Functional Requirement & Non Functional Requirement

Functional Requirements

Functional Requirements (FR)

Menggambarkan fungsionalitas sistem atau layanan- layanan system. Sangat bergantung dari jenis perangkat lunak, pengguna sistem, dan jenis sistem dimana perangkat lunak tersebut digunakan.

Kebutuhan fungsional dapat berupa pernyataan-pernyataan tingkat tinggi dari: Apa yang sistem harus lakukan. Harus dapat menggambarkan layanan-layanan yang dapat diberikan oleh sistem kepada pengguna secara mendetail

Functional Requirements

Contoh Sistem Perpustakaan

Sistem perpustakaan menyediakan antarmuka tunggal untuk mengakses artikel-artikel dalam database perpustakaan yang berbeda-beda. Pengguna dapat mencari, men-download, dan mencetak artikel yang ditampilkan.

Contoh FR sistem perpustakaan:

- Pengguna harus dapat mencari di seluruh database yang ada, atau mencari di sebagian database yang disediakan (filtering).
- Sistem harus menyediakan program yang memungkinkan penggunanya membaca artikel atau dokumen dalam sistem perpustakaan.
- Sistem harus dapat mencatat buku-buku atau dokumen yang dipinjam oleh pengguna.

Non-functional Requirements (NFR)

Non-functional requirements atau kebutuhan non-fungsional menentukan atribut atau kualitas secara keseluruhan dari suatu sistem.

Kebutuhan non-fungsional menempatkan batasan pada produk yang sedang dikembangkan, proses pengembangannya, dan menentukan batasan-batasan eksternal yang harus dipenuhi oleh produk tersebut.

Menentukan Quality Attribute yang harus dimiliki oleh Sistem yang kita bangun.

- Beberapa Kategori NFR
 - Keamanan (safety dan security)
 - Ketergunaan (usability)
 - Reliabilitas
 - Performansi

 "Sistem harus dapat memastikan bahwa data yang digunakan dalam sistem harus terlindung dari akses yang tidak berwenang."

Dengan kata lain, data tidak dapat diakses oleh pengguna yang tidak berhak. Secara konvensional kebutuhan ini termasuk kebutuhan non-fungsional, karena tidak menyebutkan secara spesifik kebutuhan fungsional yang harus disediakan oleh sistem.

Namun dapat dispesifikasikan lebih lanjut dengan:

"Sistem harus menyertakan sebuah prosedur otorisasi dimana penggunanya harus mengidentifikasi diri dengan sebuah username dan password. Hanya pengguna yang memiliki wewenang melalui prosedur ini yang dapat mengakses data dalam sistem."

Jenis-jenis NFR (IEEE-Std 830-1993)

- Performance
- Interface/Usability
- Operational
- Resource
- Verification
- Acceptance
- Documentation
- Security

- Portability
- Quality
- Reliability
- Maintainability
- Safety

*critical system NFR

 Contoh NFR Sistem layanan X harus memiliki tingkat ketersediaan 999/1000 atau 99%.

Hal ini merupakan kebutuhan akan reliabilitas yang berarti setiap 1000 permintaan layanan, 999 permintaan harus dapat terpenuhi.

 Sistem Y harus mampu mengolah transaksi sekurang-kurangnya 8 transaksi dalam setiap detik.

Hal ini merupakan sebuah kebutuhan akan performansi.

Besarnya program sistem Z dibatasi sebesar 512 Kbytes.

Hal ini merupakan kebutuhan sumber daya (resource) yang menentukan ukuran memory maksimum sistem tersebut.

- Mengidentifikasi NFR Umumnya NFR dapat dikenali dari "keinginan", "kebutuhan", atau "kepentingan" dari para stakeholders.
- Keinginan, kebutuhan, atau kepentingan tersebut umumnya bersifat non-fungsional:
 - Tujuan bisnis
 - Karakteristik sistem
 - Keamanan, performa, fungsionalitas, dan perawatan sistem

Hubungan antara kebutuhan user, kepentingan, dan NFR

Kebutuhan	Kepentingan	NFR
Fungsi	 Kemudahan penggunaan Otoritas Akses Kemungkinan kegagalan sistem 	 Usability Security Reliability
Performansi	 Penggunaan sumber daya Verifikasi performa Kemudahan antarmuka 	 Efficiency Verifiability Interoperability
Perubahan	 Kemudahan perbaikan Kemudahan perubahan Kemudahan transport/lokasi penggunaan Kemudahan dalam meningkatkan kapasitas dan performa 	 Maintainability Flexibility Portability Expandability

Reliability

• Memberikan batasan perilaku sistem pada saat beroperasi

Availability - Ketersediaan sistem dalam memberikan layanan ketika diperlukan oleh pengguna.

Tingkat kegagalan - Seberapa sering sistem gagal untuk dapat memberikan layanan yang diharapkan oleh pengguna.

Performance

- Memberikan batasan mengenai kecepatan operasional sebuah sistem
 - Kebutuhan akan respon
 - Kebutuhan throughput (keluaran)
 - Kebutuhan akan pewaktuan (timing)

Security

- Kebutuhan akan keamanan disertakan dalam sistem untuk menjamin:
 - Akses ke dalam sistem tanpa otorisasi tidak dimungkinkan
 - Memastikan integritas sistem dari kecelakaan atau kerusakan
- Contoh:
 - Data hanya dapat diubah oleh administrator sistem
 - Seluruh data harus di-backup setiap 24 jam, dan hasil backup-nya disimpan di lokasi yang berbeda dengan sistem
 - Seluruh komunikasi antara client-server harus dienkripsi

Usability

- Terkait dengan penentuan antarmuka dan interaksi pengguna dengan sistem.
- User manual yang terstruktur, pesan kesalahan yang informatif, fasilitas pertolongan, dan antarmuka yang konsisten dapat meningkatkan kebutuhan usability ini.

Safety

- Umumnya diasumsikan sebagai seluruh kebutuhan yang terkait dengan keamanan sistem
- Umumnya sangat berhubungan dengan kebutuhan untuk memastikan keamanan dalam operasional sistem
 - Kebutuhan untuk melindungi sistem
 - Kebutuhan untuk menghindari kecelakaan dalam sistem maupun dalam menggunakan sistem.
- Penggunaannya seringkali bergantung dengan budaya dan perilaku dalam organisasi

Contoh Kebutuhan Safety

- Sistem tidak mengizinkan pengoperasian alat/perangkat kecuali terdapat petugas di lokasi.
- Sistem tidak boleh memberikan obat kepada pasien dengan dosis yang lebih dari yang diizinkan oleh dokter pasien yang bersangkutan.
- Sistem tidak boleh beroperasi jika suhu di luar ruangan berada di bawah 4 derajat Celcius.
- Petugas yang melaksanaan proses perawatan sistem harus menggunakan gelang anti-statik selama berada di dalam ruangan.

Petunjuk Penulisan Requirements

- Gunakan format yang standar dan gunakan pada seluruh kebutuhan yang dituliskan.
- Gunakan bahasa yang konsisten dan tidak membingungkan atau ambigu.
- Gunakan kata "harus" pada kondisi kebutuhan yang memang harus dipenuhi oleh sistem, dan gunakan kata "seharusnya/sebaiknya" untuk menuliskan kondisi kebutuhan yang diinginkan.
- Hindari penggunaan bahasa yang terlalu bersifat teknis dalam penulisan requirements.

Be S.M.A.R.T

- Specific
 - Spesifik dan detail
- **▶** Measurable
 - Dapat diukur
- ▶ **A**ttainable
 - Dapat dicapai
- ▶ Realistic
 - Realistis
- Tangible
 - Berwujud atau dapat diwujudkan

Tabel Functional Requirement

User Story	Functional Requirements & Priorities			
	High Priority (i.e. Must have)	Medium Priority (i.e. Should have)	Low Priority (i.e. Could have)	No Priority (i.e. Won't have)
I (Angie) as the director of Latinitas would like to be able to export donor history data (name, data, amount, payment type) so that I can report to my board on increases and decreases in donor dollars.	 Create Donor Table to store info about donor Create Donation Table used to store donations but Donor and Date. Build report query that pulls all donors and donation info Report must be able to run via a button on a Reports page that only admin people have access to see 	Allow report to run with filter criteria Allow user to save preset settings on report filter criteria to speed up running reports	Have reports be automated so they run via a nightly program that emails report to admins	Reports to be accessible via mobile devices

Tabel Non Functional Requirement

Quality attribute	Requirement definition	Scope/How
Scalability	The number of concurrent players in the game is $2 - n$.	Client-Server architecture style Component Game Server at server side
Portability	The game can be played with devices supporting either GPRS or UMTS connection.	Client application. This requirement is realized by component Communication Manager
	End-user devices support: PalmOS, EPOC, WinCE.	Layered architecture style Client application includes a virtual machine layer
Extendibility	Features can be enriched, hence ensuring service evolution (in the WISE project, evolution is simulated by carrying out three development iterations).	Extension points in the architecture and code. - On the Server side e.g. quest sending, fights & attacks, manage high score list - On the Client side e.g. quest receiving and handling, fights/attacks/defending, buy/sell items
Modifiability	Services should be easily modified under the evolution of mobile terminals' hardware capabilities.	Client application. By separating communication manipulation and game management, and by splitting logically related functionality, modification is easier. Of course any modification is isolated if interfaces are not influenced.

Studi Kasus

PT. Bendi Car adalah sebuah perusahaan yang bergerak dibidang penyewaan mobil Semua transaksi di perusahaan masih dilakukan secara manual. Berikut ini adalah kegiatan kegiatan yang dilakukan oleh petugas dalam melaksanakan transaksi penyewaan mobil di dalam perusahaan.

Studi Kasus

1. Prosedure peminjaman

Penyewa mobil yang ingin melakukan peminjaman dapat melihat harga penyewaan mobil pada daftar harga sewa mobil. Penyewa dapat menggunakan jasa supir atau tidak sesuai dengan kebutuhan penyewa sendiri. Setiap jenis kendaraan memiliki harga sewa yang berbeda-beda begitu juga harga sewa jasa sopir untuk daerah Jabodetabek dan diluar jabodetabek pun berbeda. Setelah itu penyewa mengisi Formulir Penyewaan (FS) disertai fotocopy identitas diri. Kemudian Formulir penyewaan yang telah diisi beserta pembayaran dimuka diserahkan kepada petugas kemudian petugas membuatkan kwitansi pembayaran sebagai bukti pembayaran.

2. Prosedure pengembalian

Pada saat pengembalian kendaraan oleh penyewa, petugas membawa Formulir Pengembalian dan memeriksa kondisi kendaraan apakah terdapat kerusakan atau tidak. Bila ada (misalnya spion pecah, body penyok, cat tergores, dll), maka diperhitungkan penggantiannya dan dibebankan kepada penyewa. Bila tidak rusak/terlambat maka tidak perlu mengisi Formulir pengembalian. Namun Bila penyewa terlambat dalam pengembalian, maka jumlah keterlambatan mobil dan sopir akan dibebankan kepada penyewa. Setelah membayar kerusakan dan keterlambatan, maka petugas membuatkan kwitansi sebagai tanda bukti pembayaran denda.

3. Prosedure laporan

Diakhir bulan petugas membuat laporan penyewaan berikut denda atas kerusakan atau keterlambatan yang terjadi dan laporan kendaraan. Laporan tersebut diserahkan ke pemilik penyewaan Bendi car.

Tugas: Buatlah FRS dan NFRS dari sistem diatas!
Emailkan Tugas Tersebut ke danny sttb. tugas@gmail.com dengan subject:
OOAD-KELAS-NAMA-TUGAS03