

PERTEMUAN 4 (PEMBUATAN MODEL SDLC)

A. TUJUAN PEMBELAJARAN

Setelah mempelajari materi ini mahasiswa diharapkan mampu untuk :

1. Dapat Melakukan Analisa Kebutuhan
2. Mengetahui Kebutuhan Fungsional dan Non-fungsional

B. URAIAN MATERI

1. Pembuatan Model SDLC (System Develop Life Cycle)

SDLC merupakan singkatan dari (*Systems Development Life Cycle*) dan dalam bahasa indonesia disebut juga (Siklus Hidup Pengembangan Sistem), pada rekayasa sistem serta rekayasa perangkat lunak, ialah suatu langkah-langkah pembuatan serta perubahan sistem juga model dan metodologi yang nantinya akan dipakai untuk mengembangkan sistem-sistem tersebut. Pada umumnya konsep ini lebih merujuk pada sistem informasi ataupun komputer. Berikut ini adalah pengertian SDLC menurut ahli :

SDLC menurut Prof. Dr. Sri Mulyani, AK., CA. (2017) merupakan suatu proses atau tahapan logika yang dipakai oleh seorang analis sistem demi mengembangkan sebuah sistem informasi yang juga melibatkan requirements, validation, training dan pemilik sistem.

SDLC juga ialah sebuah pola yang akan diambil sebagai mengembangkan sistem perangkat lunak, dan terdiri dari tahap-tahap dibawah ini :

a. Perencanaan Sistem (*Systems Planning*)

Menekankan lebih dalam aspek studi kelayakan pengembangan sistem (*feasibility study*). Aktivitas-aktivitasnya meliputi :

- 1) Melakukan pembentukan serta konsolidasi sistem pengembang.
- 2) Menjabarkan tujuan serta ruang lingkup dari pengembangan.
- 3) Mengenali apa masalah-masalah tersebut bisa diselesaikan dengan menggunakan pengembangan sistem.

- 4) Menentukan kemudian mengevaluasi strategi apa yang nantinya akan dipakai pada pengembangan sistem.
- 5) Menentukan prioritas pada teknologi serta pemilihan pada aplikasi.

b. Analisis Sistem (*System Analysis*)

Didalam fase ini, akan dilakukannya analisis pada sistem untuk kemudian akan dijalankan nantinya. Kemudian hasil dari analisis yang berupa kelebihan serta kekurangan sistem, fungsi sistem, sampai pembaharuan yang bisa diterapkan. Pada bagian ini termasuk pada bagian perencanaan. Lalu bagian lainnya yang ikut termasuk dalam perencanaan adalah alokasi sumber daya, perencanaan kapasitas, penjadwalan proyek, estimasi biaya, serta penetapan. Demikian, adalah hasil pada tahap perencanaan yaitu rencana pada proyek, jadwal, estimasi biaya, dan ketentuan. Idealnya manajer proyek serta si pengembang bisa bekerja dengan maksimal dalam tahap ini.

Analisis sistem merupakan tahap yang mana dilaksakannya beberapa aktivitas berikut ini :

- 1) Melaksanakan studi literatur demi mendapatkan sebuah kasus yang nantinya bisa ditangani oleh sistem.
- 2) Brainstorming pada tim pengembang terkait kasus yang paling tepat kemudian dimodelkan dengan sistem.
- 3) Melakukan klasifikasi pada masalah, peluang, dan solusi yang berkemungkinan bisa diterapkan pada kasus itu.
- 4) Melakukan analisis kebutuhan dalam sistem kemudian membuat suatu tatasan-batasan pada sistem.
- 5) Menjabarkan segala kebutuhan-kebutuhan pada sistem.

c. Perancangan Sistem (*System Design*)

Pada fase atau tahap ini akan menghasilkan sebuah prototype serta beberapa output lain mencakup dokumen berisi desain, pola, juga komponen yang dibutuhkan demi mewujudkan proyek tersebut. Kemudian setelah spesifikasi, akan dilaksanakan perancangan sistem sebagai fase atau tahap lanjutannya. Fase ini merupakan fase di mana semua hasil analisis serta pembahasan tentang spesifikasi sistem diterapkan menjadi suatu rancangan ataupun cetak biru sebuah sistem. Fase ini disebut sebagai cetak biru, karena

di mana sistem sudah siap untuk dikembangkan di mulai dari implementasi, analisis sistem, sampai tenaga pendukung sistem yang nantinya akan dikembangkan.

Dalam fase ini, *features* serta operasi-operasi pada sistem dijelaskan dengan sangat terperinci. Aktivitas-aktivitasnya meliputi:

- 1) Melakukan analisa interaksi pada obyek serta fungsi pada sistem.
- 2) Melakukan analisa pada data serta membuat skema database.
- 3) Membuat rancangan pada UI (*User Interface*).

d. Implementasi Sistem (*Systems Implementation*)

Fase atau tahap selanjutnya ialah implementasi yaitu mengimplementasikan rancangan pada tahap-tahap sebelumnya serta melaksanakan sebuah uji coba. Pada implementasi, dilakukannya aktivitas-aktivitas berikut ini :Pembuatan database sesuai skema rancangan.

- 1) Melakukan pembuatan database yang sesuai dengan skema rancangan.
- 2) Melakukan pembuatan aplikasi yang berdasarkan desain sistem.
- 3) Melakukan pengujian serta perbaikan aplikasi (*debugging*).

e. Pemeliharaan Sistem (*Systems Maintenance*)

Dan yang terakhir adalah pemeliharaan sistem, dilaksanakan oleh admin yang telah dipilih untuk menjaga sistem supaya tetap mampu beroperasi dengan benar melalui kemampuan sistem dalam mengadaptasikan diri sesuai dengan kebutuhannya.

2. Teknik Konsep Dasar Spesifikasi Formal

Dalam istilah teknik atau metode formal meliputi beberapa kegiatan, termasuk juga dalam spesifikasi sistem formal, analisa serta bukti pada spesifikasi, pengembangan transformasional serta verifikasi pada program. Seluruh kegiatan ini bertumpu pada spesifikasi formal perangkat lunak. Sebuah spesifikasi perangkat lunak formal adalah sebuah spesifikasi yang dijelaskan dalam bahasa yang perbendaharaan kata, sintaks, dan semantiknya didefinisikan secara formal. Kebutuhan akan definisi formal ini mesti diartikan bahwa bahasa spesifikasi selayaknya berbasiskan konsep matematika yang

propertinya telah diselidiki dan dipahami dengan baik. Cabang matematika yang dipakai disebut matematika diskrit dan konsep matematikanya diambil dari teori himpunan, logika, dan aljabar.

Pada tahun 1980-an, spesifikasi formal dan metode formal yang lebih umum dianggap oleh banyak peneliti sebagai rule yang paling mungkin dijalani demi mencapai perbaikan dramatis pada kualitas perangkat lunak. Mereka berpendapat bahwa analisis yang teliti dan rinci, yang merupakan bagian sangat penting pada metode formal, akan menghasilkan program dengan error yang lebih sedikit dan lebih sesuai dengan keperluan user. Mereka meramalkan bahwa, pada abad ke-21, sebagian besar perangkat lunak akan dikembangkan dengan memakai metode formal. Jelas, ramalan ini belum terbukti. Ada beberapa alasan untuk hal ini:

- a. Rekayasa perangkat lunak yang berhasil. Penggunaan metode-metode rekayasa perangkat lunak seperti metode terstruktur, manajemen konfigurasi, penyembunyian informasi, dll pada proses perancangan dan pengembangan menghasilkan perbaikan kualitas perangkat lunak. Hal ini berlawanan dengan ramalan bahwa bukti program sangat penting untuk perbaikan kualitas.
- b. Perubahan pasar. Pada tahun 1980-an, kualitas perangkat lunak terlihat sebagai kunci masalah rekayasa perangkat lunak. Namun demikian, sejak saat itu, isu kunci untuk banyak kelas pengembangan perangkat lunak bukanlah kualitas perangkat lunak tetapi waktu-pemasarannya. Perangkat lunak harus dikembangkan dengan cepat dan pelanggan seringkali bersedia menerima perangkat lunak dengan beberapa kesalahan, asalkan penyerahan yang cepat dapat dilakukan. Teknik pengembangan perangkat lunak yang cepat tidak berhadapan dengan baik dengan spesifikasi formal. Tentu saja, kualitas merupakan faktor yang penting, tetapi tetap saja kualitas tersebut harus dicapai dalam konteks penyerahan yang cepat.
- c. Lingkup yang terbatas dari metode formal. pada umumnya metode formal tidak sesuai untuk menspesifikasi interface user dan interaksi user. Walaupun komponen interface user menjadi bagian yang makin besar dari banyak sistem, keuntungan yang didapat dari pemakaian metode formal dibatasi.
- d. Skala metode formal yang terbatas. Skala metode formal tidak bertambah dengan baik. Proyek-proyek sukses yang telah memakai teknik ini telah membatasi penggunaannya pada sistem kernel kritis yang relatif kecil.

Masalah ini diperburuk dengan ketiadaannya dukungan alat bantu untuk teknik-teknik ini.

Faktor-faktor ini mengandung arti bahwa risiko yang kita hadapi dalam pemakaian metode formal pada sebagian besar proyek perangkat lunak lebih besar daripada keuntungan yang mungkin kita dapat dari pemakaiannya. Biaya dan masalah dalam pengenalan metode formal pada proses perangkat lunak sangat tinggi. Namun demikian, spesifikasi formal merupakan cara yang sangat bagus untuk menemukan error spesifikasi dan mempresentasikan spesifikasi sistem dengan cara yang unik. Semua proyek berhasil yang memakai metode formal telah melaporkan error yang lebih sedikit pada perangkat lunak yang diserahkan.

Dengan demikian, pada sistem dimana kesalahan harus dihindari, pemakaian metode formal dapat dibenarkan dan mungkin akan mengefektifkan biaya. Pemakaian metode formal akan bertambah dalam area tertentu pengembangan sistem kritis di mana properti sistem yang baru (*emergent*) seperti keselamatannya, keandalannya, serta keamanannya penting sekali. Sistem-sistem kritis ini, mempunyai biaya validasi yang sangat tinggi dan biaya kegagalan sistem berjumlah besar dan bertambah. Metode formal dipakai karena dapat memperkecil biaya-biaya ini.

a. Metode Formal

- 1) Spesifikasi formal adalah komponen dalam kumpulan yang sangat lumrah teknik yang diketahui sebagai metode formal.
- 2) Berlandaskan pada delegasi matematis serta menganalisis sebuah software.
- 3) Metode formal meliputi :
 - a) Spesifikasi formal
 - b) Spesifikasi analisis dan bukti
 - c) Transformasional pembangunan
 - d) Program verifikasi.

b. Penerimaan Metode Formal

Metode formal belum menjadi sebuah teknik yang banyak digunakan dalam pengembangan perangkat lunak seperti yang pernah diperkirakan

- 1) Teknik rekayasa perangkat lunak lain sudah berhasil menaikkan kualitas sistem. Oleh sebab itu dibutuhkan metode formal menjadi berkurang
- 2) Perubahan pada pasar telah membuat bukan perangkat lunak dengan kesalahan yang rendah dalam menghitung factor kunci. Metode formal tidak dapat mengurangi waktu ke pasar
- 3) Ruang lingkup metode formal cukup memiliki batas. Mereka kurang cocok untuk menetapkan serta menganalisis user interface dan interaksi pengguna
- 4) Metode formal masih terbilang cukup sulit untuk skala dengan sistem yang besar

c. Penggunaan Metode atau Teknik Formal

- 1) Manfaat yang paling utama pada teknik atau metode formal ini ialah mampu mengurangi jumlah kesalahan pada system
- 2) Sehingga, daerah penting mereka akan diterapkan dalam rekayasa sistem kritis. Terdapat beberapa proyek yang berhasil dimana di daerah ini metode formal telah dipergunakan
- 3) Pada daerah ini, penggunaan teknik atau metode formal yang paling mungkin untuk biaya-biaya sistem efektif karena kegagalan yang tinggi harus dihindari
- 4) Metode formal belum menjadi sebuah teknik yang banyak digunakan dalam pengembangan perangkat lunak seperti yang pernah diperkirakan

d. Spesifikasi dalam Proses Perangkat Lunak

- 1) Spesifikasi dan desain adalah dua hal yang sangat erat
- 2) Desain arsitektur merupakan hal yang sangat penting untuk struktur spesifikasi dan proses spesifikasi
- 3) Spesifikasi formal dinyatakan dalam notasi matematika dengan tepat didefinisikan, sintaks dan sistematik kosakata.

e. Penggunaan Spesifikasi Formal

- 1) Spesifikasi formal ini akan sangat melibatkan investasi yang lebih banyak usaha pada tahap awal pengembangan perangkat lunak
- 2) Hal tersebut akan mengurangi kesalahan persyaratan sebagai analisis rinci tentang persyaratan
- 3) Ketidakkonsistenan serta ketidaklengkapan bisa didapatkan lalu kemudian diselesaikan
- 4) Oleh sebab tersebut, tabungan akan dibuat untuk jumlah pengerjaan ulang, karena masalah dari persyaratan tersebut akan berkurang.

f. Biaya Profil

Penggunaan spesifikasi formal berarti bahwa akan adanya biaya profil pada sebuah proyek

- 1) Adanya biaya dimuka atau didepan yang lebih besar serta lebih banyaknya usaha dan juga waktu yang harus dihabiskan untuk mengembangkan spesifikasi
- 2) Akan tetapi, pelaksanaan serta validasi biaya harus dikurangi sebagai proses spesifikasi dalam hal mengurangi kesalahan serta ambiguitas dalam persyaratan tersebut.

g. Spesifikasi Teknik

- 1) Spesifikasi aljabar, sistem tersebut akan digunakan dalam hal operasi dan hubungan dengan mereka.
- 2) Model berbasis spesifikasi, sistem ini digunakan dalam hal model negara yang dibangun menggunakan konstruksi matematika seperti set serta urutan. Operasi diartikan oleh modifikasi negara sistem.

h. Spesifikasi *Interface*

- 1) Sistem yang besar akan dijadikan suatu subsistem dengan cara dipecahkan dengan antarmuka yang sudah diartikan dengan baik antara subsistem lainnya
- 2) Spesifikasi antarmuka subsistem memiliki adanya kemungkinan pengembangan independen dari subsistem yang berbeda.

- 3) Antarmuka atau *UserInterface* dapat diartikan sebagai kelas objek atau tipe data abstrak
- 4) Pendekatan aljabar untuk spesifikasi formal sangat tepat atau pas untuk antarmuka spesifikasi seperti yang diutamakan dalam operasi yang diartikan dalam suatu objek.

i. Spesifikasi Komponen

- 1) Pengantar, yaitu menyatakan spesifikasi lainnya yang akan digunakan serta mengartikan jenis (nama jenis)
- 2) Deskripsi, yaitu informal yang akan menjelaskan operasi pada jenis
- 3) Tanda Tangan, yaitu mengartikan sintaks operasi dengan mengartikan aksioma yang mencirikan perilaku.

j. Sistematis Aljabar Spesifikasi

Pada spesifikasi aljabar dari suatu sistem dapat kita dikembangkan lagi secara sistematis, yaitu :

- 1) Spesifikasi penataan
- 2) Spesifikasi penamaan
- 3) Operasi seleksi
- 4) Informal operasi spesifikasi
- 5) Sintaks definisi
- 6) Definisi aksioma

k. Spesifikasi Operasi

- 1) Pembuatan operasi, yaitu suatu operasi yang akan melahirkan sebuah entitas pada jenis yang telah ditentukan
- 2) Inspeksi operasi, yakni suatu operasi yang akan memberikan penilaian entitas dari jenis yang akan ditentukan
- 3) Untuk menetapkan suatu perilaku, akan dilakukan penentuan inspektur operasi untuk setiap operasi konstruktor.

l. Operasi Pada Daftar ADT

- 1) Pembuatan operasi yang melakukan evaluasi demi menyusun daftar, Membuat, kontra dan tail
- 2) Melakukan pemeriksaan operasi yang mengambil daftar yang semacam untuk parameter serta pengembalian semacam yang lain, Kepala dan panjang
- 3) Ekor bisa diartikan dengan memakai konstruktor yang sederhana. Membuat dan cons. Tidak perlu menjelaskan kepala dan panjang dengan tail.

m. *Interface* Spesifikasi Sistem Kritis

Berikut ini adalah contoh simulasi dari pesawat terbang, yaitu :

- 1) Mempertimbangkan suatu sistem kontrol lalu lintas udara yang dimana pesawat terbang akan melewati sektor yang telah dikelola wilayah udara
- 2) Pada setiap sektornya mampu meliputi beberapa pesawat namun, untuk suatu alasan keamanan, ini harus dipisahkan
- 3) Dalam contoh ini, pemisahan sederhana 300m diusulkan
- 4) Sistem itu perlu memperingatkan si controller apabila pesawat diintruksikan untuk bergerak sehingga pemisahan aturan dilanggar.

n. Sebuah Objek Sektor

Operasi kritis pada objek yang mewakili sektor yang dikendalikan, yaitu:

- 1) Enter, untuk menambahkan pesawat udara ke wilayah udara yang dikendalikan
- 2) Tinggalkan, untuk menghapus pesawat terbang dari wilayah udara yang dikendalikan
- 3) Pindah, untuk memindahkan pesawat dari suatu ketinggian ke yang lain
- 4) Lookup, untuk mengingat pesawat identifier, kembali tinggi saat ini.

o. Primitif Operasi

Terkadang hal ini dibutuhkan untuk memperkenalkan operasi tambahan untuk menyederhanakan sebuah spesifikasi, operasi primitive yaitu :

- 1) Buat, yaitu membuat sebuah instance pada sebuah sektor menjadi ada

- 2) Put, yaitu tambahkan pesawat tanpa sebuah pemeriksaan keamanan
 - 3) Dalam-ruang, yaitu melakukan penentuan apakah sebuah pesawat yang diberikan dalam sektor ini
 - 4) Pendudukan, yaitu mengingat ketinggian yang sedang dilakukan, serta menentukan apa ada pesawat lainnya pada ketinggian 200m itu.
- p. Spesifikasi Komentar
- 1) Gunakan konstruktor dasar Buat serta Pasang untuk menentukan operasi lainnya
 - 2) Tentukan juga Pendudukan serta Dalam-ruang angkasa yang menggunakan Buat dan Pasang serta pakailah mereka untuk membuat cek pada definisi operasi lain
 - 3) Semua operasi yang mengakibatkan perubahan pada sektor ini harus melakukan pemeriksaan bahwa kriteria keselamatan berlaku.
- q. Perilaku Spesifikasi
- 1) Aljabar spesifikasi bisa menjadi rumit apabila operasi objek tidak berdiri sendiri atau independen dari negara objek
 - 2) Spesifikasi model berbasis akan memperlihatkan keadaan sistem serta mengartikan operasi dalam hal perubahan ke negara itu
 - 3) Notasi Z, yaitu teknik matang untuk spesifikasi model yang berbasis. Akan menggabungkan deskripsi formal dan informal juga menggunakan grafis menyoroti saat akan menyajikan spesifikasi.
- r. Skema Invarian
- 1) Setiap skema Z memiliki komponen invarian yang menjelaskan keadaan yang selalu benar
 - 2) Untuk skema pompa insulin akan selalu benar yaitu
 - a) Dosis wajib kurang dari atau dengan kapasitas reservoir insulin
 - b) Tidak ada dosis tunggal yang bisa lebih dari 4 unit insulin serta total dosis disampaikan dalam kurun waktu tidak boleh lebih dari 25 insulin. Ini merupakan kendala pada keamanan

s. Perhitungan Pada Dosis

- 1) Pompa insulin akan menghitung jumlah insulin yang akan dibutuhkan dengan cara melakukan perbandingan saat ini dengan membaca dua bacaan sebelumnya
- 2) Jika menunjukkan bahwa glukosa darah naik maka insulin disampaikan
- 3) Informasi terkait total dosis dikirimkan lalu dipertahankan demi memungkinkan invarian pengecekan keamanan yang akan digunakan
- 4) Perhatikan bahwa invarian selalu berlaku, tidak perlu mengulang dalam perhitungan dosis

t. Hal-hal Penting lainnya

- 1) Spesifikasi sistem formal akan melengkapi spesifikasi sistem informal
- 2) Spesifikasi formal tepat dan jelas, sehingga mereka dapat menghapus bidang keraguan dalam spesifikasi
- 3) Spesifikasi formal kekuatan analisis kebutuhan sistem dalam fase awal. Melakukan koreksi kesalahan dalam fase ini akan jauh lebih murah dibandingkan memodifikasi sistem
- 4) Spesifikasi teknik formal adalah yang paling banyak digunakan pada pengembangan sistem kritis dan standar
- 5) Teknik aljabar sangat tepat untuk antarmuka spesifikasi antarmuka mana diartikan sebagai satu set kelas objek
- 6) Model-model teknik berdasarkan sistem dengan menggunakan set serta fungsi. Ini menyederhanakan beberapa jenis spesifikasi perilaku
- 7) Operasi diartikan dalam model berbasis spesifikasi, dengan menjelaskan dan mengartikan kondisi sebelum serta sesudah keadaan sistem

C. SOAL LATIHAN/TUGAS

1. Sebutkan langkah-langkah pembuatan serta perubahan sistem juga model dan metodologi dalam metode SDLC ?
2. Apa yang dimaksud implementasi pada metode SDLC ?

3. Sebutkan contoh Sintaks definisi ?
4. Mengapa Lingkup yang terbatas dari metode formal ?
5. Jelaskan metode formal menurut anda ?

D. REFERENSI

1. Pressman, R. S. (2015). Software Engineering. A Practitioner's Approach (8th ed.). New York: McGraw-Hill Education.
2. Sommerville, I. (2011). Software Engineering (9th ed.). Boston: Addison-Wesley.

GLOSARIUM

SDLC merupakan singkatan dari (*Systems Development Life Cycle*) dan dalam bahasa indonesia disebut juga (Siklus Hidup Pengembangan Sistem), pada rekayasa sistem serta rekayasa perangkat lunak.

Perencanaan Sistem (*Systems Planning*) Menekankan lebih dalam aspek studi kelayakan pengembangan sistem (*feasibility study*).

Notasi Z yaitu teknik matang untuk spesifikasi model yang berbasis. Akan menggabungkan deskripsi formal dan informal juga menggunakan grafis menyoroti saat akan menyajikan spesifikasi.