

BAB II

TINAJAUAN PUSTAKA

2.1 Pengertian *Internet*

Internet merupakan singkatan dari *Interconnection Networking*. *Internet* berasal dari bahasa latin “inter” yang berarti antara. Secara kata perkata *INTERNET* berarti jaringan antara atau penghubung, sehingga kesimpulan dari defenisi internet ialah merupakan hubungan antara berbagai jenis komputer dan jaringan didunia yang berbeda sistem operasi maupun aplikasinya dimana hubungan tersebut memanfaatkan kemajuan komunikasi (telepon dan satelit) yang menggunakan protokol standar dalam berkomunikasi yaitu protokol TCP/IP (*Transmission Control/Internet Protocol*).[1]

Istilah *internet* berasal dari bahasa latin, *inter* yang berarti “antara”. Secara kata per kata internet berarti jaringan antara atau penghubung. Memang itulah fungsinya, internet menghubungkan berbagai jaringan yang tidak saling bergantung pada satu sama lain dengan sedemikian rupa sehingga mereka dapat berkomunikasi. *Internet* adalah jaringan komputer yang saling terhubung keseluruh dunia tanpa mengenal batasan teritorial, hukum dan budaya.[2]

Menurut Lani Sudharto (1996) walaupun secara fisik internet adalah *inter* koneksi antar jaringan komputer namun secara umum internet harus dipandang sebagai sumber daya informasi. *Protocol TCP / IP* menyatukan bahasa dan kode berbagai komputer didunia sehingga menjadi *standart* utama jaringan komputer. *TCP / IP* berkembang cepat dan kaya fasilitas karena bersifat terbuka dan bebas digunakan, ditambahkan kemampuan baru oleh siapapun dan gratis karena tidak dimiliki siapapun.[2]

2.2 Pengertian *Website*

Website adalah salah satu aplikasi yang berisikan dokumen-dokumen multimedia (teks, gambar, suara, animasi, dan video) didalamnya yang menggunakan protokol *HTTP (hyper text transfer protocol)* dan untuk mengaksesnya menggunakan perangkat lunak yang disebut *browser*. Beberapa jenis browser yang populer saat ini diantaranya : *Internet Explorer* yang diproduksi oleh *Microsoft*, *Mozilla Firefox*, *Ipera* dan *Safari* yang diproduksi oleh *Apple*. *Browser* (perambah) adalah aplikasi yang mampu menjalankan dokumendokumen web dengan cara diterjemahkan. Prosesnya dilakukan oleh komponen yang terdapat didalam aplikasi

browser yang biasa disebut *web engine*. Semua dokumen *web engine*. Semua dokumen web ditampilkan dengan cara diterjemahkan.[3]

Website adalah sebagai kumpulan halaman-halaman yang digunakan untuk menampilkan informasi teks, gambar diam atau gerak, animasi, suara dan atau gabungan dari semuanya baik yang bersifat statis maupun dinamis yang membentuk suatu rangkaian bangunan yang saling terkait dimana masing-masing dihubungkan dengan jaringan-jaringan halaman (*Hyperlink*).[4]

2.3 Komunitas

Komunitas adalah sekelompok orang yang saling peduli satu sama lain lebih dari yang seharusnya, dimana dalam sebuah komunitas terjadi relasi pribadi yang erat antar para anggota komunitas tersebut karena adanya kesamaan interest atau values.[5]

Komunitas dapat terbagi menjadi 3 komponen :

a. Berdasarkan Lokasi atau Tempat

Wilayah atau tempat sebuah komunitas dapat dilihat sebagai tempat dimana sekumpulan orang mempunyai sesuatu yang sama secara geografis.

b. Berdasarkan Minat

Sekelompok orang yang mendirikan suatu komunitas karena mempunyai ketertarikan dan minat yang sama, misalnya agama, pekerjaan, suku, ras, maupun berdasarkan kelainan seksual.

c. Berdasarkan Komuni

Komuni dapat berarti ide dasar yang dapat mendukung komunitas itu sendiri.

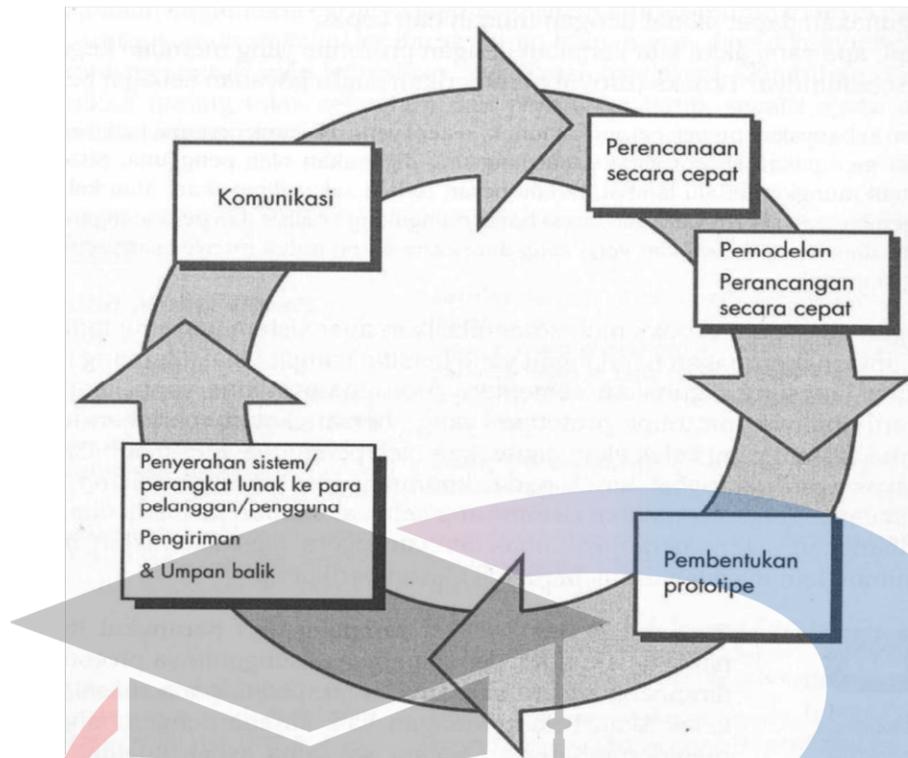
2.4 Metodologi *Prototype*

Sering kali pelanggan mendefinisikan sejumlah sasaran perangkat lunak secara umum, tetapi tidak bisa mengidentifikasi spesifikasi kebutuhan yang rinci untuk fungsi-fungsi dan fitur-fitur yang nantinya akan dimiliki perangkat lunak yang akan di kembangkan. Dalam kasus yang lain, pengembang perangkat lunak mungkin merasa tidak pasti tentang efisiensi suatu algoritma yang akan digunakan dalam pengembangan perangkat lunak, atau juga merasa tidak pasti akan kemampuan perangkat lunak untuk beradaptasi dengan sistem operasi yang akan digunakan, atau merasa tidak pasti akan bentuk interaksi manusia-komputer yang digunakan. Dalam kasus seperti ini dan dalam banyak situasi yang lain, paradigma pembuatan prototipe (*prototyping*) mungkin menawarkan pendekatan yang paling baik.[6]

Proses pengembangan sistem seringkali menggunakan pendekatan prototip (*prototyping*). Metode ini sangat baik digunakan untuk menyelesaikan masalah kesalahpahaman antara *user* dan analis yang timbul akibat *user* tidak mampu mendefinisikan secara jelas kebutuhannya.[7]

Paradigma *prototyping* dimulai dengan komunikasi antara tim pengembang perangkat lunak dengan para pelanggan. Tim pengembang perangkat lunak akan melakukan pertemuan-pertemuan dengan *stakeholder* untuk mendefinisikan sasaran keseluruhan untuk perangkat lunak yang akan dikembangkan, mengidentifikasi spesifikasi kebutuhan apapun yang saat ini diketahui, dan menggambarkan area-area dimana definisi lebih jauh pada iterasi selanjutnya merupakan keharusan. Iterasi pembuatan prototipe direncanakan dengan cepat dan pemodelan (dalam bentuk “rancangan cepat”) dilakukan. Suatu rancangan cepat berfokus pada representasi semua aspek perangkat lunak yang terlihat oleh para pengguna akhir (misalnya rancangan antarmuka pengguna [*user interface*] atau format tampilan). Rancangan cepat (*quick design*) akan memulai konstruksi pembuatan *prototyping*. *Prototyping* kemudian akan diserahkan kepada para *stakeholder* dan kemudian mereka akan melakukan evaluasi-evaluasi tertentu terhadap *prototyping* yang telah dibuat sebelumnya, kemudian akhirnya akan memberikan umpan balik yang akan digunakan untuk memperhalus spesifikasi kebutuhan. Iterasi akan terjadi saat *prototyping* diperbaiki untuk memenuhi kebutuhan dari para *stakeholder*, sementara pada saat yang sama memungkinkan kita untuk lebih memahami kebutuhan apa yang akan dikerjakan pada iterasi selanjutnya. Adapun tahapan yang ada dapat dilihat pada gambar dibawah berikut ini.[6]

UNIVERSITAS
MIKROSKIL



Gambar 2. 1 Pengembangan Perangkat Lunak dengan Metodologi Prototipe [6]

Prototyping adalah pengembangan yang cepat dan pengujian terhadap model kerja (prototipe) dari aplikasi baru melalui proses interaksi dan berulang yang biasa digunakan ahli sistem informasi dan ahli bisnis. *Prototype* menyederhanakan dan mempercepat desain sistem.[8]

Adapun kelebihan *Prototyping* adalah sebagai berikut :

- a. Adanya komunikasi yang baik antara pengembang dan pelanggan
- b. Pengembang dapat bekerja lebih baik dalam menentukan kebutuhan pelanggan
- c. Pelanggan berperan aktif dalam pengembangan sistem
- d. Lebih menghemat waktu dalam pengembangan sistem
- e. Penerapan menjadi lebih mudah karena pemakai mengetahui apa yang diharapkannya.[9]

Adapun kelemahan *Prototyping* adalah sebagai berikut :

- a. Pelanggan kadang tidak melihat atau menyadari bahwa perangkat lunak yang ada belum mencantumkan kualitas perangkat lunak secara keseluruhan dan juga belum memikirkan kemampuan pemeliharaan untuk jangka waktu lama.
- b. Pengembang biasanya ingin cepat menyelesaikan proyek. Sehingga menggunakan algoritma dan bahasa pemrograman yang sederhana untuk membuat *prototyping* lebih cepat selesai tanpa memikirkan lebih lanjut bahwa program tersebut hanya merupakan cetak biru sistem .
- c. Hubungan pelanggan dengan komputer yang disediakan mungkin tidak mencerminkan teknik perancangan yang baik.[9]

2.5 Use Case Diagram

Use case diagram adalah teknik untuk merekam persyaratan fungsional sebuah sistem. *Use case diagram* mendeskripsikan interaksi tipikal antara para pengguna sistem dengan sistem itu sendiri, dengan memberi sebuah narasi tentang bagaimana sistem tersebut digunakan.[10]

Use case diagram penting untuk memvisualisasikan, menspesifikasikan, dan mendokumentasikan kebutuhan perilaku sistem.. *Use case diagram* merupakan pusat pemodelan perilaku sistem, subsistem, dan kelas.

Use case diagram untuk mendeskripsikan apa yang seharusnya dilakukan oleh sistem. *Use case diagram* menyediakan cara mendeskripsikan pandangan eksternal terhadap sistem dan interaksi-interaksinya dengan dunia luar. Dengan cara ini, *Use case diagram* menggantikan diagram konteks pada pendekatan konvensional.[11]

Elemen *use case diagram* adalah :

1. Aktor

Aktor adalah pemakai sistem, berupa manusia atau sistem terotomatisasi lain. aktor adalah tipe (kelas), bukan instan. Aktor merepresentasikan peran bukan pemakai individu dari sistem. Aktor mempunyai nama. Nama yang dipilih harus menyatakan peran aktor Aktor berkomunikasi dengan sistem lewat pengiriman dan penerimaan pesan. *Use case* selalu diawali oleh aktor yang mengirim pesan, yang disebut stimulus. Ketika *use case* dilakukan, *use case* dapat mengirim pesan ke satu aktor atau lebih.[11]

2. Use Case

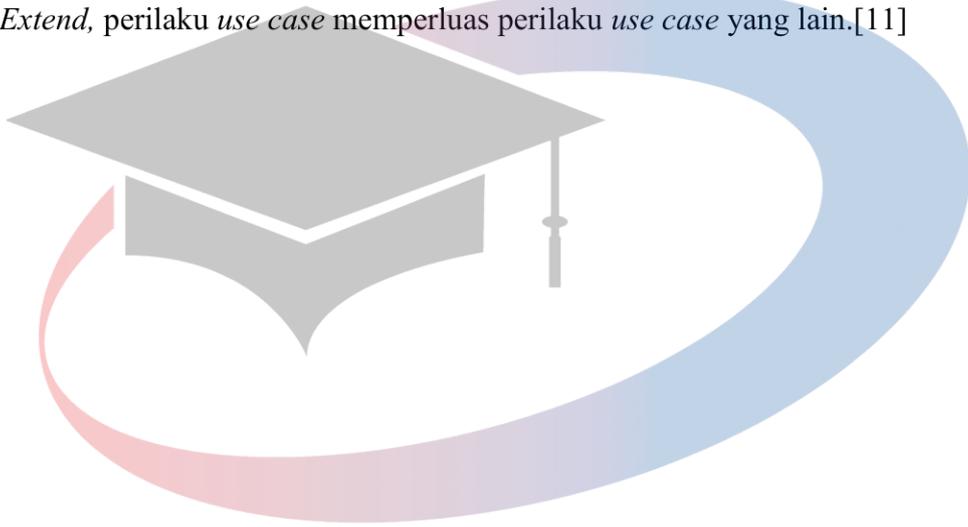
Use Case adalah cara spesifik penggunaan sistem oleh aktor. *Use Case* memspesifikasikan perilaku sistem atau bagian sistem dan merupakan deskripsi

sekumpulan sekuen aksi termasuk varian-varian yang dilakukan sistem untuk memproduksi hasil atau nilai ke aktor *Use case* dapat digunakan untuk memodelkan perilaku sistem yang dikehendaki sekaligus untuk mengembangkan kasus pengujian.[11]

3. Hubungan ketergantungan, generalisasi, dan asosiasi

Hubungan saling mempengaruhi antar *use case* dinamakan hubungan ketergantungan, yaitu *use* dimana *use case* yang satu bergantung pada *use case* lainnya. Hubungan antar aktor dinamakan asosiasi. Hubungan antar *use case* dengan *use case* lain dinamakan generalisasi, yang terdiri dari :

- a. *Include*, perilaku *use case* merupakan bagian dari *use case* yang lain
- b. *Extend*, perilaku *use case* memperluas perilaku *use case* yang lain.[11]



UNIVERSITAS MIKROSKIL