



**MODUL REKAYASA PERANGKAT LUNAK (RPL)
(CCC-110)**

MODUL 04
REQUIREMENTS ANALYSIS AND SPECIFICATION

DISUSUN OLEH
MALABAY,S.KOM,M.KOM

Universitas
Esa Unggul

UNIVERSITAS ESA UNGGUL

2020

REQUIREMENTS ANALYSIS AND SPECIFICATION

A. Kemampuan Akhir Yang Diharapkan

Setelah mempelajari modul ini, diharapkan mahasiswa mampu : Mahasiswa mampu memahami pengertian *Requirements Analysis and Specification*.

B. Uraian dan Contoh

Apa itu *Requirements Analysis* atau *Analisis Kebutuhan* ?

Requirements Analysis adalah proses mendefinisikan harapan pengguna untuk aplikasi yang akan dibangun atau dimodifikasi. Ini melibatkan semua tugas yang dilakukan untuk mengidentifikasi kebutuhan pemangku kepentingan yang berbeda. Oleh karena itu analisis kebutuhan berarti menganalisis, mendokumentasikan, memvalidasi dan mengelola perangkat lunak atau persyaratan sistem.

Persyaratan kualitas tinggi didokumentasikan, dapat ditindaklanjuti, terukur, dapat diuji, dapat dilacak, membantu mengidentifikasi peluang bisnis, dan didefinisikan untuk memfasilitasi desain sistem.

Proses analisis kebutuhan perangkat lunak melibatkan langkah / fase berikut:

1. Memperoleh persyaratan
2. Menganalisis persyaratan
3. Pemodelan persyaratan
4. Review dan retrospektif

1- Memperoleh persyaratan

Proses pengumpulan persyaratan dengan berkomunikasi dengan pelanggan dikenal sebagai persyaratan perolehan.

2- Menganalisis persyaratan

Langkah ini membantu untuk menentukan kualitas persyaratan. Ini melibatkan mengidentifikasi apakah persyaratan tidak jelas, tidak lengkap, ambigu, dan kontradiktif. Masalah ini diselesaikan sebelum melanjutkan ke langkah berikutnya.

3- Pemodelan persyaratan

Dalam pemodelan Persyaratan, persyaratan biasanya didokumentasikan dalam format yang berbeda seperti kasus penggunaan, cerita pengguna, dokumen bahasa alami, atau spesifikasi proses.

4- Review dan retrospektif

Langkah ini dilakukan untuk merefleksikan iterasi pengumpulan persyaratan sebelumnya dalam upaya melakukan perbaikan dalam proses ke depan.

Teknik Analisis Persyaratan:

Ada teknik berbeda yang digunakan untuk Analisis Kebutuhan bisnis. Di bawah ini adalah daftar dari berbagai Teknik Analisis Kebutuhan bisnis:

1. Notasi pemodelan proses bisnis (BPMN)
2. UML (Unified Modeling Language)
3. Teknik diagram alir
4. Diagram aliran data
5. Diagram Aktivitas Peran (RAD)
6. Diagram Gantt
7. IDEF (Definisi Terpadu untuk Pemodelan Fungsi)
8. Analisis Kesenjangan

1- Notasi pemodelan proses bisnis (BPMN)

Teknik ini mirip dengan pembuatan diagram alur proses, meskipun BPMN memiliki simbol dan elemennya sendiri. Pemodelan dan notasi proses bisnis digunakan untuk membuat grafik untuk proses bisnis. Grafik ini menyederhanakan pemahaman proses bisnis. BPMN sangat populer sebagai metodologi peningkatan proses.

2- UML (Unified Modeling Language)

UML terdiri dari sekumpulan diagram terintegrasi yang dibuat untuk menentukan, memvisualisasikan, membangun, dan mendokumentasikan artefak dari sistem perangkat lunak. UML adalah teknik yang berguna saat membuat perangkat lunak berorientasi objek dan bekerja dengan proses pengembangan perangkat lunak. Dalam UML, notasi grafis digunakan untuk merepresentasikan desain proyek perangkat lunak. UML juga membantu dalam memvalidasi desain arsitektur perangkat lunak.

3- Teknik diagram alir

Diagram alir menggambarkan aliran sekuensial dan logika kontrol dari serangkaian aktivitas yang terkait. Diagram alir memiliki format yang berbeda seperti linier, fungsi silang, dan atas-bawah. Diagram alir dapat mewakili interaksi sistem, aliran data, dll. Diagram alir mudah dipahami dan dapat digunakan oleh anggota tim teknis dan non-teknis. Teknik diagram alir membantu dalam menampilkan atribut kritis dari suatu proses.

4- Diagram aliran data

Teknik ini digunakan untuk merepresentasikan sistem dan proses secara visual yang kompleks dan sulit dijelaskan dalam teks. Diagram aliran data merepresentasikan aliran informasi melalui suatu proses atau sistem. Ini juga mencakup input dan output data, penyimpanan data, dan berbagai subprocess di mana data bergerak. DFD menggambarkan berbagai entitas dan hubungannya dengan bantuan notasi dan simbol standar. Dengan memvisualisasikan semua elemen sistem, lebih mudah untuk mengidentifikasi kekurangan apa pun. Kekurangan ini kemudian dieliminasi dalam upaya untuk menciptakan solusi yang kuat.

5- Diagram Aktivitas Peran (RAD)

Diagram aktivitas peran (RAD) adalah model proses berorientasi peran yang merepresentasikan diagram aktivitas peran. Diagram aktivitas peran adalah

tampilan tingkat tinggi yang menangkap dinamika dan struktur peran organisasi. Peran digunakan untuk mengelompokkan aktivitas menjadi unit tanggung jawab. Aktivitas adalah bagian dasar dari sebuah peran. Suatu kegiatan dapat dilakukan secara terpisah atau mungkin memerlukan koordinasi dengan kegiatan lain dalam peran tersebut.

6- Grafik Gantt

Bagan Gantt digunakan dalam perencanaan proyek karena memberikan representasi visual dari tugas yang dijadwalkan bersama dengan garis waktu. Grafik Gantt membantu untuk mengetahui apa yang dijadwalkan untuk diselesaikan pada tanggal berapa. Tanggal mulai dan berakhir semua tugas dalam proyek dapat dilihat dalam satu tampilan.

7- IDEF (Definisi Terpadu untuk Pemodelan Fungsi)

Definisi terintegrasi untuk teknik pemodelan fungsi (IDEFM) mewakili fungsi dari suatu proses dan hubungannya dengan sistem anak dan induk dengan bantuan sebuah kotak. Ini memberikan cetak biru untuk mendapatkan pemahaman tentang sistem organisasi.

8- Analisis Kesenjangan

Analisis kesenjangan adalah teknik yang membantu menganalisis kesenjangan dalam kinerja aplikasi perangkat lunak untuk menentukan apakah persyaratan bisnis terpenuhi atau tidak. Ini juga melibatkan langkah-langkah yang harus diambil untuk memastikan bahwa semua persyaratan bisnis dipenuhi dengan sukses. Gap menunjukkan perbedaan antara keadaan saat ini dan keadaan sasaran. Analisis kesenjangan juga dikenal sebagai analisis kebutuhan, penilaian kebutuhan atau analisis kesenjangan kebutuhan.

Untuk keberhasilan suatu proyek, sangat penting untuk menganalisis persyaratan proyek ketika mereka dikumpulkan serta sepanjang siklus hidup proyek. Analisis Persyaratan Perangkat Lunak membantu menjaga persyaratan sejalan dengan

kebutuhan bisnis. Proses analisis kebutuhan proyek yang baik akan menghasilkan aplikasi perangkat lunak yang memenuhi tujuan bisnis yang ditetapkan.

Analisis Kebutuhan, juga dikenal sebagai Teknik Kebutuhan, adalah proses menentukan harapan pengguna untuk perangkat lunak baru yang sedang dibangun atau dimodifikasi. Dalam rekayasa perangkat lunak, kadang-kadang disebut secara longgar dengan nama seperti pengumpulan persyaratan atau penangkapan persyaratan. Analisis kebutuhan mencakup tugas-tugas yang menentukan kebutuhan atau kondisi yang harus dipenuhi untuk produk atau proyek baru atau yang diubah, dengan mempertimbangkan persyaratan yang mungkin bertentangan dari berbagai pemangku kepentingan, menganalisis, mendokumentasikan, memvalidasi, dan mengelola perangkat lunak atau persyaratan sistem. Berikut adalah tujuan untuk melakukan analisis kebutuhan pada tahap awal proyek perangkat lunak:

From What to How: Tugas rekayasa perangkat lunak yang menjembatani kesenjangan antara rekayasa persyaratan sistem dan desain perangkat lunak.

3 Orthogonal Views: Menyediakan perancang perangkat lunak dengan model:

informasi sistem (tampilan statis)

fungsi (tampilan fungsional)

perilaku (tampilan dinamis)

Arsitektur Perangkat Lunak: Model dapat diterjemahkan ke data, arsitektur, dan desain tingkat komponen.

Proses Iteratif dan Inkremental: Berharap untuk melakukan sedikit desain selama analisis dan sedikit analisis selama desain.

Kebutuhan perangkat lunak adalah kemampuan yang dibutuhkan oleh pengguna untuk memecahkan suatu masalah atau untuk mencapai suatu tujuan. Dengan kata lain, persyaratan adalah kemampuan perangkat lunak yang harus dipenuhi atau dimiliki oleh sistem atau komponen sistem untuk memenuhi kontrak, standar, spesifikasi, atau dokumentasi lain yang diberlakukan secara formal. Pada akhirnya, yang ingin kami capai adalah mengembangkan perangkat lunak

berkualitas yang memenuhi kebutuhan nyata pelanggan tepat waktu dan sesuai anggaran.

Mungkin tantangan terbesar yang dihadapi oleh para pengembang perangkat lunak adalah berbagi visi produk akhir dengan pelanggan. Semua pemangku kepentingan dalam sebuah proyek - pengembang, pengguna akhir, manajer perangkat lunak, manajer pelanggan - harus mencapai pemahaman yang sama tentang produk yang akan dan akan dilakukan, atau seseorang akan terkejut ketika produk itu dikirim. Kejutan dalam perangkat lunak hampir tidak pernah menjadi kabar baik.

Oleh karena itu, kami memerlukan cara untuk menangkap, menafsirkan, dan merepresentasikan suara pelanggan secara akurat saat menentukan persyaratan untuk suatu produk perangkat lunak.

Aktivitas untuk Analisis Kebutuhan

Analisis kebutuhan sangat penting untuk keberhasilan atau kegagalan sistem atau proyek perangkat lunak. Persyaratan harus didokumentasikan, dapat ditindaklanjuti, dapat diukur, dapat diuji, dapat dilacak, terkait dengan kebutuhan atau peluang bisnis yang teridentifikasi, dan ditetapkan ke tingkat detail yang cukup untuk desain sistem. Secara konseptual, analisis kebutuhan mencakup empat jenis kegiatan:

Memperoleh persyaratan: tugas berkomunikasi dengan pelanggan dan pengguna untuk menentukan apa persyaratan mereka. Ini terkadang juga disebut pengumpulan persyaratan.

Menganalisis persyaratan: menentukan apakah persyaratan yang dinyatakan tidak jelas, tidak lengkap, ambigu, atau kontradiktif, dan kemudian menyelesaikan masalah ini.

Pemodelan persyaratan: Persyaratan dapat didokumentasikan dalam berbagai bentuk, seperti dokumen bahasa alami, kasus penggunaan, cerita pengguna, atau spesifikasi proses.

Tinjau dan retrospektif: Anggota tim merenungkan apa yang terjadi dalam iterasi dan mengidentifikasi tindakan untuk perbaikan di masa mendatang.

Analisis kebutuhan adalah upaya tim yang menuntut kombinasi dari perangkat keras, perangkat lunak dan keahlian rekayasa faktor manusia serta keterampilan dalam berurusan dengan orang. Berikut adalah kegiatan utama yang terlibat dalam analisis kebutuhan:

Identifikasi kebutuhan pelanggan.

Evaluasi sistem untuk kelayakan.

Lakukan analisis ekonomi dan teknis.

Alokasikan fungsi ke elemen sistem.

Tetapkan jadwal dan batasan.

Buat definisi sistem.

Teknik Analisis Kebutuhan

Analisis kebutuhan membantu organisasi untuk menentukan kebutuhan sebenarnya dari para pemangku kepentingan. Pada saat yang sama, ini memungkinkan tim pengembangan untuk berkomunikasi dengan pemangku kepentingan dalam bahasa yang mereka pahami (seperti bagan, model, diagram alir,) alih-alih halaman teks.

Setelah persyaratan dikumpulkan, kami mendokumentasikan persyaratan dalam dokumen Spesifikasi Kebutuhan Perangkat Lunak (SRS), kasus penggunaan atau sebagai Kisah Pengguna, yang dibagikan dengan pemangku kepentingan untuk disetujui. Dokumen ini mudah dipahami oleh pengguna dan pengembang biasa. Setiap perubahan dalam persyaratan juga didokumentasikan dan melalui prosedur kontrol perubahan dan diselesaikan dengan persetujuan.

Persyaratan Bisnis vs Persyaratan Perangkat Lunak

Sebuah rencana bisnis atau proyek membutuhkan berbagai persyaratan untuk membantu menentukan tujuan dan menetapkan ruang lingkup pekerjaan yang akan dilakukan. Persyaratan juga memberikan konteks dan cara obyektif untuk mengukur kemajuan dan kesuksesan. Setelah persyaratan bisnis ditetapkan,

persyaratan perangkat lunak ditentukan dan dikembangkan untuk memajukan proyek.

Persyaratan Bisnis

Persyaratan bisnis berkaitan dengan tujuan, visi, dan sasaran bisnis. Mereka juga menyediakan ruang lingkup kebutuhan atau masalah bisnis yang perlu ditangani melalui aktivitas atau proyek tertentu. Misalnya, jika asosiasi perdagangan memiliki tujuan untuk mempromosikan layanan yang ditawarkan oleh anggotanya, persyaratan bisnis untuk suatu proyek dapat mencakup pembuatan direktori anggota yang meningkatkan kesadaran anggota. Persyaratan bisnis yang baik haruslah:

Jelas dan biasanya ditentukan pada level yang sangat tinggi.

Berikan informasi dan panduan yang cukup untuk membantu memastikan bahwa proyek memenuhi kebutuhan yang diidentifikasi.

Memahami mandat, tujuan atau sasaran organisasi, kebutuhan atau masalah bisnis tertentu yang sedang ditangani

Harus didefinisikan dan dipahami dengan jelas sebelum mengembangkan persyaratan bisnis.

Kebutuhan atau masalah dapat terkait dengan organisasi atau bisnis secara umum atau fokus pada kelompok pemangku kepentingan, seperti pelanggan, klien, pemasok, karyawan, atau kelompok lain.

Persyaratan Software

Persyaratan perangkat lunak menguraikan langkah-langkah yang diperlukan untuk memenuhi persyaratan atau persyaratan bisnis. Sedangkan persyaratan bisnis menyatakan 'mengapa' untuk sebuah proyek, persyaratan perangkat lunak menguraikan 'apa'.

Misalnya, jika persyaratan bisnis adalah membuat direktori anggota untuk asosiasi perdagangan, persyaratan perangkat lunak akan menguraikan siapa yang memiliki akses ke direktori, bagaimana anggota mendaftar dengan direktori, siapa yang

akan memiliki data, kendaraan atau kendaraan apa yang akan digunakan. digunakan seperti situs web atau direktori berbasis kertas, dan sebagainya.

Teknik Menemukan Kebutuhan Bisnis

Ada berbagai teknik analisis kebutuhan yang dapat digunakan sesuai dengan peningkatan bisnis dan proses pengembangan perangkat lunak. Di bawah ini adalah beberapa dari teknik ini.

Analisis Kesenjangan Menggunakan BPMN atau ArchiMate

Kesenjangan sering dikatakan sebagai "jarak antara di mana Anda berada dan di mana Anda ingin berada". Analisis kesenjangan adalah proses perbandingan antara skenario bisnis baseline dan target. Dengan kata lain, analisis kesenjangan adalah studi tentang apa yang dilakukan bisnis saat ini dan ke mana tujuannya di masa depan, dan dilakukan sebagai sarana untuk menjembatani ruang di antara mereka. Tujuan dari analisis kesenjangan adalah untuk mengidentifikasi kesenjangan dalam mengoptimalkan kinerja. Ini memberikan bisnis dengan wawasan tentang peningkatan potensial. Ini menjawab pertanyaan seperti apa keadaan proyek saat ini? Dimana kita ingin berada? dll.

ArchiMate - Analisis Kesenjangan

ArchiMate adalah bahasa pemodelan arsitektur perusahaan yang terbuka dan independen untuk mendukung deskripsi, analisis, dan visualisasi arsitektur di dalam dan di seluruh domain bisnis dengan cara yang tidak ambigu. Ini adalah alat pemodelan yang sempurna dan dengan notasi yang diperlukan untuk melakukan analisis celah.

Teknik untuk Mengidentifikasi Persyaratan Perangkat Lunak dari Kebutuhan: Bisnis

Diagram aliran data

Diagram Alir Data (DFD) dapat dirancang di awal proses perolehan persyaratan tahap analisis dalam SDLC (Siklus Hidup Pengembangan Sistem) untuk menentukan ruang lingkup proyek. DFD sering digunakan sebagai langkah awal

untuk membuat gambaran umum sistem tanpa harus terlalu detail, yang nantinya dapat diuraikan.

Misalnya, jika ada kebutuhan untuk menampilkan lebih detail dalam proses tertentu, proses tersebut diuraikan menjadi sejumlah proses yang lebih kecil di DFD tingkat yang lebih rendah. Dengan cara ini, Diagram Isi atau DFD Tingkat Konteks diberi label "DFD Tingkat-0" sedangkan tingkat dekomposisi berikutnya diberi label "DFD Tingkat-1", yang berikutnya diberi label "DFD Tingkat-2", dan begitu seterusnya.

Penggunaan UML adalah bentuk utama dari persyaratan sistem / perangkat lunak untuk program perangkat lunak baru yang sedang dikembangkan. Kasus penggunaan menentukan perilaku yang diharapkan (apa), dan bukan metode pasti untuk mewujudkannya (bagaimana). Kasus penggunaan setelah ditentukan dapat dilambangkan dengan representasi tekstual dan visual (seperti UML). Konsep utama dari pemodelan kasus penggunaan adalah membantu kami merancang sistem dari perspektif pengguna akhir. Ini adalah teknik yang efektif untuk mengkomunikasikan perilaku sistem dalam istilah pengguna dengan menentukan semua perilaku sistem yang terlihat secara eksternal.

Diagram use case biasanya sederhana. Ini tidak menunjukkan detail kasus penggunaan.

Ini hanya merangkum beberapa hubungan antara kasus penggunaan, aktor, dan sistem.

Ini tidak menunjukkan urutan langkah yang dilakukan untuk mencapai tujuan dari setiap use case.

Salah satu tugas tersulit dalam proyek apa pun adalah menetapkan dengan tepat apa itu persyaratan sistem. Ini adalah masalah yang dihadapi oleh manajer proyek dari waktu ke waktu dahulu kala, yang menyadari bahwa melakukannya dengan benar di awal pekerjaan adalah salah satunya paling penting. Insinyur telah lama menyadari bahwa disiplin, terorganisir dan pendekatan formal harus digunakan

saat mengevaluasi persyaratan system (apakah itu selalu dipraktikkan adalah masalah lain). Ini belum selesai melalui perasaan 'melakukan hal yang benar'. Tidak. Pengalaman, seringkali menyakitkan, telah menunjukkan bahwa metode seperti itu diperlukan. Secara khusus, dengan proyek apa pun ukuran sebenarnya, mereka sangat penting.

Bagaimana situasi mengenai proyek perangkat lunak? Mempertimbangkan jumlahnya dari kisah-kisah kegagalan besar yang beredar, jawabannya pasti 'sangat buruk'. Dalam Di masa lalu situasi ini sering kali dimaafkan karena perangkat lunak pada dasarnya sangat kompleks; orang tidak dapat mengharapkan hal lain dalam proyek semacam itu. Ini tidak masuk akal. Masalah sebenarnya adalah apa pun yang berbau organisasi dan disiplin dianggap mengganggu kreativitas pemrogram.

Namun, akhirnya, intinya datang ketika pendekatan seperti itu kepada profesional perangkat lunak tidak dapat ditoleransi lagi. Akibatnya, analisis kebutuhan dan spesifikasi perangkat lunak sekarang secara teratur dipraktikkan sebagai aspek formal desain perangkat lunak berkualitas. Tujuan bab ini adalah untuk:

- Bedakan antara model siklus hidup perangkat lunak yang mistis dan realistis.
- Tunjukkan di mana analisis kebutuhan dan spesifikasi perangkat lunak (tahap persyaratan) sesuai dengan siklus hidup.
- Soroti pentingnya tahap persyaratan.
- Jelaskan mengapa dan bagaimana kesalahan dibuat selama periode ini.
- Diskusikan analisis praktis dan metode spesifikasi.
- Perkenalkan topik pembuatan prototipe perangkat lunak.

Pada kenyataannya ini bukan tentang perangkat lunak; itu rekayasa proyek.

Masalah mendasar dari rekayasa perangkat lunak adalah masalah skala. Kompleksitas dan ukuran aplikasi yang menggunakan otomatisasi, dan akibatnya kompleksitas dan ukuran sistem perangkat lunak, terus menerus meningkat. Ketika scale berubah menjadi sistem perangkat lunak yang lebih kompleks dan lebih besar, baru masalah terjadi yang tidak ada dalam sistem yang lebih kecil

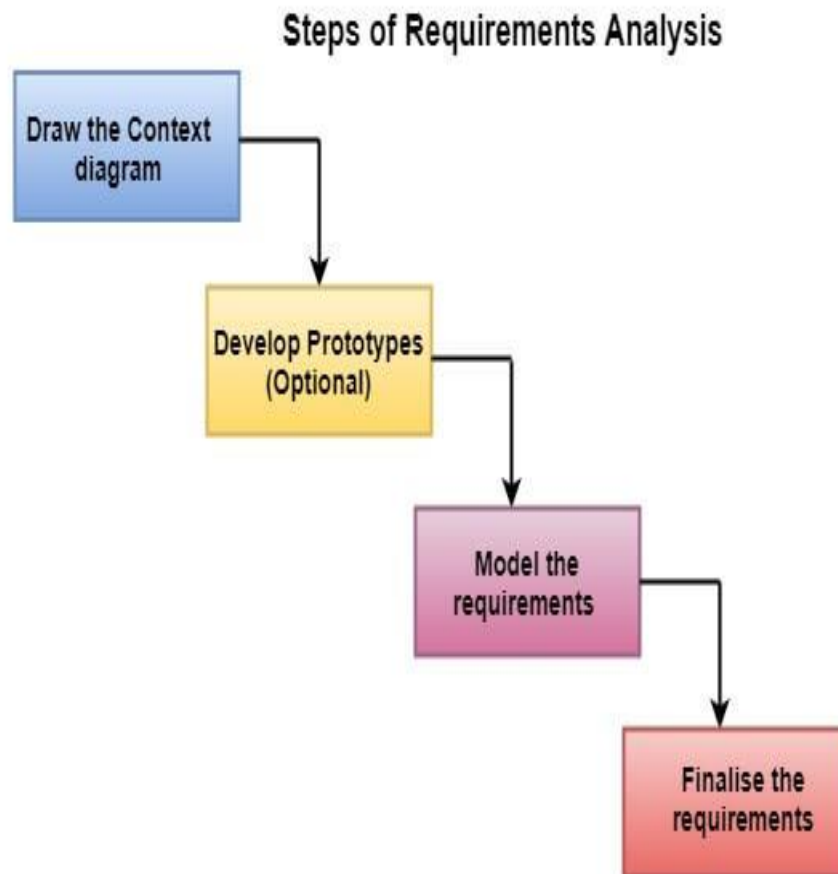
(atau yang memiliki signifikansi kecil), yang mengarah pada pendefinisian ulang prioritas kegiatan yang akan dikembangkan perangkat lunak. Persyaratan perangkat lunak adalah salah satu bidang seperti itu, yang tidak begitu penting dilampirkan pada hari-hari awal pengembangan perangkat lunak, karena penekanannya pada pengkodean dan desain. Asumsi diam-diam adalah bahwa pengembang memahami masalahnya jelas ketika dijelaskan kepada mereka, umumnya secara informal.

Ketika sistem tumbuh lebih kompleks, menjadi jelas bahwa tujuan dari keseluruhan sistem tidak dapat dengan mudah dipahami. Makanya kebutuhan yang lebih ketat fase persyaratan muncul. Sekarang, untuk sistem perangkat lunak besar, analisis kebutuhan mungkin merupakan aktivitas yang paling sulit dan sulit dilakukan; itu juga sangat rawan kesalahan. Banyak insinyur perangkat lunak percaya bahwa disiplin rekayasa perangkat lunak adalah yang terlemah area kritis ini.

Beberapa kesulitan disebabkan oleh cakupan fase ini. Proyek perangkat lunak ini diprakarsai oleh kebutuhan klien. Pada awalnya, kebutuhan ini ada di benak berbagai orang di organisasi klien. Analisis persyaratan harus mengidentifikasi persyaratan dengan berbicara kepada orang-orang ini dan memahami kebutuhan mereka. Di banyak situasi di mana perangkat lunak mengotomatiskan proses manual saat ini kebutuhan dapat dipahami dengan mengamati praktik saat ini. Tapi tidak seperti itu ada metode untuk sistem yang proses manualnya tidak ada (misalnya, perangkat lunak untuk sistem kendali rudal) atau untuk "fitur baru", yang sering ditambahkan saat mengotomatiskan proses manual yang ada.

Fase persyaratan menerjemahkan ide-ide di benak klien (masukan), ke dalam dokumen formal (keluaran dari tahap persyaratan). Jadi, keluaran dari fase ini adalah sekumpulan persyaratan yang ditentukan secara formal, yang diharapkan lengkap dan konsisten, sedangkan input tidak memiliki satu pun properti ini. Jelasnya, proses penetapan persyaratan tidak bisa sepenuhnya formal; setiap

proses penerjemahan formal yang menghasilkan keluaran formal harus memiliki masukan yang tepat dan tidak ambigu. Inilah sebabnya mengapa aktivitas persyaratan perangkat lunak tidak dapat sepenuhnya otomatis, dan metode apa pun untuk mengidentifikasi persyaratan terbaik dapat berupa seperangkat pedoman.



Gambar1. Tahapan Analisis Kebutuhan

C. Latihan

1. Proses mendefinisikan harapan pengguna untuk aplikasi yang akan dibangun atau dimodifikasi, disebut ?
2. Teknik yang digunakan untuk merepresentasikan sistem dan proses secara visual, disebut ?

D. Kunci Jawaban

1. *Requirements Analysis*
2. Diagram aliran data

E. Daftar Pustaka

1. Roger S. Pressman, Software Engineering A Practioner's Apporach, 2014
2. Ian Sommerville, Software Engineering (10th Edition), 2015
3. <https://reqtest.com/requirements-blog/requirements-analysis/>
4. <https://www.visual-paradigm.com/guide/requirements-gathering/requirement-analysis-techniques/>
5. https://link.springer.com/chapter/10.1007/978-1-4684-9312-2_3

