

PERTEMUAN KE-3

ANALISA KEBUTUHAN

3.1 TUJUAN PEMBELAJARAN

Adapun tujuan pembelajaran yang akan dicapai sebagai berikut:

- 3.1. Dapat Melakukan Analisa Kebutuhan
- 3.2. Mengetahui Kebutuhan fungsional dan non-fungsional

3.2 URAIAN MATERI

Tujuan Pembelajaran 3.1:

Analisa kebutuhan

Analisis kebutuhan perangkat lunak (software requirements analysis) merupakan aktivitas awal dari siklus hidup pengembangan perangkat lunak. Untuk proyek-proyek perangkat lunak yang besar, analisis kebutuhan dilaksanakan setelah aktivitas sistem information engineering dan software project planning.

Tahap analisis adalah tahapan pengumpulan kebutuhan-kebutuhan dari semua elemen sistem perangkat lunak yang akan di bangun. Pada tahap ini dibentuk spesifikasi kebutuhan perangkat lunak, fungsi perangkat lunak yang dibutuhkan, performansi (unjuk kerja) sistem perangkat lunak, penjadwalan proyek, identifikasi sumber daya (manusia , perangkat keras dan perangkat lunak yang dibutuhkan) dan taksiran biaya pengembangan perangkat lunak.

Kegunaan analisis adalah untuk memodelkan permasalahan dunia nyata agar dapat dimengerti. Permasalahan dunia nyata harus dimengerti dan dipelajari supaya spesifikasi kebutuhan perangkat lunak dapat diungkapkan. Tujuan aktivitas ini adalah untuk mengetahui ruang lingkup produk (product space) dan pemakai yang akan menggunakannya. Analisis yang baik akan mengungkapkan hal-hal yang penting dari permasalahan, dan mengabaikan yang tidak penting.

Setiap metode analisis mempunyai pandangan yang berbeda. Tetapi pada dasarnya semua metode analisis memiliki prinsip analisis yang sama, yaitu:

1. Menggambarkan domain informasi masalah
2. Mendefinisikan fungsi perangkat lunak
3. Menghasilkan model yang menggambarkan informasi, fungsi dan kelakuan yang dibagi secara rinci pada sebuah model lapisan (hirarki)
4. Informasi pokok pada tahap analisis memudahkan tahap implementasi yang lebih rinci.

Tujuan tahap analisis adalah :

1. Menjabarkan kebutuhan pemakai
2. Meletakkan dasar-dasar untuk tahap perancangan perangkat lunak
3. Mendefinisikan semua kebutuhan pemakai sesuai dengan lingkup kontrak yang disepakati kedua belah pihak (pengembang dan pengguna)

Tujuan Pembelajaran 3.2:

Kebutuhan fungsional dan non-fungsional

Definisi Kebutuhan Fungsional Dan Non-Fungsional adalah sebagai berikut:

1. Kebutuhan Fungsional

Kebutuhan fungsional adalah pernyataan layanan sistem yang harus disediakan, bagaimana sistem bereaksi pada input tertentu dan bagaimana perilaku sistem pada situasi tertentu. Sedangkan kebutuhan fungsional user merupakan pernyataan level tinggi dari apa yang seharusnya dilakukan sistem tetapi kebutuhan fungsional sistem menggambarkan layanan sistem secara detail.

Mendeskripsikan layanan, fitur atau fungsi yang disediakan atau diberikan oleh sistem bagi penggunaanya. Kebutuhan fungsional awal merupakan fungsi atau layanan yang merepresentasikan goal dari pengguna ketika hendak menggunakan sistem.

Contoh pada Sistem Mesin ATM :

- Mengecek saldo
- Menarik uang

- Mentransfer uang
- Melakukan pembayaran

2. Kebutuhan Non – Fungsional

Kebutuhan non – fungsional adalah batasan layanan atau fungsi yang ditawarkan sistem seperti batasan waktu, batasan pengembangan proses, standarisasi dll. Kebutuhan non-fungsional lebih kritis daripada kebutuhan fungsional. Jika tidak dapat bertemu, sistem menjadi tidak berguna.

Mendesripsikan sekumpulan batasan, karakteristik dan properti pada sistem, baik dalam lingkungan pengembangan maupun operasional, atau atribut kualitas yang harus dipenuhi oleh sistem. Contoh pada mesin ATM:

- Pengguna baru membutuhkan waktu belajar maksimal 10 menit untuk dapat menggunakan fungsi-fungsi utama sistem
- Sistem harus tetap berfungsi minimal 10 jam setelah pasokan listrik dari PLN terhenti
- Waktu yang dibutuhkan untuk kembali beroperasi setelah sistem mati minimal 2 menit

3.3 LATIHAN SOAL/TUGAS

1. Buatlah daftar kebutuhan fungsional dan nonfungsional sistem pengelolaan perpustakaan

3.4 DAFTAR PUSTAKA

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.