

# IMPLEMENTASI SOFTWARE DEVELOPMENT LIFE CYCLE (SDLC) DALAM PENERAPAN PEMBANGUNAN APLIKASI PERANGKAT LUNAK

Yoyok Seby Dwanoko

Sistem Informasi, Fakultas Sains dan Teknologi, Universitas Kanjuruhan Malang  
bpkyoyokseby@gmail.com

## Abstrak

Membangun sebuah aplikasi perangkat lunak diperlukan beberapa tahapan yang harus dilalui mulai dari tahapan persiapan analisa program sampai dengan perawatan perangkat lunak yang sudah dibuat. Sering sekali permasalahan dalam pembuatan perangkat lunak tidak mengikuti suatu metode atau langkah-langkah dalam membuat aplikasi perangkat lunak yang akan menjadi sebuah produk gagal atau biasanya disebut *software crisis*. Untuk menghindari *software crisis* diperlukan sebuah metode untuk mengarahkan pembangunan perangkat lunak, metode tersebut yaitu *Software Development live cycle* atau bisa disebut SDLC. Metode ini mepresentasikan tahapan pembuatan aplikasi perangkat lunak mulai dari tahapan analisis, desain, kontruksi, implementasi, testing, perawatan sistem. Dalam penelitian ini menerapkan metode SDLC pada masing-masing tahapan dengan menggunakan *Unified Modeling Language* (UML) yang akan menghasilkan sebuah *Software Requirement Specification* (SRS) atau bisa dikenal Spesifikasi Kebutuhan Perangkat Lunak (SPKL).

**Kata Kunci:** Tahapan SDLC, UML, SRS.

## PENDAHULUAN

Pembangunan sebuah aplikasi perangkat lunak sering kali kita temui pembuatan aplikasi tersebut mengalami kegagalan. Kegagalan pembangunan perangkat lunak biasanya disebut dengan *software crisis*, yang pada intinya aplikasi perangkat lunak yang dibuat tidak sesuai dengan tujuan yang ingin tercapai. Untuk menghindari *software crisis* ada beberapa metode didalam membuat sebuah aplikasi perangkat lunak, salah satunya adalah *Software Development Life Cycles* (SDLC).

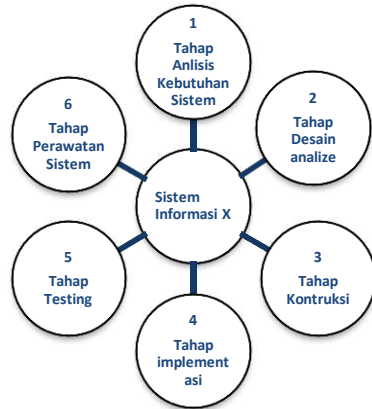
SDLC adalah tahapan-tahapan pekerjaan yang dilakukan oleh analis sistem dan programmer dalam membangun sistem informasi. Ada 6 tahapan secara umum didalam SDLC yaitu: 1. Analisis sistem, yaitu membuat analisis aliran kerja manajemen yang sedang berjalan. 2. Desain Spesifikasi kebutuhan sistem, yaitu melakukan perincian mengenai apa saja yang

dibutuhkan dalam pengembangan sistem dan membuat perencanaan yang berkaitan dengan proyek sistem. 3. Kontruksi sistem, yaitu membuat desain aliran kerja manajemen dan desain pemrograman yang diperlukan untuk pengembangan sistem informasi. 4. Implementasi sistem, yaitu tahap menjalankan sistem yang sesuai dengan fungsi masing-masing. 5. Pengujian sistem, yaitu melakukan pengujian terhadap sistem yang telah dibuat. 6. Pemeliharaan sistem, yaitu menerapkan dan memelihara sistem yang telah dibuat.

Tujuan dari penelitian ini yaitu menyusun SKPL/SRS dengan meimplentasikan metode SDLC ini sebagai panduan dalam menyusun Spesifikasi kebutuhan perangkat lunak (SKPL) atau bisanya disebut *Software Requirement Specification* (SRS). SKPL / SRS dipresentasikan dalam bentuk dokumen dan menggunakan *Unified Modeling Language* (UML) dari segi content.

## METODE

Metode yang dipakai didalam artikel jurnal ini adalah Software Development Life Cycles SDLC. Tahapan dari metode SDLC ini adalah sebagai berikut :



**Gambar 1. Tahapan Pembangunan Sistem Informasi dengan metode SDLC**

berikut adalah keterangan dari gambar 1 yaitu tahapan-tahapan SDLC sebagai berikut :

1. Tahap Analisis Kebutuhan Sistem

a. Analisis sistem (*analyze Requirements*)

Pada dasarnya kegiatan yang dilakukan pada tahap analisis sistem ada dua bagian, yaitu tahap survei pengumpulan data dan analisis terstruktur yang secara garis besar berfungsi untuk memperoleh pengertian dari permasalahan-permasalahan, efisiensi, dan pertimbangan-pertimbangan yang mengarah ke Rancang Bangun Sistem Informasi, serta mencari kendala yang dihadapi dalam sistem sehingga dapat menentukan solusi alternatif pendahuluan. Mempelajari konsep sistem dan permasalahan yang hendak diselesaikan. apakah sistem baru tersebut realistis dalam masalah pembiayaan, waktu, serta perbedaan dengan sistem yang ada sekarang.

b. Kegiatan wawancara dan Survei sistem (*survey Requirement*)

Kegiatan pada tahap survei pengumpulan data ini adalah mengumpulkan informasi dan data selengkap-lengkapnyanya. Selain melaksanakan survei untuk pengumpulan data dan informasi, dilakukan pula proses wawancara. Proses wawancara diperlukan sebagai dasar untuk mengetahui kebutuhan aktor yang terlibat dalam sistem, kebutuhan fungsional mulai dari input, proses dan output pada sistem yang dibutuhkan. Kebutuhan non fungsional juga dibutuhkan pada tahap ini seperti kebutuhan hardware, kebutuhan koneksi, bandwidth, keamanan sistem dan kebutuhan non fungsional yang lain. Studi literatur peraturan-peraturan yang terkait. Setelah hasil pengumpulan data ini diperoleh, akan didiskusikan dengan tim pengembang untuk pemaparan hasil studi guna penyesuaian, konfirmasi, dan menambahkan masukan. Pada tahap ini akan diperiksa kelengkapan tiap data yang sangat diperlukan dalam pembangunan database.

c. Analisis terstruktur

Pada tahap ini tim pengembang akan menganalisis bahan masukan yang telah diperoleh dari hasil survei. Kemudian diolah menjadi sebuah dokumen analisa terstruktur sebagai gambaran umum dari pembangunan sistem informasi yang akan buat atau dikembangkan.

d. Identifikasi masalah

Identifikasi masalah merupakan kegiatan mengidentifikasi masalah yang terjadi dalam sistem yang sedang berjalan. Dari hasil wawancara dan observasi akan menemukan beberapa identifikasi masalah yang ada pada pihak customer sehingga hasil dari indentifikasi masalah terbentuk sebuah dokumen daftar identifikasi masalah pada sistem yang akan dibuat atau dikembangkan.

e. Analisa planning

Mempelajari konsep sistem dan permasalahan yang hendak diselesaikan. apakah sistem baru tersebut realistis dalam masalah pembiayaan, waktu, serta perbedaan dengan sistem yang ada sekarang.

2. Tahap Desain (*Design Analize*)

Tahap Desain, penetapan pembenahan, dan pengembangan sistem tahap ini meliputi presentasi awal, desain konseptual, desain basis data dan sistem, desain detail input/output Sistem Informasi. Dalam sistem desain fungsi desain dan operasi dijelaskan secara rinci, termasuk tata letak layar, aturan bisnis, diagram proses dan dokumentasi lainnya. Output dari tahap ini akan menjelaskan sistem yang baru sebagai koleksi modul atau subsistem. Tahap desain diperlukan sebagai masukan awal persyaratan identifikasi dalam dokumen persyaratan disetujui. Untuk setiap persyaratan, satu set satu atau lebih elemen desain akan diproduksi sebagai hasil dari wawancara, lokakarya, dan / atau upaya prototipe.

3. Tahap konstruksi (*Design and Coding Program*)

Tahap ini yaitu melakukan coding sistem yaitu menyusun bahasa pemrograman yang dipilih misalnya yang berbasis web dengan menggunakan HTML dan PHP, membuat dan mengkoneksi database dengan sistem, membuat form sistem dan pembuatan manual book. Disamping itu juga desain Grapical User Interface(GUI). Desain GUI ini dibutuhkan untuk mepresentasikan kebutuhan sistem yang dibuat dalam bentuk form-form mulai dari input, proses dan output, bentuk grafik statistik, bentuk simulasi atau bentuk visual lainnya. Dari GUI ini nantinya akan kelihatan sebuah sistem yang dibuat.

4. Tahap implementasi sistem (*Implemention program*)

Tahap ini meliputi menjalankan / implementasi program dan uji coba /evaluasi sistem, bertujuan untuk mengetahui hasil dari kebutuhan fungsional sistem yang disusun dicocokkan

dengan hasil implementasi dalam bentuk Rancang Bangun Sistem Informasi. Tahap Implementasi Sistem secara keseluruhan sebagai berikut :

- a. Perancangan Implementasi terdiri dari tugas implementasi, jadwal implementasi, perkiraan biaya, penanggung jawab.
- b. Pengembangan dan uji coba software terdiri dari : Menentukan kebutuhan pemakai, Rencana pengembangan, Menulis program, Uji coba program, Pendokumentasian program, Meneliti pemakai, Installasi sistem
- c. Penyiapan lokasi meliputi berbagai aktifitas dan penyiapan peralatan sampai dengan lokasi dimana sistem siap digunakan.
- d. Memilih dan melatih pemakai yaitu pemakai sistem perlu dilatih agar dapat menggunakan dengan benar.
- e. Dokumentasi terdiri dari :
  - Dokumentasi pengembangan berisi penjelasan sistem , salinan output, input, file dan database, flowchart program dan hasil uji coba.
  - Dokumentasi operasi berisi penjadwalan operasi, hasil akses file, dan database, alat yang digunakan, keamanan dan penyimpanan file.
  - Dokumentasi pemakai berisi prosedur manual dan materi pelatihan.

5. Tahap Tes Program ( *test unit, integrate and test system*)

Tahap ini melakukan uji coba sistem berdasarkan pengolahan data yang real / nyata agar diperoleh hasil tes. Selain itu melakukan uji coba sistem dari segi konektifitas, fungsional sistem untuk mengetahui program bisa diaplikasikan dan mendapatkan evaluasi uji coba sistem. tiga cara uji coba :

- a. *Walk- through* yaitu melihat kembali setiap langkah dari prosedur program yang ditulis

- b. Uji coba pemrosesan transaksi yaitu menggunakan data transaksi pura pura, tujuannya untuk melihat apakah program dapat dioperasikan sesuai dengan yang diharapkan.
- c. Uji coba sesungguhnya yaitu menggunakan data transaksi sesungguhnya.

Pada tahapan ini tool yang dipakai yaitu dengan menggunakan metode black box testing yang fungsinya untuk menguji pada kebutuhan fungsional sistem.

#### 6. Perawatan perangkat lunak (*Maintenance*)

Tahap ini adalah tahapan terakhir dari Rancang Bangun Sistem Informasi yaitu mengevaluasi seluruh program dengan mengetahui data-data fungsional sistem berjalan sesuai rencana dan fungsional sistem yang belum jalan sebagaimana mestinya, tahap ini juga mengkaji untuk pengembangan berikutnya tentang Sistem Informasi yang dibuat. *Maintenance* bertujuan untuk memastikan bahwa sistem yang digunakan oleh pihak pengguna benar-benar telah stabil dan terbebas dari *error* dan *bug*. Pemeliharaan ini biasanya berkaitan dengan masa garansi yang diberikan oleh pihak pengembang sesuai dengan perjanjian dengan pihak pengguna. Lamanya waktu pemeliharaan sangat bervariasi. Namun pada umumnya sistem informasi yang kompleks membutuhkan masa pemeliharaan dari enam bulan hingga seumur hidup program aplikasi. Sistem perlu dirawat karena beberapa hal yaitu

- a. Sistem mengandung kesalahan yang dulunya belum terdeteksi, sehingga kesalahan sistem perlu diperbaiki.
- b. Sistem mengalami perubahan-perubahan karena permintaan baru dari pemakai sistem
- c. Sistem mengalami perubahan lingkungan luar.
- d. Sistem perlu ditingkatkan.

## HASIL DAN PEMBAHASAN

Untuk menyusun SKPL/SRS dengan Metode SDLC diperlukan tool sebagai bahan pembuatan dokumen. Tool itu adalah *astah community* yaitu sebuah aplikasi untuk pembuatan diagram UML meliputi diagram-diagram *use case*, *activity*, *sequence* dan *class*. Untuk desain GUI menggunakan aplikasi *GUI Designer*. Berikut hasil dari penelitian ini adalah sebagai berikut :

### 1. Tahapan Analisa Kebutuhan sistem

- a. Tahapan ini menghasilkan tabel wawancara atau observasi dengan format tabel yang berisi : kode wawancara, sumber wawancara, detail pertanyaan, harapan yang ingin dicapai hasil wawancara, keterangan, kesimpulan, dan apabila prosesnya dilakukan dengan observasi, maka tabel observasi berisi : kode observasi, tanggal observasi, target/ tempat observasi, hasil observasi, dokumen yang diambil, keterangan, kesimpulan. Contoh pengisian tabel wawancara :

**Kode wawancara :** IN01

**Sumber wawancara :** Ketua HIPAM

**Detail pertanyaan :** Data apa saja yang terkait dengan master yang akan dikelola oleh sistem ?

**Harapan yang ingin dicapai :** Mendapatkan informasi tentang Master data Input yang akan digunakan didalam sistem.

**Hasil wawancara :** data master user, master pelanggan, master jenis kas masuk, master jenis kas keluar, master data jumlah harga per meter kubik.

**Keterangan :** sistem berbasis web butuh DBMS MySQL.

**kesimpulan :** membuat data master sesuai hasil wawancara

dan memperhatikan jenis-jenis fieldnya.

Apabila proses observasi, tabel observasi berisi : kode observasi, tanggal observasi, target/ tempat observasi, hasil observasi, dokumen yang diambil, keterangan, kesimpulan. Contoh :

**Kode Observasi : OB01**

**Tanggal Observasi : 20**  
Oktober 2016

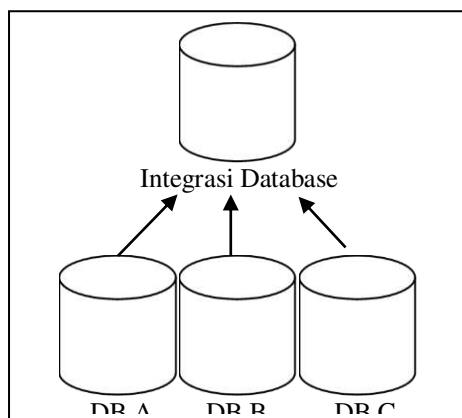
**Target/ tempat observasi :**  
Kantor HIPPAM Bumi Asih  
Kota Malang.

**Hasil observasi :**  
meidentifikasi form manual  
tagihan bulanan penggunaan  
air minum .

**Dokumen yang diambil :** Slip  
Tagihan manual penggunaan  
air minum, **Keterangan :** 2  
berkas

**Kesimpulan :** Slip fisik ini  
digunakan untuk GUI dari Slip  
Tagihan.

- b. Analisa terstruktur, tahap ini menghasilkan dokumentasi gambaran terstruktur dari konsep sistem informasi yang akan dibuat bisa berupa blok diagram sistem dan database manajemen system. Contoh :



**Gambar 2. Model Integrasi Database**

- c. Identifikasi masalah, tahap ini akan menghasilkan gambaran domain permasalahan pada lokasi penelitian yang akan dibuat berisi : gambaran atau deskripsi kondisi sistem saat ini, permasalahan domain dan solusi yang ingin dikerjakan yang perlu dijelaskan secara singkat dan jelas. Contoh :

“ HIPPAM bumi asih adalah sebuah himpunan penduduk pengguna air minum yng sehari hari mengelola manajemen dan administrasi penggunaan air minum. Didalam pengelolaannya HIPPAM ini sering mengalami beberapa kendala dan masalah antara lain : penanganan tagihan air minum ke pelanggan, tidak memiliki pengelolaan database pelanggan baru dan lama, pengelolaan arus kas dan lainnya. Dari permasalahan tersebut diatas dan meperhatikan kebutuhan pengelola HIPPAM perlu dibangun sebuah sistem informasi akuntansi HIPPAM yang digunakan untuk support manajemen pengelola berbasis web.”

## 2. Tahapan Desain

Tahapan desain ini menghasilkan dokumen desain Spesifikasi kebutuhan perangkat lunak sistem yang meliputi :

- a. Tabel fungsional sistem yang perlu disediakan dalam sistem. Tujuannya yaitu menyediakan menu dan submenu pada aplikasi yang dibangun. Ini

dari tabel kebutuhan fungsional sistem dalam bentuk tabel dengan kontennya sebagai berikut sebagai berikut : Nomor urut, Kode Fungsional, Nama fungsi, Keterangan dan Prioritas. Untuk kolom prioritas diisi keterangan HARUS berarti wajib ada di dalam sistem atau OPSIONAL berarti boleh disediakan atau tidak karena tergantung kebutuhan sistem yang akan dibangun. contoh :

**Nomor urut :** 1

**Kode Fungsional :** FNI01 (fungsional input no 1).

**Nama Fungsi :** Login

**Keterangan :** Login digunakan untuk membedakan user / aktor yang terlibat didalam sistem.

**Prioritas :** Harus.

- b. Tabel actor yang terlibat dalam sistem. Actor adalah user yang akan terlibat dalam sistem yang dibangun, tabel ini berisi : nomor, nama actor, diskripsi actor, hak akses didalam sistem yang dibangun. Sebagai contoh:

**Nomor :** 1

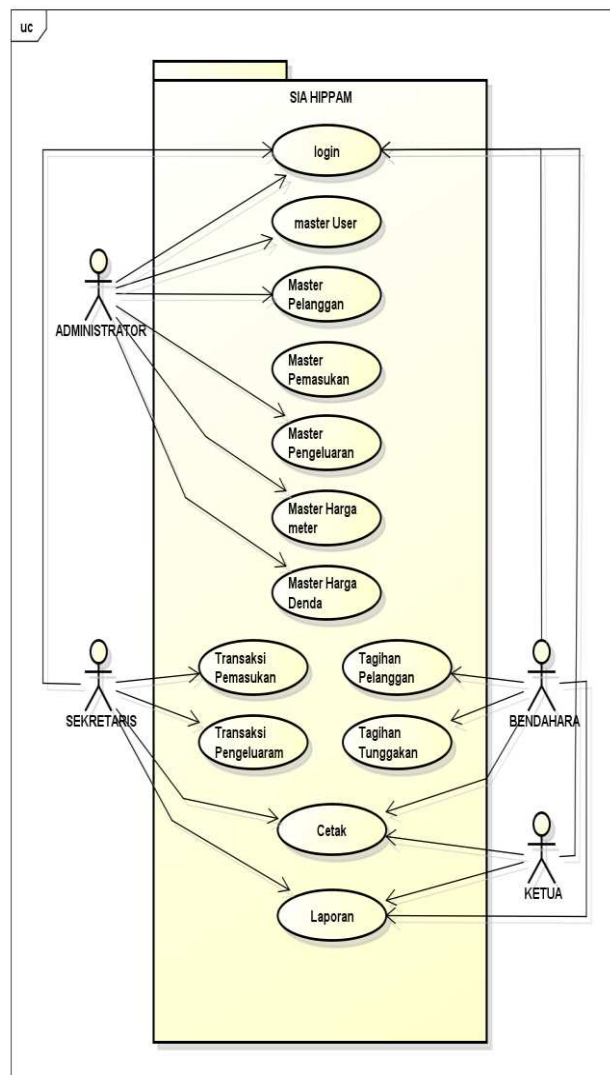
**Nama aktor :** administrator

**Diskripsi actor :** user yang bertanggungjawab pada sistem dan master yang dibutuhkan oleh sistem

**Hak akses :** master user, pegawai, master katagori barang, master supplier

- d. Membuat diagram use case utama dan sub usecase aplikasi yang dibangun dengan menggunakan *tool astah*

*community*. Pada gambar 3 dibawah ini adalah contoh diagram *Usecase* utama sebuah sistem informasi :

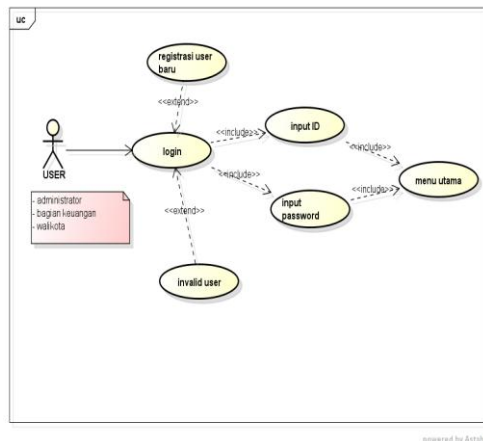


powered by Astah

**Gambar 3 : Contoh Diagram Usecase Utama Sistem Informasi HIPAM**

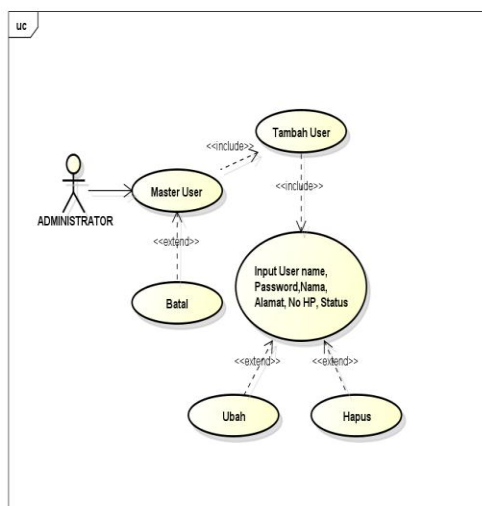
Gambar 3 mepresentasikan diagram Usecase Utama Sebuah Sistem yang dibuat berdasarkan tabel kebutuhan fungsional yang terlebih dahulu tersusun. Yang perlu diperhatikan diagram usecase utama sebuah sistem adalah aktor yang terlibat, fungsional input, proses / transaksi dan output sistem. Arah anak panah

menunjukkan hak akses aktor / user yang terlibat didalam sistem.



**Gambar 4: Contoh Diagram Sub Usecase pada login**

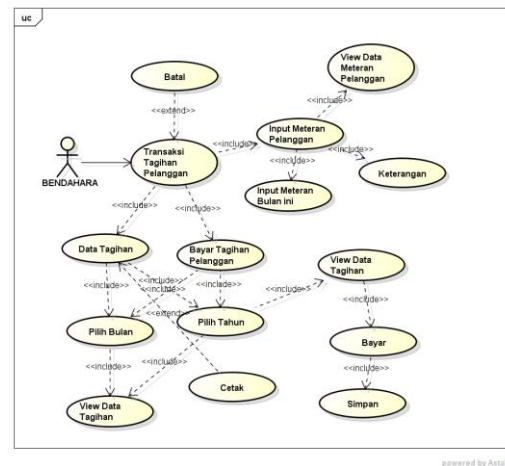
Gambar 4 mepresentasikan diagram sub usecase dari proses login. Proses ini diawali dengan meinputkan *user id* dan *password* apabila *user id* dan *password* memenuhi akan masuk ke menu utama program. Dan apabila *user id* dan *password* salah maka akan ada message invalid user.



**Gambar 5: Contoh Diagram Sub Usecase pada Master User**

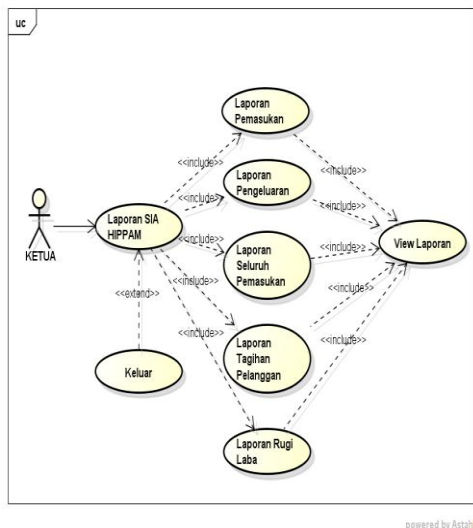
Pada gambar 5 mepresentasikan sub usecase pada master user dari model

master yang lainnya. Setiap pembuatan master yang perlu diperhatikan adalah wajib adanya simbol usecase tentang tambah, ubah, hapus dan batal hal ini karena berhubungan dengan form untuk master data user yang nantinya tersimpan di tabel user.



**Gambar 6: Contoh Diagram Sub Usecase pada transaksi**

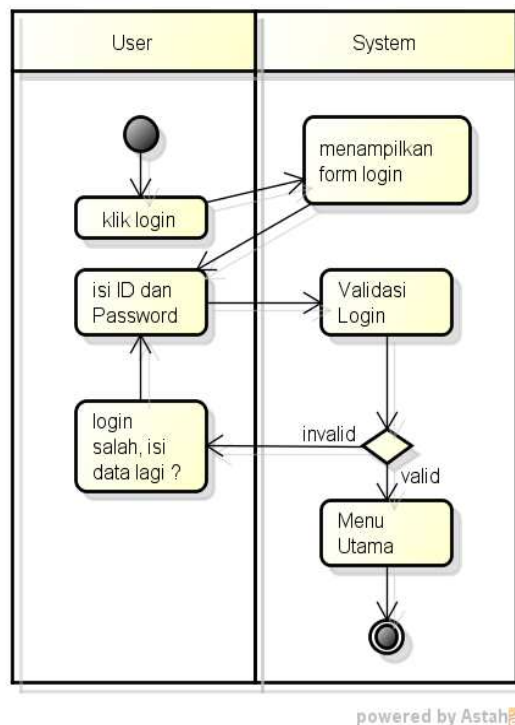
Pada gambar 6 mepresentasikan contoh diagram sub usecase pada transaksi. Hal yang perlu diperjhatikan adalah detail transaksi harus tergambarkan didalam *sub usecase* ini sehingga alur proses transaksi akan kelihatan dan memudahkan desainer dan programmer untuk mengerjakan sistemnya.



**Gambar 7: Contoh Diagram Sub Usecase pada Laporan**

Gamabr 7 mempresentasikan diagram sub usecase pada laporan. Hal yang perlu diperhatikan dalam sub usecase ini adalah kumpulan semua jenis laporan dijadikan satu dalam *sub usecase* laporan sehingga nanti *sub usecase* ini akan dibuat sebagai acuan oleh desainer GUI untuk membuat menu dan sub menu laporan.

- e. Membuat diagram Activity  
Diagram activity digunakan untuk mengetahui desain aktivitas sistem yang menggambarkan hubungan antara user dan sistem dilihat dai sisi aktivitasnya. Berikut contoh diagram activity :



**Gambar 8: Contoh Diagram Activity pada login**

Pada gambar 8 mempresentasikan diagram aktivitas sebuah fungsi login yang alurnya dilihat pada sisi user yang terlibat dan sisi sistem.

- f. Membuat sequence diagram. Diagram ini mempresentasikan proses didalam sistem yang dilakukan oleh actor dalam mengakses satu fungsi. Sequence diagram disini ada 3 class yaitu boundary class sebagai gambaran menu, Control class sebagai gambaran logic query, logic process atau transaksi dan entity class sebagai gambaran database yang diakses. Berikut contoh penggunaan sequence Diagram :





kosistensi antara desain GUI dengan Desain fisik.

ID Pelanggan	Nama	Alamat	Nomor HP
0302001	Sukardi	RT 02/03	0
0302002	Bakri	RT 02/03	0
0302003	Wawan	RT 02/03	
0302005	Agus K.	RT 02/03	
0302006	Anisa	RT 02/03	
0302007	Ida Ane	RT 02/03	

**Gambar 12. Contoh Desain Fisik**

Gambar 12 mepresentasikan desain fisik dari master data pelamggan yang berfungsi untuk membuat database master pelanggan yang diperlukan oleh sistem.

#### 4. Tahap implementasi sistem (*Implementation program*)

Tahap ini menghasilkan master data yang dibutuhkan, menjalankan transaksi atau proses dalam sistem informasi dan menampilkan laporan yang sudah dibuat pada tahapan kontruksi. Hal yang perlu diperhatikan dalam tahap implementasi ini adalah :

- Memilih user yang terlibat langsung dengan sistem.
- Menyusun Dokumen user guide dan dokumen user administrator.

#### 5. Tes Program ( *test unit, integrate and test system*)

Tahap ini melakukan uji coba sistem berdasarkan pengolahan data yang real / nyata. Metode yang dipakai untuk testing adalah metode *black box* testing yaitu metode pengujian yang uji kosentrasinya pada fungsional yang ada pada sistem informasi. Format tabel berisi : no, kode test, jenis fungsi test, bahan test, hasil yang diharapkan, hasil yang diperoleh dan kesimpulan. Contoh :

**No Kode Test :** IN01

**Jenis Fungsi Test :** Master Pelanggan

**Bahan Test :** form master pelanggan, list data pelanggan.

**Hasil Yang Diharapkan :** Fungsi master Pelanggan berjalan sesuai Desain.

**Hasil yang diperoleh :** fungsi master pelanggan sesuai dengan hasil yang diharapkan.

**Kesimpulan :** OK atau Sesuai

#### 6. Perawatan perangkat lunak (*Maintain*)

Tahap ini adalah tahapan terakhir dari Rancang Bangun Sistem Informasi yaitu mengevaluasi seluruh program dengan mengetahui data-data fungsional sistem berjalan sesuai rencana dan fungsional sistem yang belum jalan sebagaimana mestinya, tahap ini juga mengkaji untuk pengembangan berikutnya tentang Sistem Informasi yang dibuat. Hal yang perlu diperhatikan dalam *maintenance* ini adalah:

- Dokumen Kontrak sebagai acuan berapa lama durasi waktu yang diperlukan dalam tahapan perawatan ini.
- Laporan akhir dari kegiatan pembuatan perangkat lunak.
- Pengembangan aplikasi pada periode berikutnya.
- Perbaikan atau penambahan fungsi sistem yang dibuat.

## PENUTUP

### Kesimpulan

Penerapan metode *software development life cycle* (SDLC) akan mengurangi terjadinya software crisis karena dalam tahapannya sangat detail mulai dari tahapan analisa sampai dengan *maintenance* jika diimplementasikan akan menghasilkan dua produk yaitu dokumen spesifikasi kebutuhan perangkat lunak (SKPL) dan sebuah aplikasi sistem informasi yang sesuai dengan kebutuhan.

### Saran

Metode SDLC yang dimungkinkan dikembangkan dalam penelitian selanjutnya ke detail masing-masing

tahapan. Sangat disarankan untuk penelitian selanjutnya yaitu pada tahapan implementasi, testing dan maintenance pada pembangunan sistem informasi supaya dibahas dengan detail.

#### **DAFTAR PUSTAKA**

- [1] M. Eka Wijaya, 2012. Rancang Bangun Sistem Informasi Akuntansi dengan Paradigma Pengembangan Terintegrasi Menggunakan Enterprise Service Bus (ESB), Jurnal Sistem Informasi, Fakultas Teknologi Informasi, Institut Teknologi Sepuluh Nopember Surabaya, Indonesia
- [2] Nugroho, Bunafit. 2004. *Aplikasi Pemograman Web Dinamis dengan PHP dan MySQL*. Yogyakarta: Gava Media.
- [3] Rizky, Soetam. 2011. *Konsep Dasar Rekayasa Perangkat Lunak*. Jakarta: Prestasi Pustakarya.
- [4] Roger S. Pressman. 2005. *Software Engineering : A Practitioner's Approach. Sixth Edition*. Nort America. Mc Graw-Hill.
- [5] S, Rosa A dan M. Salahuddin. 2011. *Modul Pembelajaran Rekayasa Perangkat Lunak*. Bandung: Modula.
- [6] Winardi, Gunawan. 2002. *Panduan Mempersiapkan Tulisan Ilmiah*. Bandung: Akatiga.
- [7] Yoyok, 2012. Penyusunan Spesifikasi Kebutuhan Perangkat Lunak SIM Administrasi Surat Menggunakan Metode *Digital Signature* Dengan Algoritma *Base64*. Laporan Penelitian local LPPM, Universitas Kanjuruhan Malang.

