

BAB II

TINJAUAN PUSTAKA DAN LANDASAN TEORI

2.1. Tinjauan Pustaka

Mengutip dari jurnal yang dibuat oleh Dewi Agustina dan Dyah Pratiwi (2008) dengan judul “Interaksi Manusia dengan Komputer” menyatakan bahwa *human computer interaction* mempelajari hubungan interaksi antara manusia, komputer, dan penugasan. Prinsipnya bagaimana manusia dan komputer dapat secara interaktif menyelesaikan penugasan dan bagaimana sistem yang interaktif tersebut dapat dibuat.

Sebuah jurnal yang dibuat oleh Prihati, Mustad, suhartono (2011) membuat sebuah jurnal yang berjudul “Penerapan Model *Human Computer Interaction* (HCI) dalam Analisis Sistem Informasi”. Jurnal yang bertujuan merancang dan menerapkan model analisis sistem informasi dengan konsep HCI. Jurnal ini menjelaskan bahwa sering terjadinya salah persepsi manusia (pengguna) terhadap *software* yang ada, sehingga bukan efektifitas dan efisiensi kerja yang diperoleh, akan tetapi justru menyebabkan pekerjaan efisien dan efektif, pengguna sering mengalami kesulitan menggunakan *software* tersebut karena tidak familiar dengan *software*, sehingga *software* sulit untuk dipelajari. Akibat dari penerapan sistem, dihindari dengan cara menerapkan konsep HCI dengan baik. Dengan memperhatikan HCI maka dapat diciptakan suatu sistem yang sesuai dan tepat bagi pengguna. Sistem yang tepat bagi pengguna akan memberikan kenyamanan.

Pembuatan aplikasi yang menerapkan HCI juga pernah dilakukan sebelumnya dalam sebuah skripsi yang berjudul “Penerapan Konsep Interaksi

Manusia dan Komputer pada Sistem Informasi Akademik Fakultas Ekonomi Universitas Musi Rawas” yang bertujuan untuk merancang antarmuka sistem informasi akademik dengan konsep IMK sehingga dapat direkomendasikan untuk dipergunakan membantu proses pengolahan data khususnya data akademik. Pada jurnal tersebut dijelaskan juga bahwa sistem informasi yang baik adalah yang mampu memberikan kenyamanan kepada penggunanya sehingga pengguna akan terus menggunakan sistem tersebut. Maka dari itu dalam merancang sistem informasi perlu memperhatikan teori dasar interaksi manusia dan komputer yaitu dalam kegunaan, keamanan, fungsional, sistem, dan efektifitas serta efisiensi (Syafri, 2014).

Penerapan HCI pada aplikasi juga pernah dibuat dalam jurnal yang berjudul “Tinjauan *Human Computer Interaction* (HCI) Terhadap Aplikasi Penjadwalan Sekolah” yang bertujuan untuk merancang dan menerapkan konsep HCI pada aplikasi penjadwalan sekolah. Pada jurnal tersebut dijelaskan bahwa dalam merancang sebuah aplikasi ada beberapa hal yang perlu ditimbangkan seperti bagaimana membuat aplikasi tersebut menjadi nyaman dalam interaksinya serta bagaimana aplikasi mudah digunakan (Indayah Hartani Santi, 2015).

Dari beberapa penelitian sebelumnya, konsep *human computer interface* (HCI) digunakan oleh para peneliti agar aplikasi yang telah dirancang nyaman dan mudah untuk digunakan oleh pengguna atau *user*. Begitu juga dengan penelitian ini untuk meningkatkan *user friendly* yang ada pada aplikasi. Adapun penelitian menggunakan metode delapan aturan emas atau *eight golden rules*.

2.2. Landasan Teori

2.2.1. Logistik

Logistik merupakan seni dan ilmu, barang, energi, informasi, dan sumber daya lainnya, seperti produk, dan jasa dari sumber produksi ke pasar dengan tujuan mengoptimalkan penggunaan modal. Manufaktur dan marketing akan sulit dilakukan tanpa dukungan logistik. Logistik juga mencakup integritas informasi, transportasi, inventori, pergudangan, *reverse logistics* dan pemaketan. Secara etimologi, logistik berasal dari bahasa Yunani kuno yang terdiri dari dua suku kata, yaitu “*Logic*” yang berarti rasional, masuk akal dan dapat dipertanggungjawabkan. Suku kata yang kedua adalah “*Thios*” yang berarti berpikir. Jika arti kedua suku kata dirangkai, memiliki makna berpikir rasional, masuk akal dan dapat dipertanggungjawabkan (Sutarman, 2005). Seiring berkembangnya jaman, arti logistik mengalami pergeseran. Logistik merupakan keseluruhan dari bahan, barang, alat dan sarana yang diperlakukan dan dipergunakan oleh suatu organisasi dalam rangka pencapaian tujuan dan berbagai sasarnya (Siagian, 2003).

Pendapat di atas diperkuat dengan pendapat (Dwiantara dan Rumsari, 2004) yang menyebutkan “Logistik adalah segala sesuatu atau benda yang berwujud dan dapat diperlakukan secara fisik (*tangible*), baik yang digunakan untuk menyelenggarakan kegiatan pokok maupun kegiatan penunjang (*administrasi*)”. Sementara (Yolanda Siagian, 2005) melihat logistik dari segi dunia bisnis yakni “Logistik merupakan bagian dari proses rantai suplai yang berfungsi merencanakan, melaksanakan, mengontrol secara efektif, efisien proses

pengadaan, pengelolaan, penyimpanan barang, pelayanan dan informasi mulai dari titik awal (*point of origin*) hingga titik konsumsi (*point of consumption*) dengan tujuan memenuhi kebutuhan konsumen. Dengan demikian logistik adalah segala sesuatu baik itu berupa bahan, barang, alat, atau sarana yang digunakan untuk membantu kegiatan organisasi dalam rangka pencapaian tujuan.

2.2.2. Logistik Pada Skala Rumah Tangga

Pada skala rumah tangga, logistik merupakan pengadaan kebutuhan rumah tangga secara rutin yang dapat direncanakan dan distrukturkan agar dapat lebih efisien. Daftar kebutuhan rumah tangga dapat diklasifikasikan menjadi 3 bagian utama:

A. Pangan:

- Bahan segar: Daging, ikan, ayam, sayuran, bumbu, telur, buah dan sebagainya.
- Bahan olahan: Makan beku seperti baso, nugget, sosis, makanan kering seperti ikan asin, kembang tahu, makanan kaleng seperti sarden, kornet dan, buah, makanan instan seperti mie dan pasta.
- Pelengkap: Kecap, sambal, saus, *dressing*, gula, garam, merica, coklat, selai, minyak goreng, dan sebagainya.
- Bahan jadi: Susu, mentega, keju, roti, kopi, sereal dan sebagainya.
- Makanan pokok: Beras, tepung, gandum dan sebagainya.

B. Perawatan tubuh:

- Perlengkapan mandi: Sabun, shampoo, odol dan sebagainya.
- Kosmetik: Bedak, *lipstick*, *lotion* dan sebagainya.

C. Perawatan rumah:

- Laundry: deterjen, pelembut, dan *spray* setrika.
- Alat dapur: deterjen, spons, dan sebagainya.
- Pembersih rumah: cairan pel, *spray* pembersih mebel, pembersih kaya, dan sebagainya.

2.2.3. Industri

Industri sering diidentikkan dengan semua kegiatan ekonomi manusia yang mengolah barang mentah atau bahan baku menjadi barang setengah jadi atau barang jadi. Dari definisi tersebut, istilah industri sering disebut sebagai kegiatan manufaktur (*manufacturing*). Padahal, pengertian industri sangatlah luas, yaitu menyangkut semua kegiatan manusia dalam bidang ekonomi yang sifatnya produktif dan komersial.

Karena merupakan kegiatan ekonomi yang luas maka jumlah dan macam industri berbeda-beda untuk tiap negara atau daerah. Pada umumnya, makin maju tingkat perkembangan perindustrian di suatu negara atau daerah, makin banyak jumlah macam industri, dan kompleks pula sifat kegiatan dan usaha tersebut. Cara penggolongan atau klasifikasian industri pun berbeda-beda. Tetapi pada dasarnya, pengklasifikasian industri didasarkan pada kriteria yaitu berdasarkan bahan baku, tenaga kerja, pangsa pasar, modal, atau jenis teknologi yang digunakan. Selain faktor-faktor tersebut, perkembangan dan pertumbuhan ekonomi suatu negara juga turut menentukan keanekaragaman industri negara tersebut, semakin besar dan kompleks kebutuhan masyarakat yang harus dipenuhi, maka semakin beranekaragam jenis industrinya.

Adapun klasifikasi industri berdasarkan kriteria masing-masing. Adalah sebagai berikut:

A. Klasifikasi industri berdasarkan bahan baku

Tiap-tiap industri membutuhkan bahan baku yang berbeda, tergantung pada apa yang akan dihasilkan dari proses industri tersebut. Berdasarkan bahan baku yang digunakan, industri dapat dibedakan menjadi:

- a. Industri ekstraktif, yaitu industri yang bahan bakunya diperoleh langsung dari alam. Misalnya: industri hasil pertanian, industri hasil perikanan, dan industri hasil kehutanan.
- b. Industri nonekstraktif, yaitu industri yang mengolah lebih lanjut hasil industri lain. Misalnya: industri kayu lapis, industri permintalan, dan industri kain.
- c. Industri fasilitatif atau disebut juga industri tertier. Kegiatan industrinya adalah dengan menjual jasa layanan untuk keperluan orang lain. Misalnya: perbankan, perdagangan, angkutan, dan pariwisata.

B. Klasifikasi industri berdasarkan tenaga kerja

Berdasarkan jumlah tenaga kerja yang digunakan, industri dapat dibedakan menjadi:

- a. Industri rumah tangga, yaitu industri yang menggunakan tenaga kerja kurang dari empat orang. Ciri industri ini memiliki modal yang sangat terbatas, tenaga kerja berasal dari anggota keluarga, dan pemilik atau pengelola industri biasanya kepala rumah tangga itu sendiri atau anggota

keluarganya. Misalnya: industri anyaman, industri kerajinan, industri tempe/tahu, dan industri makanan ringan.

- b. Industri kecil, yaitu industri yang tenaga kerjanya berjumlah sekitar 5 sampai 19 orang. Ciri industri kecil adalah memiliki modal relative kecil, pekerjanya berasal dari lingkungan sekitar atau masih ada hubungan saudara. Misalny: industri genteng, industri batubata, dan industri pengolahan rotan.
 - c. Indsutri sedang yaitu industri yang menggunakan tenaga kerja sekitar 20 sampai 99 orang. Ciri industri sedang adalah memiliki modal yang cukup besar, tenaga kerja memiliki keterampilan tertentu, dan pimpinan perusahaan memiliki kemampuan manajerial tertentu. Misalnya: industri konveksi, industri bordir, dan industri keramik.
 - d. Industri besar, yaitu industri dengan jumlah tenaga kerja lebih dari 100 orang. Ciri industri besar adalah memiliki modal besar yang dihimpun secara kolektif dalam bentuk pemilikan saham, tenaga kerja harus memiliki keterampilan khusus, dan pimpinan perusahaan dipilih melalui uji kemampuan dan kelayakan (*fit and profer test*). Misalnya: industri tekstil, industri mobil, industri besi baja, dan industri pesawat terbang.
- C. Klasifikasi berdasarkan proses produksi
- Berdasarkan proses produksi, industri dapat dibedakan menjadi:
- a. Industri hulu, yaitu industri yang hanya mengolah bahan mentah menjadi barang setengah jadi. Industri ini sifatnya hanya menyediakan bahan baku

untuk kegiatan industri yang lain. Misalnya: industri kayu lapis, industri aluminium, industri pemintalan, dan industri baja.

- b. Industri hilir, yaitu industri yang mengolah barang setengah jadi menjadi barang jadi sehingga barang yang dihasilkan dapat langsung dipakai atau dinikmati oleh konsumen. Misalnya: industri pesawat terbang, industri konveksi, industri otomotif, dan industri meubeler.

D. Klasifikasi industri berdasarkan produksi yang dihasilkan

Berdasarkan produksi yang dihasilkan, industri dapat dibedakan menjadi:

- a. Industri premier, yaitu industri yang menghasilkan barang atau benda yang tidak perlu pengolahan lebih lanjut. Barang atau benda yang dihasilkan tersebut dapat dinikmati atau digunakan secara langsung. Misalnya: industri anyaman, industri konveksi, industri makanan dan minuman.
- b. Industri sekunder, yaitu industri yang menghasilkan barang atau benda yang membutuhkan pengolahan lebih lanjut sebelum dinikmati atau digunakan. Misalnya: industri pemintalan benang, industri ban, industri baja, dan industri tekstil.
- c. Industri tertier, yaitu industri yang hasilnya tidak berupa barang atau benda yang dapat dinikmati atau digunakan baik secara langsung maupun tidak langsung, melainkan berupa jasa layanan yang dapat mempermudah atau membantu kebutuhan masyarakat. Misalnya: industri angkutan, industri perbankan, industri perdagangan, dan industri pariwisata.

E. Klasifikasi industri berdasarkan barang yang dihasilkan

Berdasarkan barang yang dihasilkan, industri dibedakan menjadi:

- a. Industri ringan, yaitu industri yang menghasilkan barang siap pakai untuk dikonsumsi. Misalnya: industri obat-obatan, industri makanan, dan industri minuman.
- b. Industri berat, yaitu industri yang menghasilkan mesin-mesin atau alat produksi lainnya. Misalnya: industri alat-alat berat, industri mesin, dan industri percetakan.

2.2.4. Arsip

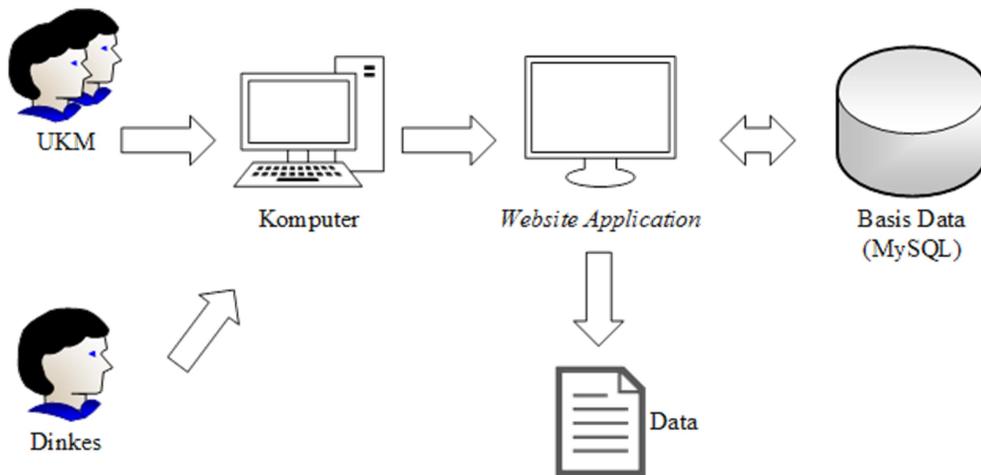
Arsip dapat diartikan sebagai suatu badan (*agency*) yang melakukan segala kegiatan pencatatan, penanganan, penyimpanan dan pemeliharaan surat-surat atau warkat-warkat yang mempunyai arti penting baik ke dalam maupun keluar, baik yang menyangkut soal-soal pemerintahan maupun non-pemerintahan, dengan menerapkan kebijaksanaan dan sistem tertentu yang dapat dipertanggungjawabkan (Basir Barthos 2009).

Kearsipan adalah setiap catatan tertulis atau bergambar yang memuat keterangan mengenai suatu hal atau peristiwa yang dibuat orang untuk membantu ingatannya (The Liang Gie 2000). Arsip merupakan keseluruhan dokumen-dokumen tertulis, lukisan-lukisan dan barang-barang cetakan yang secara resmi diterima atau dihasilkan oleh suatu badan pemerintahan atau salah seorang dari pejabat-pejabatnya sepanjang dokumen-dokumen itu di maksudkan untuk berada di bawah pemeliharaan dari badan itu atau pejabat itu.

Dapat dikatakan bahwa aktivitas pokok di bidang pencatatan ialah menyimpan data suatu barang atau produksi, tetapi tujuannya yang utama ialah menemukan kembali data yang diperlukan secara cepat.

2.2.5. Arsitektur perangkat lunak

Dalam mengembangkan aplikasi diperlukan perancangan arsitektur perangkat lunak yang bertujuan untuk menggambarkan bagaimana sistem dikembangkan dan dijalankan. Arsitektur perangkat lunak pada aplikasi dapat dilihat pada Gambar 2.1.



Gambar 2.1 Arsitektur Perangkat Lunak

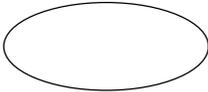
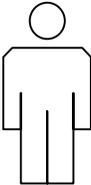
2.2.6. Unified Markup Language (UML)

Unified Markup Language (UML) merupakan bahasa pemodelan secara grafis yang digunakan untuk menspesifikasikan, memvisualisasikan, membangun, dan mendokumentasikan seluruh rancangan aplikasi perangkat lunak. Penggunaan model UML bertujuan untuk mengidentifikasi bagian-bagian yang termasuk dalam lingkup sistem di dalam aplikasi. Model UML yang dipakai dalam pengembangan aplikasi penjualan dan pembelian antara lain adalah *Use Case Diagram*, *Activity Diagram*.

a. *Use Case Diagram*

Use Case Diagram mendeskripsikan sebuah interaksi antara satu atau lebih actor dengan sistem yang dibuat. Dapat dikatakan *Use Case* digunakan untuk mengetahui fungsi yang ada di dalam sistem dan siapa saja yang berhak menggunakan fungsi-fungsi yang dibutuhkan. Simbol-simbol yang digunakan dalam *use case* diagram dapat dilihat pada Tabel 2.1 dan Tabel 2.2.

Tabel 2.1 Simbol-simbol dalam *Use Case Diagram*

No	Gambar	Nama Gambar	Keterangan
1.		<i>Use Case</i>	Merupakan fungsionalitas yang disediakan sistem sebagai unit yang bertukar pesan dengan <i>actor</i> .
2.		<i>Actor</i>	Merupakan <i>abstraction</i> dari orang yang mengaktifkan fungsi dari target sistem dan merupakan orang yang berinteraksi dengan <i>use case</i> .
3.		<i>Association</i>	Digambarkan dengan garis tanpa panah yang mengindikasikan siapa yang berinteraksi secara langsung dengan sistem.

Tabel 2.2 Simbol-simbol dalam *Use Case Diagram* (Lanjutan)

No	Gambar	Nama Gambar	Keterangan
4.		<i>Generalization</i>	Mengindikasikan siapa yang berinteraksi secara pasif dengan sistem.
5.	-<<include>>	<i>Include</i>	Mengidentifikasi hubungan antar dua <i>use case</i> dimana satu <i>usecase</i> memanggil <i>usecase</i> yang lain.
6.	-<<extend>>	<i>Extend</i>	Merupakan perluasan dari <i>use case</i> jika kondisi atau syarat terpenuhi.

b. *Activity Diagram*

Activity Diagram merupakan diagram yang digunakan untuk menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Simbol-simbol yang digunakan dalam *activity diagram* dapat dilihat pada tabel 2.3.

Tabel 2.3 Simbol-simbol dalam *Activity Diagram*

No	Gambar	Nama Gambar	Keterangan
1.		<i>End Point</i>	Merupakan akhir dalam aktifitas.

Tabel 2.4 Simbol-simbol dalam *Activity Diagram* (Lanjutan)

No	Gambar	Nama Gambar	Keterangan
2.		<i>Activities</i>	Menggambarkan suatu proses atau kegiatan bisnis
3.		<i>Decision Point</i>	Menggambarkan pilihan untuk pengambilan keputusan dalam aktifitas.
4.		<i>Fork Node</i>	Menggambarkan awalan dari percabangan aktifitas.

2.2.7. *User Interface*

User interface merupakan bagian dari sistem *computer interaktif* yang berkomunikasi langsung dengan pengguna. UI sendiri sudah semakin berkembang dengan porsi atau bagian yang lebih besar dari perangkat lunak pada sistem komputer karena semakin pesatnya orang menggunakan komputer. (A.Ralston, E.Reilly and D.Hemmeninger, 2000).

2.2.8. *Human Computer Interaction (HCI)*

Human computer interaction melibatkan lima komponen yaitu pengguna atau manusia, interaksi, sistem komputer, aktivitas, dan lingkungan kerja. Kunci utama dari HCI adalah daya guna (*usability*). *Usability* adalah tingkat produk dapat digunakan yang ditetapkan oleh pengguna, untuk mencapai tujuan secara

efektif, efisien dan memuaskan dalam menggunakannya (Anzai dan Ogawa, 1995).

Pengertian *usability* dikuatkan juga oleh Nielsen dengan mengatakan bahwa *usability* merupakan suatu atribut kualitas yang menilai kemudahan penggunaan antar muka, yang memungkinkan pengguna untuk menyelesaikan tugasnya dengan jelas, transparan, lincah, dan *useful*. Hal-hal yang berkaitan dengan interaksi diantaranya adalah antar muka (*user interface*), bahasa, orientasi, pada alat dan perangkat, fleksibilitas, kompatibilitas sistem dengan sistem lain dan komunikasi (Norman dan Draper, 1986).

Salah satu bahasa terpenting dalam HCI adalah antar muka pengguna, yang merupakan bagian sistem yang dikendalikan oleh pengguna untuk mencapai dan melaksanakan fungsi-fungsi suatu sistem. Pengguna berhubungan dengan sistem melalui antar muka pengguna. Peran antar muka pengguna dalam daya guna suatu sistem sangatlah penting. Oleh karenanya bentuk dan pembangunan antar muka pengguna perlu dilihat sebagai salah satu proses utama dalam keseluruhan pembangunan suatu sistem (Sudarmawan, 2007).

2.2.9. Prinsip 8 Golden Rules of Interface Design

Dalam bukunya yang berjudul "*Designing The User Interface*", Ben Shneiderman (1986) mengemukakan bahwa *8 golden rules principle* merupakan sebuah aturan yang menetapkan *interface* agar dapat menyempurnakan implementasi dalam hal berinteraksi pada manusia. Beberapa poin-poin yang terdapat dalam *8 golden rules* diantaranya sebagai berikut:

1. *Strive for consistency*

Konsisten terhadap jenis, ukuran, dan spasi *font* merupakan hal yang terpenting untuk membuat sebuah aplikasi yang mempunyai nilai yang baik. Tampilan aplikasi tidak seharusnya rumit, mempunyai warna yang tidak terlalu banyak. *User* biasanya tidak ingin mempelajari hal baru yang membuat waktu *user* terbuang karena menggunakan aplikasi.

2. *Cater to universal usability*

Mengenali kebutuhan *user* yang beragam dan memudahkan dalam melakukan perubahan konten. Mencari perbedaan user pemula, user ahli, rentang usia, keterbatasan, kemampuan fisik, dan perbedaan teknologi yang merupakan panduan dalam merancang *interface*. Menambahkan fitur-fitur untuk *user* pemula seperti *shortcut* untuk *intermediate user* dan untuk *user* ahli membuat tampilan yang kompleks dapat meningkatkan kualitas dari desain *interface*.

3. *Offer informative feedback*

Aplikasi yang baik selalu memberikan timbal balik ketika terjadi sesuatu di dalam aplikasi itu sendiri. Sehingga memudahkan *user* untuk bertindak seperti yang sistem inginkan.

4. *Design dialogs to yield closure*

Aplikasi yang baik juga harus mempunyai suatu perbedaan tampilan yang bisa memberi tahu *user* bahwa kegiatan yang dilakukan sudah selesai. Seperti, ketika *user* sudah melakukan *login* maka akan muncul menu utama, menandakan bahwa proses *login* yang *user* lakukan sudah selesai.

5. *Prevent errors*

Aplikasi yang baik adalah dengan adanya *validasi* yang dapat mengurangi kesalahan yang akan terjadi pada *user*. Apabila user melakukan kesalahan, sistem harus mampu mendeteksi kesalahan yang dilakukan oleh user, sehingga mengurangi peluang *user* untuk melakukan kesalahan yang sama.

6. *Permit easy reversal of actions*

Di dalam sebuah aplikasi harus adanya fasilitas untuk *user* bisa mengedit atau menghapus tindakan yang sudah dilakukan sebelumnya. Sehingga mengurangi kekhawatiran user untuk melakukan kegiatan pada aplikasi.

7. *Support internal locus of control*

Aplikasi perlu memiliki fasilitas untuk *user* agar dapat dengan bebas bernavigasi dan mengubah informasi akun yang dimilikinya sesuai dengan yang dikehendaki.

8. *Reduce short term memory load*

Keterbatasan manusia dalam mengolah informasi dalam memori jangka pendek membutuhkan antarmuka (*interface*) yang sederhana dan mengurangi pergerakan *window*. Yang dimaksud pergerakan *window* seperti ketika *user* harus mengingat suatu informasi pada satu layar dan informasi tersebut digunakan pada layar lain.

2.2.10. Front End

Front end adalah segala sesuatu yang menghubungkan antara *user* dengan sistem *back end*. Biasanya merupakan sebuah *interface* dimana *user* akan berinteraksi dengan sistem.

Pekerjaan yang sering muncul sebagai seorang *front end developer* adalah desainer *user interface*. *Front end developer* tidak membuat program yang berjalan pada logic bisnis tetapi lebih banyak fokus ke antarmuka, desain grafis (*user interface designer*) dan bagaimana membuat desain yang nyaman digunakan oleh pengguna (*user experience designer*).

Bahasa pemrograman yang biasa digunakan dalam pengembangan *front end* adalah html, java script, dan css. (Goldbot, 2016)

2.2.11. NetBeans

NetBeans merupakan perangkat lunak (*software*) yang digunakan untuk mengembangkan aplikasi, di dalam perangkat lunak terdapat IDE (*Integrated Development Environment*). Sebuah IDE merupakan lingkup pemrograman yang di integrasikan ke dalam suatu aplikasi perangkat lunak yang menyediakan GUI (*Graphic User Interface*).

2.2.12. JavaServer Faces (JSF)

JavaServer Faces merupakan spesifikasi java untuk membangun antarmuka pengguna untuk aplikasi web. JSF merupakan Platform dari JavaPlatform, Enterprise Edition. JSF 2 menggunakan Facelets sebagai sistem *template defaultnya*. Teknologi tampilan lainnya juga dapat digunakan, seperti XUL.

Cara kerja JSF berdasarkan model perancangan *user interface* berbasis komponen, menggunakan berkas XML yang disebut *template view* atau view Facelets. Permintaan (*request*) diproses oleh FacesServlet, yang memuat *template*

view. Membangun *tree* komponen, memproses berbagai *event*, dan *me-render* respon (umumnya HTML) kepada klien. *State* dari komponen UI (dan beberapa objek lain) disimpan pada setiap akhir *request* (dinamakan *stateSaving*), dan dikembalikan pada saat pembuatan selanjutnya dari *view* tersebut.

2.2.13. Balsamiq

Dalam bukuyang berjudul *Balsamiq Quickstart Guide* karya Scott Faranello (2012), menyebutkan untuk memahami balsamiq adalah dengan memahami *wireframing*. *Wireframing* adalah membuat sketsa ide untuk produk atau web antarmuka dalam bentuk kerangka, menggambarkan pada tingkat tinggi apa yang akan dilakukannya, bagaimana nanti terlihat, dan bagaimana hal itu akan berfungsi. (Faranello, 2012)