang menyediakan dukungan penghematan biaya bagi suatu sistem perangkat lunak. Aktivitas dilakukan sepanjang langkah prapenyerahan seperti halnya langkah penyerahan utama. Aktivitas prapenyerahan meliputi perencanaan untuk operasi pascapenyerahan, keterdudukan, dan penentuan logistik. Aktivitas pascapenyerahan meliputi modifikasi perangkat lunak, pelatihan, dan operasi bantuan (Pigoski, 1997).

**Kategori-Kategori Pemeliharaan Perangkat Lunak**

Menurut Lientz dan Swanson (1980) pemeliharaan terbagi ke dalam 3 komponen yaitu pemeliharaan korektif (*corrective*), adaptif (*adaptive*), dan perfektif (*perfective*). Pemeliharaan korektif meliputi semua perubahan yang dibuat dengan mencabut semua secara nyata kesalahan perangkat lunak. Pemeliharaan adaptif meliputi perubahan yang diperlukan sebagai konsekuensi dari beberapa mutasi di lingkungan tempat sistem harus beroperasi, misalnya mengubah suatu sistem untuk membuat sistem tersebut dapat berjalan pada platform perangkat keras yang baru, sistem operasi, DBMS, atau jaringan. Pemeliharaan perfektif (penyempurnaan) mengacu pada perubahan yang dimulai dari permintaan pengguna, contohnya

pemasukan atau penyisipan data, penghapusan data, perluasan, peningkatan kerj, pemodifikasian fungsi dan kemudahan penggunaan.

Operasi pemeliharaan seharusnya tidak menurunkan keandalan dan struktur dari sistem utama, tidak juga menurunkan kemampuan pemeliharaannya. Jika tidak ada perubahan pada masa depan, pemeliharaan akan menjadi semakin sulit dan mahal dalam penerapannya (Parnas, 1994).

IEEE menggambarkan kategori Swanson dan Lientz, yaitu pemeliharaan korektif, adaptif, dan perfektif, dan menambahkan pemeliharaan darurat sebagai kategori keempat. Definisi dari IEEE adalah sebagai berikut (IEEE Std. 1219-1998, 1998):

1. Pemeliharaan korektif: modifikasi yang reaktif dari suatu produk perangkat lunak yang dilakukan setelah penyerahan untuk mengoreksi kesalahan yang ditemukan,
2. Pemeliharaan adaptif: modifikasi dari suatu produk perangkat lunak yang dilakukan setelah penyerahan untuk menjamin suatu program komputer dapat dipakai dalam lingkungan yang mengubah atau diubah,
3. Pemeliharaan perfektif: modifikasi dari suatu produk perangkat lunak yang dilakukan setelah penyerahan untuk meningkatkan kemampuan pencapaian atau pemeliharaan, dan
4. Pengoreksian dalam keadaan darurat: pemeliharaan yang dilakukan tanpa dijadwalkan untuk menjaga suatu operasional sistem tetap beroperasi normal.
- 5.

## II. METODE

### Proses Pemeliharaan System Maintenance Life Cycle (SMLC)

IEEE dan ISO menunjukkan beberapa pemeliharaan perangkat lunak, yang pertama dilakukan adalah dengan sebuah standar yang spesifik dan yang selanjutnya menjadi bagian dari standarnya pada proses siklus hidup (ISO/IEEC

12207, 1995). Proses pemeliharaan SMLC menurut standard IEEE-1219 adalah dengan mengidentifikasi masalah atau permohonan modifikasi, klasifikasi, dan pemrioritasan adalah tahap yang meminta perubahan (Change Request [CR]) yang dikeluarkan oleh seorang pengguna, pelanggan, seorang pemroram atau seorang manager yang ditugaskan sebagai suatu kategori pemeliharaan, suatu prioritas, dan suatu pengidentifikasi unik. Tahapan ini juga meliputi aktivitas untuk menentukan apakah akan menerima atau menolak permintaan, dan untuk menugaskannya pada suatu batch dari jadwal modifikasi untuk implementasi. Beberapa tahapan dalam proses pemeliharaan IEEE adalah sebagai berikut:

1. Analisis

Tahap ini merencanakan persiapan untuk perancangan, implementasi, pengujian, dan penyerahan. Analisis dilakukan pada dua tingkatan, yaitu analisis kelayakan dan analisis yang terperinci. Analisis kelayakan mengidentifikasi solusi alternatif dan menilai biaya-biaya dan dampaknya, sedangkan analisis yang terperinci menggambarkan kebutuhan untuk modifikasi, merencanakan suatu strategi pengujian, dan mengembangkan sebuah rencana implementasi

2. Perancangan

Modifikasi untuk sistem benar-benar dirancang pada tahap ini. Tahap ini memerlukan penggunaan semua proyek dokumentasi dan sistem berjalan, basis data dan perangkat lunak yang ada, dan keluaran dari tahap analisis. Aktivasnya meliputi identifikasi dari modul perangkat lunak yang dipengaruhi, modifikasi dari dokumentasi modul perangkat lunak, penciptaan dari kasus-kasus pengujian untuk rancangan yang baru, dan identifikasi dari pengujian regresi

3. Implementasi

Tahap ini meliputi aktivitas pengkodean dan pengujian unit, integrasi dari kode

yang dimodifikasi, integrasi dan pengujian regresi, analisis risiko, dan peninjauan ulang. Tahap ini juga meliputi suatu *test-readiness review* untuk menilai kesiapan sistem dan pengujian regresi.

4. Pengujian sistem

Tahap ini digunakan secara bertahap dengan keseluruhan sistem yang diuji untuk memastikan bahwa kebutuhan utamanya dimodifikasi. Sebagai tambahan terhadap fungsionalitas dan pengujian antarmuka, tahapan ini meliputi pengujian regresi untuk mengesahkan ada tidaknya kesalahan baru yang telah ditambahkan. Terakhir, tahapan ini bertanggungjawab atas kesiapan pembuktian untuk menerima pengujian.

5. Penerimaan Pengujian

Tingkat pengujian ini mempunyai kaitan dengan sistem yang terintegrasi secara penuh dan melibatkan pengguna, pelanggan atau pihak ketiga yang ditunjuk oleh pelanggan. Penerimaan pengujian meliputi pengujian fungsionalitas, pengujian interoperabilitas, dan pengujian regresi.

6. Penyerahan

Pada tahap ini, sistem yang dimodifikasi diluncurkan untuk operasi dan instalasi. Tahap ini meliputi aktivitas yang memberitahukan kepada pengguna, melakukan pelatihan dan instalasi, dan menyiapkan versi arsip untuk *backup*.

Standar IEEE-1219 menjelaskan secara terperinci terkait dengan pemeliharaan perangkat lunak, sedangkan standar ISO-12207 berhadapan dengan keseluruhan proses yang terdiri dari siklus hidup perangkat lunak. Beberapa tahapan dalam proses pemeliharaan ISO-12207 adalah sebagai berikut:

1. Implementasi Proses

Aktivitas ini meliputi tugas untuk rencana pengembangan dan prosedur untuk pemeliharaan perangkat lunak,

menciptakan prosedur untuk penerimaan, perekaman, menjajaki permintaan pemeliharaan, dan menetapkan suatu antarmuka organisasional dengan proses manajemen konfigurasi. Implementasi proses dimulai dari awal siklus hidup sistem. Pigoski menyatakan bahwa rencana pemeliharaan harus disiapkan secara paralel dengan rencana pengembangan. Aktivitas pemeliharaan memerlukan definisi dari lingkup pemeliharaan dan analisis dan identifikasi alternatif, termasuk yang diserahkan pada pihak ketiga. Aktivitas ini juga meliputi susunan kepegawaian

2. Analisis modifikasi dan masalah  
Aktivitas ini mempunyai kaitan dengan analisis dari permintaan pemeliharaan, salah satu dari laporan masalah atau permintaan modifikasi, untuk menggolongkannya, untuk menentukan lingkungannya dalam hal ukuran, biaya-biaya, dan waktu yang diperlukan, dan untuk menilai kepentingannya. Direkomendasikan bahwa organisasi pemeliharaan dapat memverifikasi permintaan.
3. Implementasi Modifikasi  
Aktivitas ini memerlukan identifikasi dari materi yang perlu untuk dimodifikasi dan mengharapkan proses pengembangan secara benar menerapkan perubahan. Kebutuhan tambahan dari proses pengembangan mempunyai kaitan dengan prosedur pengujian untuk memastikan bahwa kebutuhan baru yang telah dimodifikasi telah tepat diterapkan dan kebutuhan yang tidak dimodifikasi tidak memengaruhi kebutuhan yang asli.
4. Tinjauan ulang pemeliharaan  
Tugas dari aktivitas ini disediakan untuk menilai integritas dari sistem yang

dimodifikasi dan ketika organisasi pemeliharaan mendapat persetujuan untuk menyelesaikan permintaan pemeliharaan tersebut. Beberapa proses pendukung mungkin dilibatkan, termasuk proses jaminan kualitas, proses verifikasi, proses validasi, dan proses tinjauan ulang gabungan.

#### 5. Migrasi

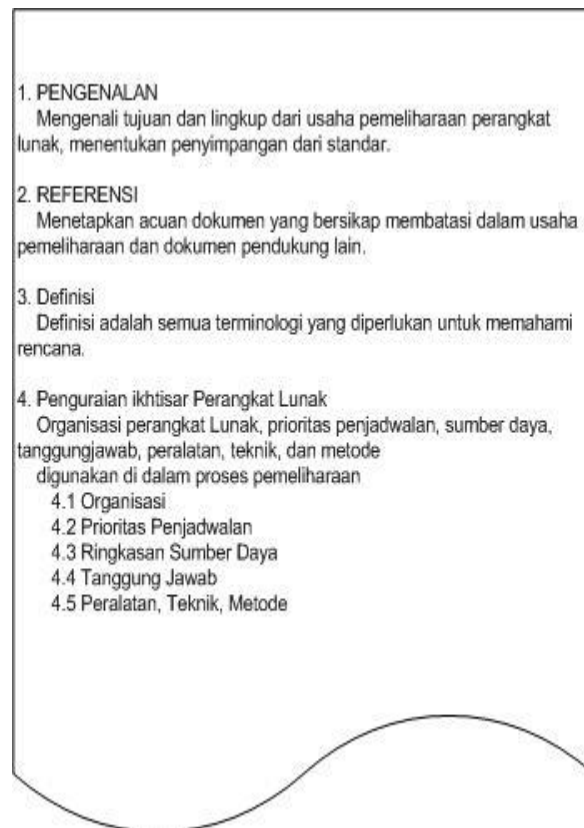
Aktivitas ini terjadi ketika sistem perangkat lunak dipindahkan dari suatu lingkungan ke lingkungan yang lain. Pemindahan tersebut diperlukan agar pengguna bisnis diberikan lingkungan yang berbeda dengan lingkungan pengembang di sisi IT. Fungsinya agar dapat dilakukan operasi paralel antara lingkungan di sisi IT dengan lingkungan bagi pengguna bisnis agar dapat ditinjau dan dinilai dampak dari perpindahan lingkungan yang baru. Sebagai contoh pada saat tim pemelihara di sisi IT telah selesai menjalankan aktivitas pengkodean dan pengujian terhadap modifikasi di lingkungan pemeliharaan Systems Integration Testing (SIT). Maka tim IT akan mengimplementasikan di lingkungan pengguna yang dinamakan lingkungan User acceptance Testing (UAT), sehingga pihak pengguna dari sisi bisnis dapat melakukan pengujian terhadap hasil modifikasi yang telah selesai dikerjakan oleh pihak IT, kemudian jika hasil pengujian sudah sesuai dengan yang diharapkan oleh pengguna maka pengguna di sisi bisnis akan melakukan validasi dan persetujuan yang menyatakan bahwa hasil pengujian UAT sudah sesuai dan benar. Atas persetujuan tersebut maka pihak IT sebagai penanggungjawab pemelihara sistem perangkat lunak akan menjalankan proses implementasi ke lingkungan *production* agar kebutuhan perubahan atas produk perangkat lunak dapat

dinikmati oleh pengguna bisnis dalam menjalankan aktivitas bisnisnya.

6. Perangkat lunak yang masanya habis  
Aktivitas pemeliharaan yang terakhir terdiri dari pengunduran diri suatu sistem perangkat lunak yang memerlukan pengembangan dari suatu rencana *retirement* dan pemberitahuan kepada pengguna.

Pemeliharaan adalah proses dari perancangan dan pemeliharaan suatu lingkungan ketika individu bekerja bersama di dalam kelompok menyelesaikan secara efisien tujuan yang dipilih (Wehrich, 1997). Tujuan utama pemeliharaan adalah untuk menyediakan dukungan yang hemat biaya pada suatu sistem perangkat lunak selama jangka hidup keseluruhannya. Manajemen mempunyai kaitan dengan produktivitas dan kualitas yang menyiratkan efisiensi dan keefektivan. Manajemen terdiri dari lima fungsi terpisah. Fungsi tersebut terdiri dari *planning*, *organizing*, *staffing*, *leading*, dan *controlling*. *Planning* terdiri dari pemilihan sasaran dan misi, dan lebih dulu menetapkan suatu tindakan untuk memenuhinya. Komitmen dari sumber daya material dan manusia, dan penjadwalan dari tindakan termasuk aktivitas yang paling kritis pada fungsi ini. *Organizing* adalah fungsi manajemen yang menetapkan suatu struktur yang disengaja tentang peran pihak-pihak yang mengisi suatu organisasi. Fungsi ini memerlukan pengaturan hubungan antarperan, mengabdikan tanggungjawab, dan otoritas yang diperlukan. *Staffing* melibatkan pengisian posisi di dalam organisasi dengan pemilihan dan pelatihan orang-orang. Dua aktivitas utama dari fungsi ini adalah pengevaluasian dan penilaian personel proyek dan penyediaan untuk pemeliharaan dan pengembangan sistem perangkat lunak, yaitu peningkatan dari pengetahuan, sikap dan keterampilan. *Leading* menciptakan suatu lingkungan kerja dan sebuah atmosfer yang akan membantu dan memotivasi orang-orang sehingga mereka akan berperan untuk mencapai prestasi organisasi dan tujuan kelompok. *Controlling* mengukur kinerja yang nyata terhadap tujuan yang direncanakan dan jika terjadi hal-hal menyimpang maka akan langsung dilakukan tindakan korektif. Fungsi ini memerlukan

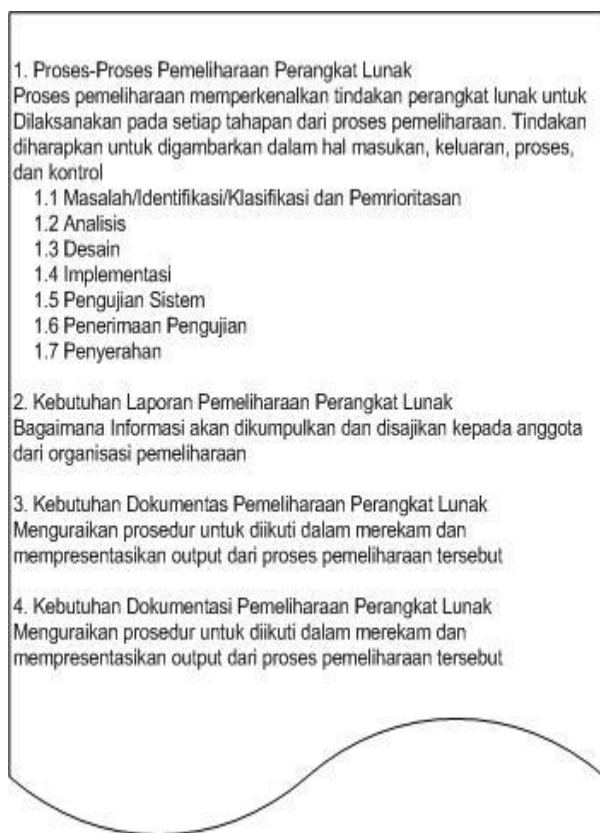
perhatian dan kedisiplinan pihak yang bertanggungjawab dalam pemeliharaan sistem perangkat lunak pada organisasi. Standard IEEE-1219 menyarankan suatu *template* untuk memandu persiapan dari suatu rencana pemeliharaan perangkat lunak yang berdasarkan pada standardnya sendiri sendiri Gambar 1 s/d Gambar 2 menunjukkan sebuah garis besar *template* dokumen SMLC.



Gambar 1 Template Dokumen Requirement

Pada *template* Gambar 1 menunjukkan bahwa dalam menjalankan proses pengumpulan kebutuhan atau requirements perlu adanya dokumen yang sesuai dengan standard organisasi agar informasi yang disampaikan oleh pengguna kepada pihak pengelola pemeliharaan sistem perangkat lunak menjadi lebih mudah dipahami dan disepakati bersama. *Template* dokumen kebutuhan ini akan

dibuat oleh pengguna sistem perangkat lunak jika mereka menginginkan adanya perbaikan, perubahan atau peningkatan kinerja perangkat lunak yang disampaikan kepada pihak pengelola pemeliharaan sistem perangkat lunak (IT). Dokumen kebutuhan ini akan menjadi landasan utama terkait latar belakang masalah yang dihadapi, batasan dalam usaha pemeliharaan, proses analisa dari sisi pengkodean dan menjadi acuan agar output yang dihasilkan oleh sistem perangkat lunak sesuai dengan kebutuhan yang sudah ditetapkan pada dokumen *requirement*.



Gambar 2 Dokumen Proses Pemeliharaan

Pada gambar 2.2 menunjukkan *template* dari dokumen proses pemeliharaan memerlukan dokumentasi secara baik. Dokumen proses pemeliharaan menyediakan informasi secara *end-to-end* mulai dari informasi kebutuhan sampai dengan tahap akhir yaitu penyerahan. Dokumentasi proses pemeliharaan harus sesuai dengan prosedur yang ditetapkan oleh organisasi.

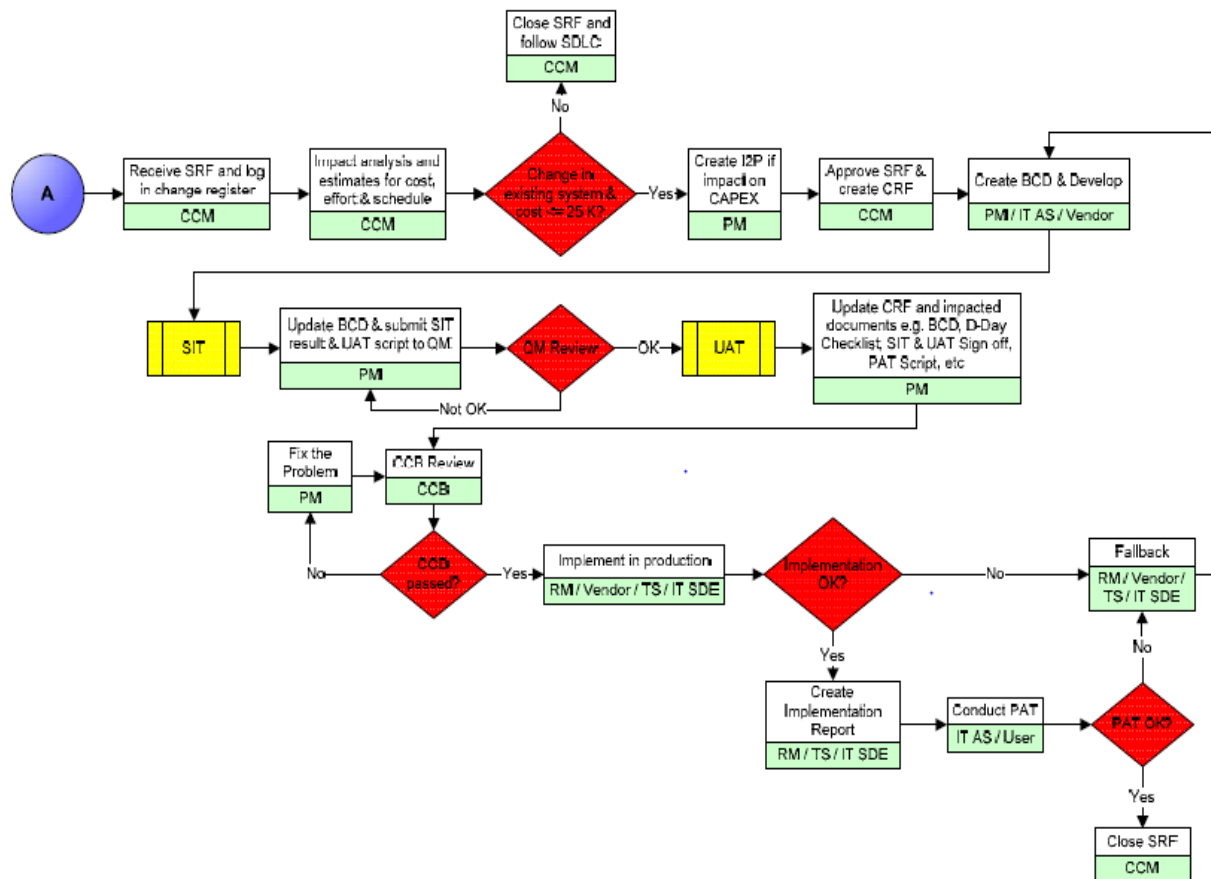
Proses pemeliharaan memerlukan dokumen:

1. Kebutuhan akan perubahan
2. Dokumentasi dari dampak perubahan yang diusulkan
3. Evaluasi dari perubahan yang diusulkan
4. Koordinasi proses perubahan, mencakup apakah perubahan telah disetujui atau tidak disetujui
5. Pengubahan produk
6. Pengubahan pada dokumentasi yang sesuai

### III. HASIL DAN PEMBAHASAN

Pada tahapan untuk pemeliharaan system atau aplikasi maka diperlukan tahapan proses Maintenance Change Request dengan mengelompokkan masing-masing aktivitas yang dimiliki oleh masing masing *Personal In Charge* (PIC). Tahapan aktivitas SMLC adalah sbb:

Pada tahapan SMLC ini diaplikasikan jika budget yang digunakan dibawah 25.000 USD. Jika pemeliharaan dilakukan dengan memakai budget diatas 25.000 USD maka diberlakukan Metode SDLC. Penggambaran alur aktivitas SMLC terangkum dalam diagram **dibawah ini**



1. Penerimaan Requirement dan Perencanaan Pemeliharaan

Pada aktivitas ini divisi IT yang mengelola perubahan pada system akan menerima SRF dari user bisnis. Dokumen SRF dapat diserahkan secara langsung atau dikirim melalui email jika pihak dari pemohon user bisnis dengan CCM *System Owner* berbeda lokasi gedung. Pada bagian ini perlu dilakukan analisa dampak, usaha, jadwal dan perkiraan apakah diperlukan biaya atau tidak bergantung pada kesepakatan kontrak dengan vendor yang memelihara system atau aplikasi tersebut. Jika hasil analisa menunjukkan bahwa

permintaan ini memerlukan biaya maka permintaan perubahan tersebut diberlakukan sebagai SDLC. Jika hasil analisa menunjukkan bahwa permintaan perubahan ini tidak membutuhkan biaya dalam pengembangannya, CCM akan memberikan persetujuan pada SRF ini dan permintaan perubahan akan diberlakukan sebagai SMLC. Setelah SRF atau Requirement Document disetujui, buat CR pada aplikasi Serena.

2. Proses Analisa dan Pengkodean (IT Development process)

Pada aktivitas ini tim IT *System Owner* yang bertanggungjawab dalam melakukan pemeliharaan aplikasi atau sistem akan melakukan analisa terhadap perubahan yang akan diimplementasikan di *Production*. Setelah proses analisa selesai dilakukan maka tim IT *System Owner* akan melakukan pengkodean pada *Environment Development*.

### 3 Proses SIT

IT Project Manager melakukan review terhadap rancangan dan naskah SIT Dan mulai untuk melakukan pengembangan sistem sesuai dengan yang ada pada script SIT. Lakukan pencatatan dan perbaikan jika masih terjadi kesalahan atau *error* melakukan SIT. Jika SIT sudah ok, buatlah formulir SIT Signoff untuk ditandatangani. SIT Signoff wajib ditandatangani oleh IT Application Manager atau CCM, Application support dan Manajer Vendor.

### 3 Proses UAT

Pada saat state di aplikasi Serena sudah berada di UAT maka Implementor melakukan *deploy object* pada server UAT didampingi oleh Appsup. Pastikan implementor mengikuti instruksi *deployment* yang dituliskan di dalam DDCL. Dan *update* jika ada perubahan dalam deployment. *User* melakukan UAT berdasarkan UAT Script yang sudah disiapkan dan disetujui. Pastikan semua langkah pada UAT Script sudah dilakukan dan catat setiap hasil aktualnya. Apabila terdapat kegagalan selama proses UAT, IT Aplikasi support dan atau *Vendor* harus melakukan perbaikan pada kegagalan yang dimaksud berdasarkan list defect yang dilaporkan oleh user. Apabila hasil keseluruhan UAT sudah sesuai, user membuat dokumen formulir persetujuan UAT. Pastikan kelengkapan dokumen UAT sudah ok dan ditandatangani oleh BPM & User Department Head.

### 3 Changes Control Board (CCB)

Setelah semua kelengkapan dokumen SIT Dan UAT sudah lengkap dan diupload ke Serena, maka IT QM akan memajukan state ke forum CCB Member untuk dilakukan pembahasan perubahan oleh Dewan Pengawas Perubahan BDI. Di dalam forum CCB bertujuan untuk memastikan CR yang terdaftar di rapat CCB telah dibahas, melakukan pemeriksaan tujuan perubahan dan waktu implementasi. Memastikan risiko implementasi telah ditinjau dan terdapat langkah mitigasinya. Memberikan persetujuan atau penolakan terhadap permintaan perubahan.

### 4 Implementasi Production

Implementasi production adalah aktivitas *deployment object* yang dilakukan pada lingkungan production. Dengan arti objek perbaikan akan diimplementasikan pada system yang sudah ada dan perbaikannya dapat dirasakan oleh user pada environment production.

### 5 Proses PAT

Proses PAT (Production Acceptable Test) adalah aktivitas pengujian pada lingkungan production. Pada proses PAT user bisnis melakukan pengujian terhadap pengajuan perubahan pada sistem yang dijalankan, apakah script yang tertuang pada PAT sudah diuji dan sudah sesuai dengan yang diharapkan. Jika testing PAT sudah OK maka PAT akan ditandatangani oleh user bisnis dan kepala bagian user bisnis tersebut

## IV. KESIMPULAN

Pada hasil dan pembahasan dapat disimpulkan bahwa aturan-aturan standard yang diterapkan oleh IEEE dan ISO sudah sepenuhnya diterapkan oleh Bank Swasta Nasional dengan baik dan benar.





**REFEREBSI:**

- [1] Simarmata, Janner (2010). *Rekayasa Perangkat Lunak*. Penerbit: Andi.  
[2] Standard Operational Procedure Bank Swasta Nasional, *Systems Maintenance Life Cycles*.

- [3] Verdi Yasin (2012). *Rekayasa Perangkat Lunak Berorientasi Objek*. Jakarta : Mitra Wacana Media  
[4] Roger S. Pressman, Bruce R. Maxim (2014) *Software Engineering: A Practitioner;s Approach*, 8th Edition, Mc Graw Hill Education