

Nama : Nuramalia Putri
NIM : 222101286
Prodi : Teknik Informatika
Matkul : Rekayasa Perangkat Lunak
Dosen Pengampu : Eko Marmanto P.U.,S.Kom.,M.Kom.,MOS.

Formatif 2 RPL – 09/Feb/2023

1. Komponen perangkat lunak terdiri dari
 - a. Sistem Informasi -> pada saat computer dinyalakan hal yang pertama kali dapat dibaca adalah booting sebuah *system software* seperti DOS, Linux dan Windows. Mereka berperan sebagai rumah utama bagi software tambahan yang nantinya akan digunakan didalam computer tersebut.
 - b. Program tambahan (utility) -> merupakan perangkat lunak dengan fungsi tertentu. Contohnya jika kita ingin memeriksa perangkat keras disket yang rusak, kita bisa mengatur ulang harddisk (defrag atau partisi) dengan Norton utility.
 - c. Bahasa Pemrograman -> merupakan Bahasa yang digunakan untuk membuat suatu program computer, disebut sebagai Bahasa computer karena kosakata dan tata bahasa tersebut perlu ditaati oleh setiap user agar tidak terjadi error.
 - d. Program aplikasi -> merupakan perangkat lunak yang siap digunakan sehingga dapat membantu pekerjaan yang dilakukan oleh user.
2. Perangkat Lunak merupakan Life Time Product -> maksud dari pernyataan tersebut bisa jadi karena software berbentuk file dan ditaruh dimanapun, ditanam dalam hardware, harddisk atau cloud. Selama penyimpanan tersebut tidak ada masalah, maka akan tetap ada. Maka asumsi dari pernyataan tersebut adalah *software yang sudah di compile tidak bisa di edit lagi*.
3.
 - a. Waterfall -> merupakan metode pengembangan software tradisional yang memiliki 5 tahapan proses. Prinsip dari metode ini adalah pengerjaan yang dilakukan secara tertata, berurutan, dan linear. Sederhananya, jika langkah satu belum dikerjakan, maka langkah selanjutnya pun tidak bisa dikerjakan. Metode ini dianggap tidak fleksibel karena semua proses dan fase harus dimulai kembali dari awal jika terjadi perubahan atau kesalahan pada proyek.
 - b. RAD (Rapid Application Development) -> merupakan metode pengembangan software dengan menekankan pada *feedback* dan kecepatan sepanjang siklus pengembangan dan pengujian langsung. Merupakan bagian dari metode *agile development* dimana developer bisa dengan mudah mengubah fungsi bahkan mengubah desain tanpa perlu merusak produk akhir. Prinsip utama dari metode RAD adalah untuk mengurangi proses perencanaan dan fokus pada proses konstruksi serta desain yang berulang sehingga developer bisa menyelesaikan proyek lebih cepat tanpa memengaruhi kepuasan klien.
 - c. Spiral Model -> merupakan model yang sangat berguna untuk melakukan pembangunan proyek-proyek besar dan prosesnya dilakukan dengan memperhatikan resiko proyek sehingga pada akhirnya akan menghasilkan model proses yang tepat sesuai kebutuhan pengguna. Merupakan gabungan antara model prototype dan waterfall. Dimana fokusnya pada Analisa resiko setiap tahapannya.
4. Divide -> berarti membagi masalah menjadi beberapa sub masalah yang memiliki kemiripan dengan masalah semula namun dengan ukuran kecil

Conquer -> berarti memecah atau menyelesaikan masing-masing masalah tersebut secara rekursif. Jika dalam pengembangan sistem hal ini tidak dilakukan maka akan berdampak pada kelanjutan sistem kedepannya Ketika akan melakukan improvisasi.

- Contoh sistem yang dapat meningkatkan kinerja sebuah organisasi, disini saya memakai contoh sistem yang saya buat sebagai divisi IT di dept EHS PT Century Batteries Indonesia. Sistem ini dibuat agar nantinya bisa memantau kinerja yang dilakukan dari setiap grup masing-masing department yang ada di PT CBI. Hal ini bertujuan agar masing-masing grup atau organisasi dapat lebih giat mengerjakan tugas atau menjalankan kinerja pekerjaan masing-masing dan nantinya diakhir akan mendapatkan sebuah penilaian dan penghargaan bagi grup yang menang.

ehs dept

Klik icon profile untuk login ke dalam sistem ->

Login

DOWNLOAD BERKAS SGA

BERKAS SGA

- Buku Panduan SGA
- Form Pendaftaran
- Template SGA
- Standard of Risk Rank
- Contoh Presentasi SGA 1
- Contoh Presentasi SGA 2

<- menu untuk mendownload berkas-berkas SGA

DATA SGA 2023

NO	NAMA DEPT	NAMA SIE	NAMA KASIE	NAMA GRUP	GRUP		JUMLAH ANGGOTA	STATUS
					NAMA KAGRUP	NO HP		
1	EHS	EHS	Ahmad Zaelani	JAKSEL	Dedi R		7	Finish
2	MIS	MIS	Rinta SN	PHP	Rico A	234234324	4	Finish
3	MKT	MARKETING	Rendi Widi Nugroho	ARTIS	Joko SP	085811151770	3	Tahap 1
4	FINACT	FINANCE,TREASURY,INV,COSTING	Agnes RA	SINCOS	Yulika Aristiana	082110397351	4	Tahap 1
5	MAINT	TOOLING 2	Latif Usman	SUNRISE	Cevi Doo		6	Tahap 1
6	FINACT	GENERAL ACC,TAX	Khanifatturahmah	SMART THINK	Salam Rudi		5	Tahap 5

Tampilkan keseluruhan data SGA

© 2023, made with ❤ by EHS Dept

- Menurut saya tahapan dalam pengembangan PL perlu memiliki jalur alternative Ketika, tahapan yang sudah disepakati dan disetujui oleh pihak user (client) dan pengembang (developer) yang saat ini sedang dilakukan tidak bisa berjalan dengan baik. Maka sebagai pengembang perlu memikirkan jalan lain agar sistem yang dikembangkan masih tetap bisa berjalan walau dengan cara yang berbeda tapi tetap menghasilkan hasil akhir yang sesuai dengan keinginan client.
- SRS -> Software Requirement Specification (SRS) atau Spesifikasi Kebutuhan Perangkat Lunak (SKPL) merupakan sebuah penjelasan tentang cara pengembangan dari sebuah software. SRS berisikan roadmap tentang semua pihak yang terlibat dalam sebuah proyek development software. Dokumen ini menawarkan spesifikasi fungsional dan non fungsional dari perangkat lunak dan juga mencakup use cases yang menggambarkan interaksi pengguna yang berada di dalam sistem dari awal hingga akhir.

Manfaat SRS ->

- Sebagai bentuk perjanjian antara customer dan supplier tentang software apa yang akan dibuat.
- Mengurangi beban dalam proses pengembangan software.
- Sebagai bahan perkiraan biaya dan rencana penjadwalan.
- Sebagai dasar validasi dan verifikasi software di ujung penyelesaian proyek nantinya.
- Memfasilitasi transfer, semisal software tersebut ingin ditransfer ke pengguna atau mesin-mesin yang lain. Customer pun merasa mudah jika ingin mentransfer software ke bagian-bagian lain dalam organisasinya. Bahkan, jika terjadi pergantian personil developer,

proyek dapat mudah ditransfer ke personil baru dengan memahami SRS ini.

6. Mendasari perbaikan produk software di kemudian hari. Jadi, kadang SRS boleh diperbaiki dengan alasan dan mekanisme tertentu serta atas kesepakatan antara customer dan developer.

Struktur Dokumen SRS ->

1. Planning -> Development Plan (SDP)
2. Requirements / Design -> High-Level Requirements and Design Documents
 - (SRS) -> WHAT
 - (SDD) -> HOW

->Detailed Requirements and Design Documents

- ER Diagram (Database,UML Diagram,Code)

- CAD Drawings

(The stakeholders and software team dibuat, dilanjutkan dengan pembuatan arsitektur,UX Designers dan developers.)

2(1) Testing -> Test Plans

-How to Test (STP)

-What to test (STD)

-> Test Documentation (buktikan bahwa sudah dilakukan pengujian dan sistem berjalan sesuai dengan keinginan)

3. End-User Documentation

-> System Documentation (Technical Stuff) -> super user, IT Dept.

-> Installation Guides (How to install it)

-> User Manuals (How to user it) -> End user.

8. Siklus Pengembangan Perangkat Lunak -> Merupakan serangkaian tahapan yang terstruktur dengan baik dalam rekayasa perangkat lunak untuk mengembangkan produk perangkat lunak yang dimaksudkan. SDLC (Siklus Hidup Pengembangan Sistem) atau Systems Life Cycle (Siklus Hidup Sistem), dalam rekayasa sistem dan rekayasa perangkat lunak, adalah proses pembuatan dan perubahan sistem serta model dan metodologi yang digunakan untuk mengembangkan sistem-sistem tersebut. Konsep ini umumnya merujuk pada sistem komputer atau informasi. SDLC juga merupakan pola yang diambil untuk mengembangkan sistem perangkat lunak, yang terdiri dari tahap-tahap: rencana (planning), analisis (analysis), desain (design), implementasi (implementation), uji coba (testing) dan pengelolaan (maintenance).

a. Tahap Perencanaan -> bertujuan untuk mengidentifikasi dan memprioritaskan sistem informasi apa yang akan dikembangkan, sasaran-sasaran yang ingin dicapai, jangka waktu pelaksanaan serta mempertimbangkan dana yang tersedia dan siapa yang melaksanakan.

b. Tahap Analisis -> merupakan kegiatan penguraian dari suatu sistem informasi yang utuh ke dalam bagian komponennya dengan maksud untuk mengidentifikasi dan mengevaluasi

permasalahan-permasalahan, hambatan-hambatan yang terjadi dan kebutuhan-kebutuhan yang diharapkan sehingga dapat diusulkan perbaikannya.

c. Tahap Desain Sistem -> memberikan gambaran rancang bangun (blue print) yang lengkap, sebagai penuntun (guideline) bagi programmer dalam membuat aplikasi.

Beberapa hal yang dilakukan dalam desain sistem adalah:

- Pemodelan sistem
- Desain Basis data
- Desain Aplikasi
- Desain Perangkat Keras/Jaringan
- Desain Antar Muka
- Deskripsi Pengguna

d. Tahap Implementasi Sistem -> tahap dimana sistem yang telah dibuat diinstalasi atau dipasangkan guna nantinya akan digunakan oleh user.

e. Tahap Pemeliharaan Sistem -> mencakup seluruh proses yang diperlukan untuk menjamin kelangsungan, kelancaran, dan penyempurnaan sistem yang telah dioperasikan.

9. **UML** -> UML (Unified Modelling Language) adalah suatu metode dalam pemodelan secara visual yang digunakan sebagai sarana perancangan sistem berorientasi objek. ML juga dapat didefinisikan sebagai suatu bahasa standar visualisasi, perancangan, dan pendokumentasian sistem, atau dikenal juga sebagai bahasa standar penulisan blueprint sebuah software.

Manfaat dari UML -> Dapat memberikan bahasa pemodelan visual atau gambar kepada para pengguna dari berbagai macam pemrograman maupun proses umum rekayasa. Menyatukan informasi-informasi terbaik yang ada dalam pemodelan. Memberikan suatu gambaran model atau sebagai bahasa pemodelan visual yang ekspresif dalam pengembangan sistem. Tidak hanya menggambarkan model sistem software saja, namun dapat memodelkan sistem berorientasi objek. Mempermudah pengguna untuk membaca suatu sistem. Berguna sebagai blueprint, jelas ini nantinya menjelaskan informasi yang lebih detail dalam perancangan berupa coding suatu program.

Diagram pada UML :

Usecase Diagram -> satu jenis dari diagram UML (Unified Modelling Language) yang menggambarkan hubungan interaksi antara sistem dan aktor. Use Case dapat mendeskripsikan tipe interaksi antara si pengguna sistem dengan sistemnya.

Activity Diagram -> merupakan sebuah diagram yang dapat memodelkan berbagai proses yang terjadi pada sistem. Seperti layaknya runtutan proses berjalannya suatu sistem dan digambarkan secara vertikal. Activity diagram adalah salah satu contoh diagram dari UML dalam pengembangan dari Use Case.

Sequence Diagram -> merupakan diagram yang menjelaskan interaksi objek berdasarkan urutan waktu. Sequence dapat menggambarkan urutan atau tahapan yang harus dilakukan untuk dapat menghasilkan sesuatu, seperti yang tertera pada Use Case diagram.

Class Diagram -> merupakan suatu diagram yang digunakan untuk menampilkan kelas-kelas berupa pake-paket untuk memenuhi salah satu kebutuhan paket yang akan digunakan nantinya.

10. Tahapan Pengujian Sistem merupakan proses pengeksesian untuk menemukan kesalahan kesalahan yang terdapat di dalam sistem, kemudian dilakukan pembenahan. Tahap ini merupakan tahap yang penting dalam pengembangan sistem karena pada tahap ini merupakan tahapan untuk memastikan bahwa suatu sistem terbebas dari kesalahan.