

ANALISIS SISTEM DESKTOP BERBASIS CLIENT/SERVER DENGAN APLIKASI BERBASIS WEB STUDI KASUS PENGOLAHAN DATA KAPAS PADA PT.ABC

Zen Munawar

Program Studi Manajemen Informatika
Politeknik LP3I Bandung
E-mail: munawarzen@gmail.com

Abstrak : Sistem pengolahan data kapas di PT. ABC menggunakan aplikasi desktop berbasis client/server (arsitektur 2 tier). data penerimaan dan data pengeluaran sudah berjalan baik, namun informasi yang dibutuhkan oleh pengguna sistem lebih luas, tidak hanya pada tingkat jaringan lokal. sehingga dapat terjadi kelambatan informasi mengenai pengelolaan stok bahan baku yang dibutuhkan oleh kantor pusat, karena sistem hanya dapat menyediakan informasi pada tingkat jaringan lokal. Oleh karena itu diperlukan pengembangan arsitektur sistem dengan menggunakan arsitektur 3 tier. Aplikasi web merupakan aplikasi berbasis client/server dengan arsitektur 3 tier, yang dapat menangani kebutuhan informasi yang lebih luas sehingga kebutuhan akan informasi pengelolaan persediaan bahan baku kapas oleh kantor pusat secara real time mudah didapatkan danantisipasi kenaikan dan penurunan permintaan barang dari konsumen dapat diatasi dengan baik. Kelebihan dari penggunaan aplikasi web ini adalah pemeliharaan lebih efektif dan efisien, yaitu tidak memerlukan deployment atau pengembangan program pada setiap client, cukup menggunakan browser web sebagai user interface. Aplikasi web yang dikembangkan menggunakan PHP scripting language, MySQL database server dan Apache web server.

Kata Kunci : *desktop, client/server*, pengolahan data kapas, PHP, MySQL

1. Pendahuluan

Model konektivitas pada jaringan yang membedakan fungsi komputer sebagai terminal akses serta pusat pengolahan dan layanan disebut Client-server. Client merupakan terminal yang digunakan oleh pengguna untuk meminta layanan tertentu yang dibutuhkan. Terminal client dapat berupa PC, ponsel, komunikator, robot, televisi dan peralatan lain yang membutuhkan informasi. Server merupakan komputer khusus yang bertugas melayani aplikasi-aplikasi jaringan / pihak yang menyediakan layanan. Arsitektur ini menempatkan sebuah komputer sebagai server yang bertugas sebagai pusat pengolahan dan layanan bagi terminal-terminal lain (client) yang terhubung dalam sistem jaringan itu. Komputer yang difungsikan sebagai server hanya berperan untuk melayani permintaan layanan dari client. Hubungan yang terjadi antara server dan client adalah *one-to many*, yang berarti bahwa satu server melayani banyak client. Client selalu memulai transaksi dengan meminta layanan sedangkan server menanti permintaan layanan secara pasif. Model ini dapat menjawab problematika rendahnya kualitas antarmuka pada terminal-terminal akses dalam arsitektur Master-slave karena pada model arsitektur Client-server, komputer client bisa merupakan *intelligent terminal*, yaitu memiliki CPU yang dapat membantu proses dalam penyajian grafis yang tinggi.

Salah satu komponen utama keberhasilan dari pemasaran hasil produksi adalah terjaminnya persediaan produksi (produk). Keberhasilan produksi sangat dipengaruhi oleh manajemen pengelolaan stok bahan baku, dalam hal ini adalah kapas sebagai bahan baku utama dari proses produksi. Informasi ketersediaan stok harus dapat diketahui setiap saat dengan mudah. PT. ABC adalah perusahaan yang bergerak dibidang produksi benang dan kain dengan lokasi di Bandung sebagai pabrikasi, sedangkan untuk pemasaran hasil produksi ditangani oleh kantor pusat yang berlokasi di Jakarta. Pemasaran produksi dilakukan berdasarkan permintaan dan disesuaikan dengan keinginan konsumen.

Pengolahan data bahan baku kapas di perusahaan tersebut menggunakan aplikasi *desktop* berbasis *client/server* dengan arsitektur sistem model *2 tier* sehingga pengelolaan dan pengawasan serta pembuatan laporan stok telah berjalan cukup baik. Namun terdapat kelemahan karena informasi yang tersedia hanya dapat diakses pada tingkat jaringan lokal. Hal ini menyebabkan keterlambatan pengaksesan informasi tersebut oleh bagian pemasaran dan permintaan konsumen dan pabrikasi yang berada pada kota yang berbeda. Selain Aplikasi *desktop* ini memiliki beberapa kelemahan yaitu manajemen pengelolaan dan perawatan mesin-mesin *client* yang membutuhkan pengembangan serta proses konfigurasi yang memerlukan waktu dan tenaga, misalnya *front-end* untuk suatu database harus dipasang satu persatu pada setiap komputer *client* dan apabila suatu saat pada komputer *client* diperlukan perubahan atau pengembangan, prosesnya harus diulang pada setiap komputer *client*. Setelah dilakukan analisis sistem maka diusulkan untuk membangun sebuah aplikasi web berbasis *client/server* dengan arsitektur sistem *3 tier* yang dapat mengolah informasi sehingga menjadi media informasi yang dinamis serta dapat diakses oleh penggunaanya di berbagai tempat yang berbeda

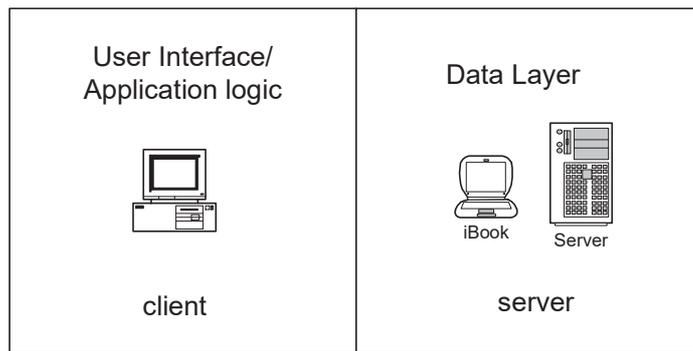
2. Tinjauan Pustaka

Arsitektur jaringan *client/server* merupakan pengembangan dari arsitektur *file server*. Arsitektur ini adalah model konektivitas pada jaringan yang mengenal adanya *server* dan *client*, dimana masing-masing memiliki fungsi yang berbeda satu sama lain. *Server* dapat berbagi pakai data, aplikasi dan *peripheral* seperti *harddisk*, *printer*, *modem* dan lain-lain. Oleh karena itu, tidak jarang juga tercipta sebutan *print server*, *communication server* dan lain sebagainya. Prinsip kerjanya sangat sederhana, dimana *server* akan menunggu permintaan dari *client*, memproses dan memberikan hasilnya kepada *client*. Sedangkan *client* akan mengirimkan permintaan ke *server*, menunggu proses dan melihat visualisasi hasil prosesnya. Sistem *client server* ini menggunakan protocol TCP/IP (*Transmission Control Protocol/Internet Protocol*). Unix dan Windows NT merupakan contoh yang baik dari sistem operasi jaringan *client server*.

2.1 Arsitektur Sistem (Model 2 Tier)

Tier pertama dalam aplikasi *desktop* berbasis *client/server* adalah *client tier* atau *presentation tier*, yang berjalan di *client*. *Tier* ini memiliki kode yang merepresentasikan data dan berinteraksi dengan user. Aplikasi *client* melakukan permintaan data dari database dan menampilkannya pada layar. Komputer *client* mampu untuk memanipulasi data sedangkan *server* tidak dilibatkan dalam pemrosesan data. *Tier* kedua adalah *database server*, atau DBMS. *Tier ini* memanipulasi objek yang sangat kompleks dan

database. DBMS tidak dapat memproses data sebelum melalui *client*. [Evangelos Petroustos : 2000 : 14].

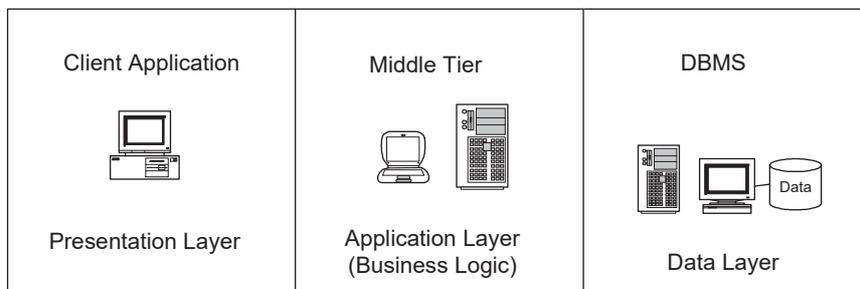


Gambar 1.

Arsitektur 2 tier

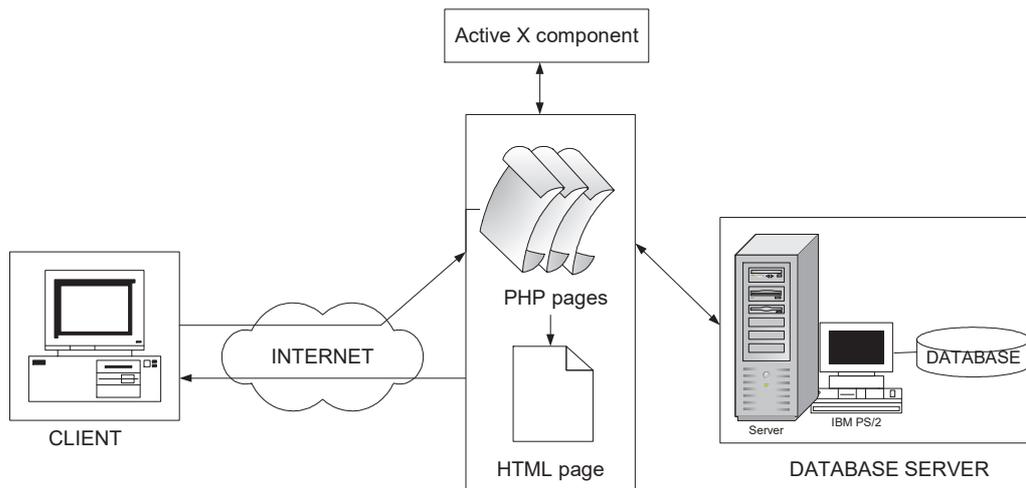
2.2 Arsitektur Sistem (Model 3 Tier)

Model 3 tier pada aplikasi web, dibagi menjadi beberapa komponen aplikasi dalam 3 kategori atau *layer*. Yang pertama adalah *presentation layer*, program ini berjalan pada *client* dan berinteraksi dengan *user*, yaitu menyediakan informasi bagi *user*. Kedua adalah *application layer* atau dikenal dengan *business layer*, *layer* ini berisi *logic application* serta mempermudah *client* mengakses ke database. Komponen dari *application* atau *business layer* sering juga disebut *business service*. *Layer* terakhir adalah *data layer*, *layer* ini adalah *database server*, yang melayani permintaan dari *client*. Permintaan biasanya berupa *query*. Database server harus memperbarui database dan pada waktu yang sama melindungi integritasnya. [Evangelos Petroustos : 2000 : 19].



Gambar 2. Arsitektur 3 tier

Ber macam macam *tier* dari *server* dapat berjalan pada mesin yang sama ataupun mesin yang berbeda. *Tier* dibedakan menurut fungsi yang mereka sediakan, bukan pada mesin yang berjalan. Berikut adalah gambar web sebagai sebuah jaringan *three-tier* menurut Evangelos Petroustos dalam bukunya “Database Programming With Visual Basic 6” [Evangelos Petroustos : 2000 : 651].



Gambar 3. Web sebagai lingkungan *three-tier*

2.3 Transmission Control Protocol /Internet Protocol (TCP/IP)

TCP/IP (*Transmission Control Protocol /Internet Protocol*) adalah nama keluarga *protocol* jaringan. *Protocol* adalah sekelompok aturan yang harus diikuti oleh perusahaan-perusahaan dan produk *software* agar produk mereka bisa kompatibel satu dengan yang lainnya. Suatu protokol menentukan bagaimana suatu *software* berkomunikasi dengan *software* lain juga menentukan bagaimana setiap bagian dari keseluruhan paket mengatur perjalanan informasinya. Internet pada awalnya telah menggunakan TCP/IP pada tingkat protokolnya yang memungkinkan sistem apapun yang terhubung ke dalamnya bisa berkomunikasi dengan sistem lain tanpa memperdulikan bagaimana sistem masing masing bekerja. TCP/IP sebenarnya merupakan dua protokol yang berbeda. TCP/IP dirancang untuk menjadi komponen perangkat lunak dari sebuah jaringan. Semua bagian didalam keluarga TCP/IP memiliki tugas tersendiri, misalnya mengirim *e-mail*, mentransfer *file*, menyediakan *log-in* jarak jauh (*remote login*) dan menangani informasi *routing* jaringan.

Protokol TCP bertanggung jawab memecah informasi ke dalam beberapa paket, sedangkan IP bertanggung jawab mengangkut (mentransfer) paket-paket tersebut sesuai tujuannya. Kemudian TCP bertugas menyatukan kembali paket-paket itu ke urutan yang benar. Kesalahan dalam aplikasi: Aplikasi pada *server* bisa memiliki kesalahan dan tidak berfungsi tepat.

Layanan dalam TCP/IP yang berbeda dikelompokan menurut fungsi-fungsinya. Protokol-protokol transport mengendalikan pergerakan data antara dua mesin, dan mencakup:

1. TCP (*transmission Control Protocol*), protokol ini bersifat *connection based*, artinya kedua mesin pengirim dan penerima tersambung dan berkomunikasi satu sama lain sepanjang waktu.
2. UDP (*User Datagram Protocol*), protokol ini bersifat *connectionless* (tanpa koneksi) artinya data dikirim tanpa kedua mesin penerima dan pengirim saling berhubungan

Sementara itu ada juga protokol-protokol *routing* untuk menangani pengalamatan (*addressing*) data dan menentukan jalur terbaik untuk mencapai tujuan. Protokol-protokol tersebut juga bertanggung jawab memecah informasi berukuran besar dan menyusunnya kembali kepada tujuan.

2.4 Hypertext Transfer Protocol

Server dan *web browser* berkomunikasi satu sama lain dengan protokol yang memang dibuat khusus untuk ini, yaitu HTTP. HTTP bertugas menangani permintaan-permintaan dari *browser* untuk mengambil dokumen dokumen web.

HTTP bekerja diatas TCP/IP yang menjamin data sampai ditujuan dengan urutan yang benar. Bila sesuatu kesalahan terjadi pada saat proses pengiriman, pihak pengirim akan mendapat pemberitahuan bahwa telah terjadi kesalahan. Karenanya *server* dan *client* tidak perlu menyediakan mekanisme untuk memeriksa kesalahan transmisi data, yang berarti mempermudah pekerjaan pemrograman. Namun demikian HTTP tidak memiliki apa yang disebut *session*, Setiap halaman web yang dikirim akan melibatkan satu proses penyambungan antara *client* dan *server*, baru kemudian datanya ditransfer. Setelah data selesai ditransfer, koneksi antara *server* dan *client* akan diputus. Sifatnya ini membuat HTTP sering disebut dengan istilah protokol *hit and run*.

Suatu halaman web seringkali berisi file gambar, atau beberapa file-file lain. HTTP memaksa server untuk menjalin hubungan baru setiap kali hendak mengirimkan satu buah file. Ini tidak menguntungkan dan tidak efisien mengingat proses hubung-putus ini menyebabkan beban bagi jaringan. Standar HTTP yaitu HTTP 1.1 di rancang untuk mengatasi masalah diatas, web diarahkan untuk mengarah ke penggunaan *persistent connection* (sambungan yang terjaga berkesinambungan) secara lebih efisien. Dalam HTTP 1.1 *server* tidak akan memutuskan hubungan dengan *client* pada akhir pentransferan dokumen. Hubungan tetap dibuka untuk melayani bila saja ada permintaan lagi dalam waktu yang singkat. Hubungan yang baru akan diputuskan apabila setelah melewati suatu batas waktu tertentu, *client* tidak mengirimkan permintaan lagi.

.2.5 Hypertext Markup Language (HTML)

HTML dikenal sebagai bahasa standar untuk membuat dokumen web. Perintah-perintah HTML diletakkan dalam file berekstensi *.html dan ditandai dengan menggunakan tanda berupa karakter “<” dan “>”. Tidak seperti bahasa pemrograman terstruktur prosedural seperti Pascal atau C, HTML tidak mengenal *jumping*, ataupun *looping*. Kode-kode HTML dibaca oleh browser dari atas ke bawah tanpa adanya lompatan-lompatan. Struktur sebuah dokumen HTML pada dasarnya dibagi menjadi dua bagian besar, yaitu *Header* dan *Body*. Masing-masing ditandai dengan pasangan *container head* <head> dan <body>. Bagian *head* berisikan judul dokumen dan informasi-informasi dasar lainnya, sedangkan bagian *body* adalah data dokumennya. HTML diatur oleh konsorsium W3C semua perubahan atas standar bahasa HTML harus disahkan terlebih dahulu oleh konsorsium ini. W3C kini merupakan badan resmi yang membuat standar web. W3C meletakkan gabungan spesifikasi dalam standar web, berikut adalah hasil dari W3C :

2.6 PHP

PHP merupakan bahasa berbentuk skrip yang ditempatkan pada sisi server dan diproses di server, hasilnya akan dikirimkan ke klien, tempat pemakai menggunakan browser. Kode program PHP menyatu dengan *tag-tag* HTML dalam satu file. Kode PHP diawali dengan tag “<?” atau “<?php” dan diakhiri dengan tag “?>”. File yang berisi tag HTML dan kode PHP ini diberi ekstensi (*.php) atau ekstensi lainnya yang ditetapkan pada Apache/Web server. Berdasarkan ekstensi ini, pada saat file diakses, *server* akan tahu bahwa file ini mengandung kode PHP. *Server* akan menterjemahkan kode ini dan menghasilkan output dalam bentuk tag HTML yang akan dikirim ke *browser* yang mengakses file tersebut.

Skrip PHP berkedudukan sebagai tag dalam bahasa HTML. Sebagaimana diketahui HTML (*Hyper Text Markup Language*) adalah bahasa standar untuk membuat halaman-halaman Web. Model kerja PHP diawali dengan permintaan

suatu halaman web oleh browser. Berdasarkan URL (*Uniform Resource Locator*) atau dikenal dengan sebutan alamat Internet, browser mendapatkan alamat dari web server, mengidentifikasi halaman yang dikehendaki, dan menyampaikan segala informasi yang dibutuhkan oleh web server. Selanjutnya web server akan mencarikan berkas yang diminta dan isinya segera dikirim ke mesin PHP dan mesin inilah yang memproses dan memberikan hasilnya (berupa kode HTML) ke web server. Selanjutnya web server menyampaikan ke klien.

2.7 XAMPP

XAMPP adalah perangkat lunak bebas, yang mendukung banyak sistem operasi, merupakan kompilasi dari beberapa program. Fungsinya adalah sebagai *server* yang berdiri sendiri (localhost), yang terdiri atas program Apache HTTP Server, MySQL *database* dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan Perl. Nama XAMPP merupakan singkatan dari X (empat sistem operasi apapun), APACHE, MySQL, PHP dan Perl [Anhar, 2010].

2.8 Web Aplikasi

Web merupakan terobosan baru sebagai teknologi sistem informasi yang menghubungkan data dari berbagai sumber dan layanan yang beragam macamnya di internet. Aplikasi web memiliki bermacam-macam *form* yang ditampilkan pada halaman-halaman HTML. Pada aplikasi *windows* misalnya, terdapat berbagai pilihan untuk *form* tersebut berkomunikasi antara satu dengan yang lainnya seperti menggunakan *public variabel*, dapat membaca langsung nilai dari setiap *control* di setiap *form*, juga dapat menyeting *properties-properties* lainnya. Situasinya berbeda dalam sebuah aplikasi web, setiap halaman adalah sebuah *entity* yang terpisah dan tidak dapat berinteraksi secara langsung dengan halaman yang lain pada *site* yang sama.

Aplikasi web adalah sebuah *site* yang bekerja seperti aplikasi *windows* biasa. Kita memasukkan informasi pada *form*, informasi tersebut di transmisikan pada *server* dan hasil dari prosesnya di kembalikan pada *client* sebagai dokumen HTML yang lain. Perbedaan yang utama antara aplikasi *windows* dengan aplikasi web yaitu *form* dapat di

buka pada *desktop* dan *user* dapat berpindah dari satu *form* ke *form* yang lain dengan mengklik *mouse*, namun berbeda pada aplikasi web yang hanya dapat menampilkan sebuah *form* dalam satu waktu. Apabila ingin membuka halaman yang lain harus memilih *link-link hypertext* yang ada pada halaman tersebut atau mengklik tombol *back* untuk menampilkan halaman yang telah di buka. Link-linknya sendiri bisa mengacu pada dokumen web, server FTP (*File Transfer Protocol*), *e-mail* ataupun layanan-layanan lain.

2.9 Entity Relationship Diagram (ERD)

Entity Relationship Diagram menurut Hanif Al Fatta mengemukakan bahwa ERD adalah gambar atau diagram yang menunjukkan informasi dibuat, disimpan dan digunakan dalam sistem bisnis [V.Yasin, 2012]

2.10 Konsep Persediaan Barang

Konsep persediaan barang : menurut Sofjan Assauri adalah sebagai berikut : “Suatu aktiva yang meliputi barang-barang milik perusahaan dengan maksud untuk dijual dalam suatu periode usaha yang normal, atau persediaan barang-barang yang masih dalam pengerjaan/proses produksi, ataupun persediaan bahan baku yang menunggu penggunaannya dalam suatu proses produksi”[Sofjan Assauri: 1978 : 176].

Pengertian Persediaan menurut Eddy Herjanto mengemukakan bahwa *inventory* adalah bahan atau barang yang disimpan yang akan digunakan untuk memenuhi tujuan tertentu, misalnya untuk digunakan dalam proses produksi atau perakitan, untuk dijual kembali, atau untuk suku cadang dari suatu peralatan atau mesin [E. Herjanto.2007: 237].

2.11 Sistem Persediaan

Sistem penentuan persediaan menurut Sofjan Assauri (1978 : 182) yaitu :

1. Sistem persediaan berkala (*periodic*), yaitu setiap akhir periode dilakukan perhitungan secara fisik dalam menentukan jumlah persediaan akhir.
2. Sistem perpetual atau juga disebut *Book Inventories* yaitu dalam hal ini di bina catatan administrasi persediaan. Setiap mutasi daripada persediaan sebagai akibat daripada pembelian ataupun penjualan dicatat atau terlihat dalam kartu administrasi persediaannya. Bila metode ini yang di pakai, maka perhitungan secara fisik hanya dilakukan paling tidak setahun sekali. Yang biasanya dilakukan untuk keperluan counter checking antara jumlah persediaan menurut fisik dengan kartu administrasi persediaannya.

2.12 Metode Penilaian Persediaan

Metode yang digunakan untuk menilai persediaan berdasarkan aliran biaya menurut Sofjan Assauri (1978:182-184) sebagai berikut :

1. **FIFO (*First In First Out*)**, cara ini didasarkan atas asumsi bahwa harga barang yang sudah terjual dinilai menurut harga pembelian barang yang terdahulu masuk. Dengan demikian persediaan akhir dinilai menurut harga pembelian barang yang terakhir masuk.
2. ***Weight Average Methode (WAM)***, cara ini didasarkan atas harga rata-rata dimana harga tersebut dipengaruhi oleh jumlah barang yang di peroleh pada

masing-masing harganya. Dengan demikian persediaan dinilai berdasarkan harga rata-rata.

3. **LIFO (*Last In First Out*)**, cara ini didasarkan atas asumsi, bahwa harga barang yang telah terjual menurut harga pembelian barang yang terakhir masuk. Sehingga persediaan yang masih ada/stock, dinilai berdasarkan harga pembelian barang yang terdahulu.

3. Analisis

Pada PT. ABC terdapat Divisi Pemintalan mempunyai 30.000 *spindle* mesin *Rieter*, *Toyoda*, *Howa* serta 3250 *rotor schlaforst Autocord*. Dengan hasil produksi 5500 metrik ton benang ring dan 20.000 metrik ton benang open end per tahun (Ring = 2500 bale/bulan, OE = 8000 Bale/bulan). Produksi akhirnya adalah benang ring Combed, Carded dan Blended serta benang OE 5 S sampai dengan 45 S. Divisi *Weaving* (Pertenunan) kapasitas produksi 30 juta yard kain denim dan *chambray* per tahun dengan memakai 138 buah mesin *weaving double width* "Sulzer" dan 59 buah "Picanol" serta "Ishikawa". Divisi ini menghasilkan kain *chambray* 4 onz sampai dengan 16 onz (Carded/Combed) dan Denim Open End Tradisional. Hasil produksi PT. ABC berupa kain denim, benang carded dan benang blended. Produksi benang terdiri dari dua macam, yaitu benang halus dengan nomor Ne₁ 30^s – 45^s dan benang kasar dengan nomor Ne₁ 5^s – 20^s.

Pemasaran hasil produksi dari pemintalan PT. ABC yang berupa benang, sebagian dipergunakan oleh bagian pertenunan sebesar 85 % untuk menghasilkan kain, dan 15 % benang langsung dipasarkan, sedangkan untuk jenis kain denim sebanyak 70 % di ekspor ke berbagai negara antara lain : Bangladesh, Jerman, Hungaria, Perancis, Thailand, Philipina, Hongkong, Srilangka, Vietnam, Timur tengah, Amerika, Australia, dan sisanya sebanyak 30% dipasarkan kedalam negeri. Untuk pemasaran hasil produksi ini dilakukan berdasarkan pemesanan yang diterima melalui bagian perencanaan produksi maupun kantor pusat di Jakarta.

Lingkup Kerja Departemen Gudang

- a. Sebelum bahan baku kapas tersebut dipindahkan dari *container* ke gudang terlebih dahulu dilakukan pengecekan oleh seorang petugas yang bertanggung jawab untuk melakukan pengecekan terhadap dokumen-dokumen yang menyertai pengiriman barang tersebut. Jika lengkap maka akan dilakukan pembongkaran atau sebaliknya hal tersebut dikomunikasikan ke bagian import.
- b. Jika pada *packing* terdapat kelainan atau kerusakan maka akan dilakukan pemotretan sebagai dokumentasi bahwa barang tersebut rusak pada saat pengiriman.
- c. Apabila kondisi gudang penuh dengan bahan baku maka hal tersebut akan dikomunikasikan dengan kepala gudang untuk menunda penyimpanan bahan baku kapas tersebut atau dilakukan penyewaan terhadap gudang luar.
- d. Aturan pengeluaran stock bahan baku kapas tersebut menggunakan prinsip FIFO yaitu stock yang lama terlebih dahulu dikeluarkan sedangkan yang baru datang, sementara disimpan terlebih dahulu untuk stock bahan baku berikutnya.

Bagian gudang mencatat data pembelian bahan baku kapas yang tertuang dalam kartu stock dan mengatur arus masuk dan keluarnya bahan baku yang keluar dari pemasok atau diminta oleh bagian produksi untuk pembuatan kain Denim, tetapi sebelum dibawa

kebagian produksi terlebih dahulu dilakukan pembersihan bahan baku kapas tersebut pada bagian *classer* untuk penyaringan, selanjutnya baru masuk ke bagian produk.

Pengelolaan Penyediaan Bahan Baku Yang Ditetapkan Oleh PT ABC

- a. Bagian produksi membuat perencanaan persediaan bahan baku berdasarkan persediaan bahan baku di gudang dalam bentuk laporan pembelian mengenai nama-nama barang dan bahan baku dengan jumlah dan kualitas tertentu.
- b. Bagian gudang melakukan fungsi sebagai berikut :
 - Fungsi pengeluaran, akan mengeluarkan kembali persediaan stock bahan baku dan barang yang dipasarkan dari pemasok bahan baku.
 - Fungsi penyimpanan, akan mempersiapkan tempat penyimpanan.
 - Fungsi penjualan akan melakukan transaksi penjualan jika ada permintaan dari customer.
- c. Penyediaan stok bahan baku berdasarkan klasifikasi bahan baku yang diinginkan perusahaan.

Untuk menghindari terjadinya kehilangan dan kecurangan-kecurangan, pihak manajer menggunakan pengendalian internal persediaan bahan baku dengan cara sebagai berikut :

- a. pengawasan terhadap personil yang keluar masuk gudang.
- b. setiap satu bulan sekali bagian akuntansi melakukan pengecekan pembelian, penjualan, pembayaran dan persediaan stock bahan baku digudang.
- c. Pemisahan antar fungsi pembayaran pembelian untuk bagian akuntansi dan penyimpanan, penjualan, pembayaran dari transaksi penjualan oleh bagian gudang.

Di bawah ini merupakan uraian tugas dari masing-masing bagian yang berkaitan dengan lingkup kerja pergudangan PT.ABC :

- a. Divisi Non Produksi
 - Mengusahakan pada pimpinan mengenai *budget* tahunan dan bulanan tentang kebutuhan logistik berdasarkan usulan dari semua bagian dan mengelola seluruh pengadaannya.
 - Memeriksa semua laporan persediaan dan mutasi barang serta kondisi seluruh gudang serta barang-barang yang tersimpan didalamnya. Bersama dengan *team* setiap tahun mengadakan *stock of name* seluruh bahan baku digudang dalam rangka perencanaan pembelian bahan baku untuk tahun mendatang.
 - Mengawasi dan melakukan penjualan barang yang sudah lama tersimpan di gudang.
- b. Kepala Departemen Logistik
 - Me review dan mengkoordinasikan pemesanan barang dan suku cadang yang di buat oleh bagian pembelian serta menyetujui rekapitulasi hutang dan piutang yang dibuat oleh kasir.
 - Mengkoordinasi dan menetapkan rekapitulasi perhitungan upah untuk pembayaran upah karyawan atau THR dan mengajukan ke kantor pusat.
 - Memeriksa dan menyetujui anggaran bulanan terhadap kenyataan setiap departemen dan menyetujui pengeluaran uang dari kasir.
- c. Kepala Departemen Gudang
 - Menyimpan semua catatan dan memeriksa laporan secara periodik untuk mengontrol jalannya fungsi gudang.

- Memeriksa dan mengesahkan pengiriman barang, pembayaran untuk memenuhi order penjualan.
- Mengatur penggunaan transport untuk pengiriman dan penyimpanan barang.
- Memeriksa laporan harian, pembelian dan pengiriman barang dan penerimaan barang.
- Memeriksa laporan harian pembelian dan penerimaan barang dan merencanakan anggaran tahunan untuk kebutuhan pergudangan serta tiap bulan melakukan evaluasi anggaran sesuai dengan kenyataan.

Gambaran Umum Sistem Persediaan Bahan Baku Kapas

Dalam pengelolaan persediaan bahan baku kapas ini yang bertanggung jawab untuk menangani penerimaan dan pengeluaran bahan baku kapas adalah bagian gudang. Berikut ini adalah prosedur-prosedur penerimaan dan pengeluaran kapas pada bagian gudang sebagai berikut :

a. Prosedur penerimaan kapas

- 1) Sebelum kapas diterima oleh bagian gudang kapas terlebih dahulu diperiksa kelengkapan dokumen import kapas oleh kepala departemen gudang.
- 2) Kemudian diperiksa dan ditentukan tempat penyimpanan oleh bagian gudang kapas.
- 3) Memeriksa kondisi fisik container/truck sesuai dengan dokumen dan kondisi muatan / kapas secara visual bersama surveyor.
- 4) Kemudian muatan di bongkar / dikeluarkan dan dicocokkan jumlahnya dengan dokumen import. Kemudian surat penerimaan (SP) diserahkan ke perusahaan angkutan.
- 5) Kemudian bagian gudang mengisi Laporan Penerimaan dan Pengeluaran Gudang (LPPG) dan Surat Pemberitahuan Pemasukan Barang Import (SPPBI).
- 6) Mendistribusikan Laporan Penerimaan dan Pengeluaran gudang (LPPG), Laporan Penerimaan Barang Import (LPBI) dan Surat Pemberitahuan Pemasukan Barang Import (SPPBI) kepersonel yang relevan kemudian dibuat fle /arsip.

b. Prosedur pengeluaran kapas

- 1) Mengidentifikasi perlunya barang oleh bagian produksi
- 2) Memeriksa persediaan kapas di gudang
- 3) Bagian produksi menyerahkan Material Requisition (MR) dan Receipt Form (RF) ke bagian gudang.
- 4) Bagian gudang menerima Material Requisition (MR) dan Receipt Form (RF) kemudian mereview.
- 5) Bagian gudang merealisasikan kebutuhan bahan baku kapas pada bagian produksi.
- 6) Memperbaharui persediaan bahan baku kapas kemudian membuat laporan stock.

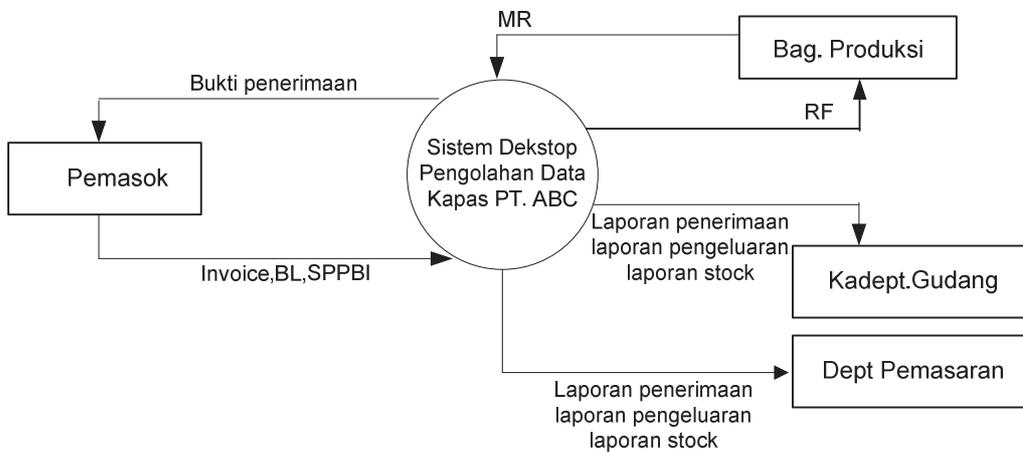
3.1 Even List

Berikut ini adalah *event list* yang menggambarkan daftar narasi stimuli (daftar kejadian) yang terjadi dalam lingkungan dan mempunyai hubungan dengan respon yang diberikan sistem :

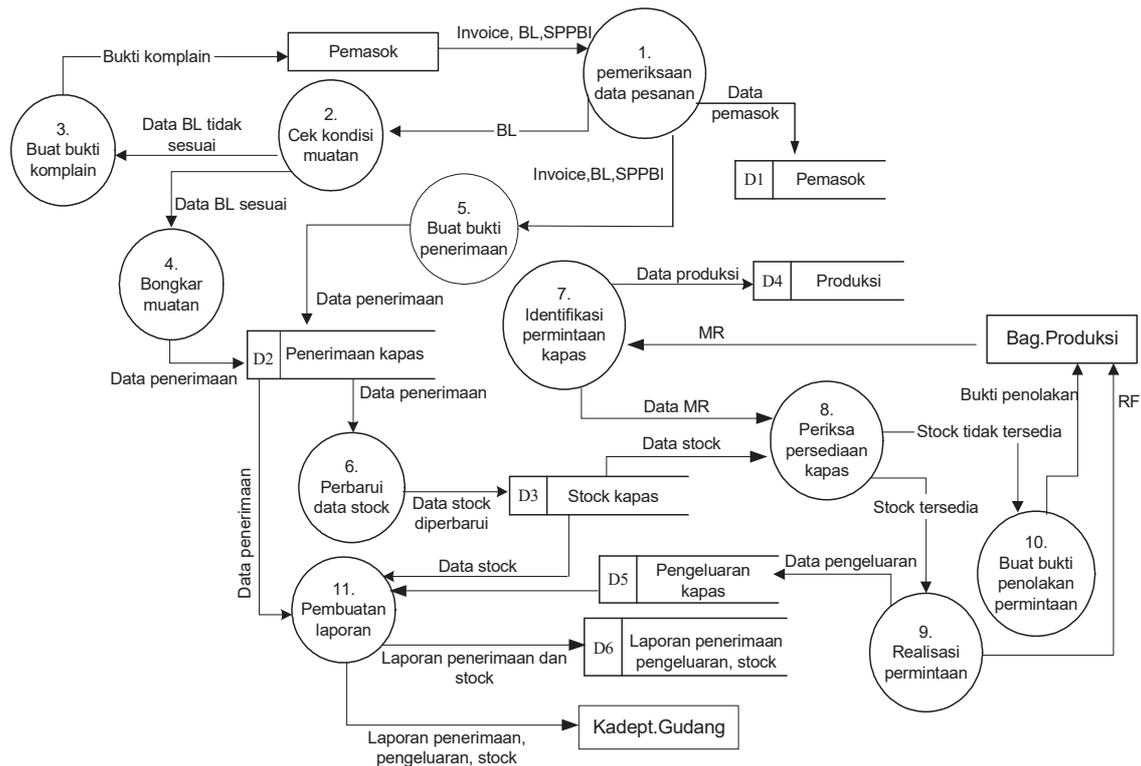
- a. Pemeriksaan terhadap dokumen-dokumen pemesanan (*invoice, BL, SPPBI*) dari pemasok.
- b. Melakukan bongkar muat jika dokumen-dokumen telah sesuai dengan kondisi muatan, sebaliknya dikomunikasikan ke bagian import.
- c. Melakukan pengecekan gudang untuk menyimpan bahan baku kapas. Bila kondisi gudang penuh dilakukan penangguhan penyimpanan atau penyewaan gudang luar.
- d. Membuat bukti penerimaan yaitu SPPBI dan LPPG.
- e. Input data penerimaan bahan baku kapas.
- f. Pembuatan laporan penerimaan bahan baku kapas
- g. Identifikasi permintaan kapas dari bag.produksi.
- h. Pengecekan stock kapas.
- i. Realisasi permintaan kapas bila tersedia stock digudang, sebaliknya dikomunikasikan dengan bagian produksi.
- j. Membuat bukti permintaan bahan baku kapas.

3.2 Data Flow Diagram (DFD)

DFD (*Data Flow Diagram*) menggambarkan sistem sebagai jaringan kerja antar fungsi yang berhubungan satu sama lain dengan aliran dan penyimpanan data. Terdapat dua bentuk DFD yaitu DFD fisik (*physical data flow diagram*) dan DFD logika (*logical data flow diagram*). DFD fisik lebih menekankan pada bagaimana proses dari sistem diterapkan, sedang DFD logika lebih menekankan proses-proses apa saja yang terdapat dalam sistem. Berikut DFD sistem berjalan :



Gambar 4. DFD Level 0 Sistem Desktop



Gambar 5. DFD Level 1 Sistem Desktop

4. Hasil Analisis

Sistem yang digunakan adalah aplikasi *desktop* berbasis *client/server* dengan arsitektur *2 tier* yang memiliki beberapa kelebihan dan kekurangan. Kelebihan dari aplikasi *desktop* berbasis *client server* (arsitektur *2 tier*) yaitu sebagai berikut :

1. Penggunaan *resource* lebih efisien.
2. Penyimpanan data terpusat.
3. lalu lintas data dalam jaringan menjadi lebih rendah.

Beberapa kelemahan dari sistem aplikasi *desktop* berbasis *client/server* (arsitektur *2 tier*) adalah sebagai berikut :

1. Pengaksesan oleh pengguna hanya terbatas pada tingkat jaringan lokal.
2. Manajemen dan perawatan mesin-mesin *client* yang membutuhkan pengembangan serta proses konfigurasi yang memakan waktu dan biaya.

Dari kelemahan-kelemahan tersebut maka diperlukan sebuah sistem yang dapat menangani masalah-masalah diatas. Dalam pengembangan sistem ini penulis mengusulkan pengembangan sistem dengan menerapkan aplikasi web berbasis *client/server* dengan arsitektur sistem *3 tier*. Dari sisi pengelolaan jaringan, sistem ini dapat mengatasi beberapa kelemahan dari aplikasi *desktop* berbasis *client/server* dengan arsitektur *2 tier* yaitu sebagai berikut :

1. Pendistribusian berlangsung secara sendirinya, karena setiap salinan dokumen (sebagai satu komponen aplikasi) di *download* ke mesin *client*, ketika mesin *client*

membutuhkan dan meminta *update*. Sehingga tidak perlu *deployment* perangkat lunak *client* untuk setiap komputer yang ada.

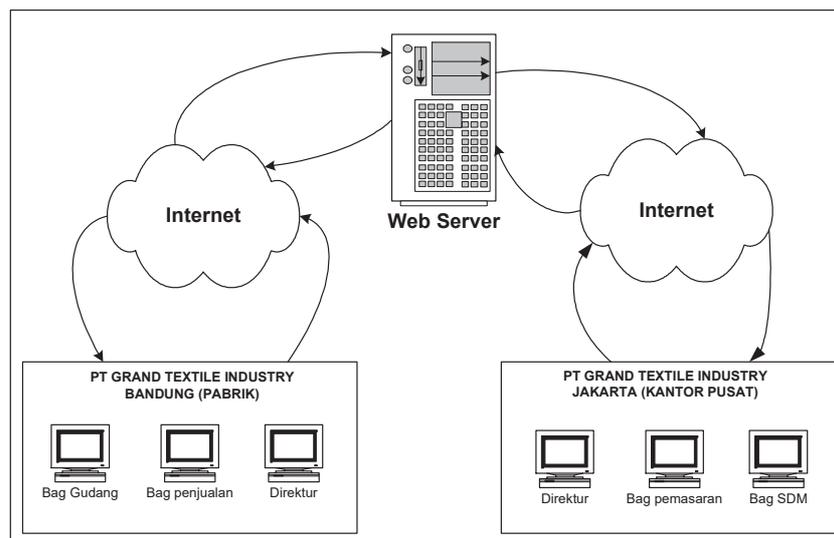
2. Perubahan-perubahan aplikasi dapat dikerjakan secara terpusat dan bisa langsung diterapkan tanpa perlu menyesuaikan semua *client*.
3. Fleksibilitas aplikasi web lebih terjamin karena tidak perlu lagi mengembangkan program-program *client* yang berbeda untuk setiap macam *platform*.

Dari sisi pengaksesan data, aplikasi web dapat diakses oleh lebih dari satu *user* yang berada pada tempat yang berbeda sehingga informasi yang dibutuhkan dapat diakses secara global. Dengan tingkat pengaksesan yang semakin luas, maka diperlukan juga keamanan sistem untuk menjaga penyalahgunaan (kewenangan) hak akses dari setiap bagian atau departemen dari perusahaan tersebut. Sistem yang dikembangkan ini dapat membatasi hak akses dari setiap departemen dengan memberikan hak-hak akses setiap *user* sesuai dengan kebijakan dari perusahaan untuk masuk ke sistem dengan menggunakan *password login*.

Sistem informasi pengelolaan persediaan bahan baku kapas ini memerlukan manajemen pengolahan data yang baik untuk memenuhi kebutuhan informasi oleh pengguna sistem. Teknologi web saat ini telah memiliki kemampuan untuk mengakses sistem database. *Web server* dapat dihubungkan dengan RDBMS (*Relational Database Management System*) seperti Oracle, SQL Server atau MySQL. Untuk keperluan pengembangan aplikasi web ini diperlukan sebuah bahasa *server side scripting* yaitu dalam pengembangan aplikasi ini menggunakan PHP (*Preprocessor Hypertext pages*), MySQL sebagai sistem manajemen database dan Apache sebagai *web server*.

5. Perancangan

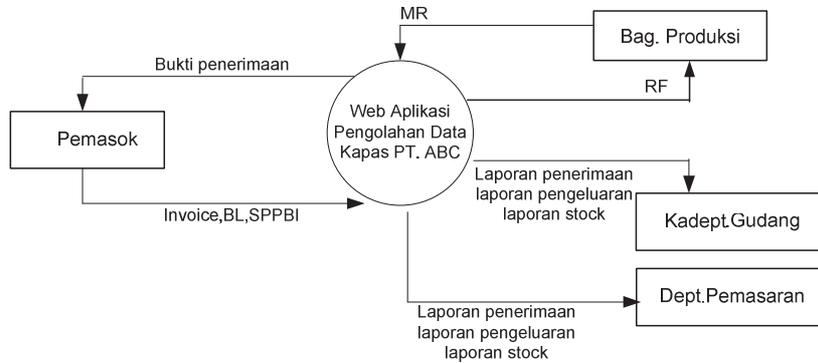
Berikut ini adalah arsitektur sistem aplikasi pengelolaan persediaan bahan baku kapas PT. ABC.



Gambar 6. Arsitektur Web Aplikasi PT. ABC

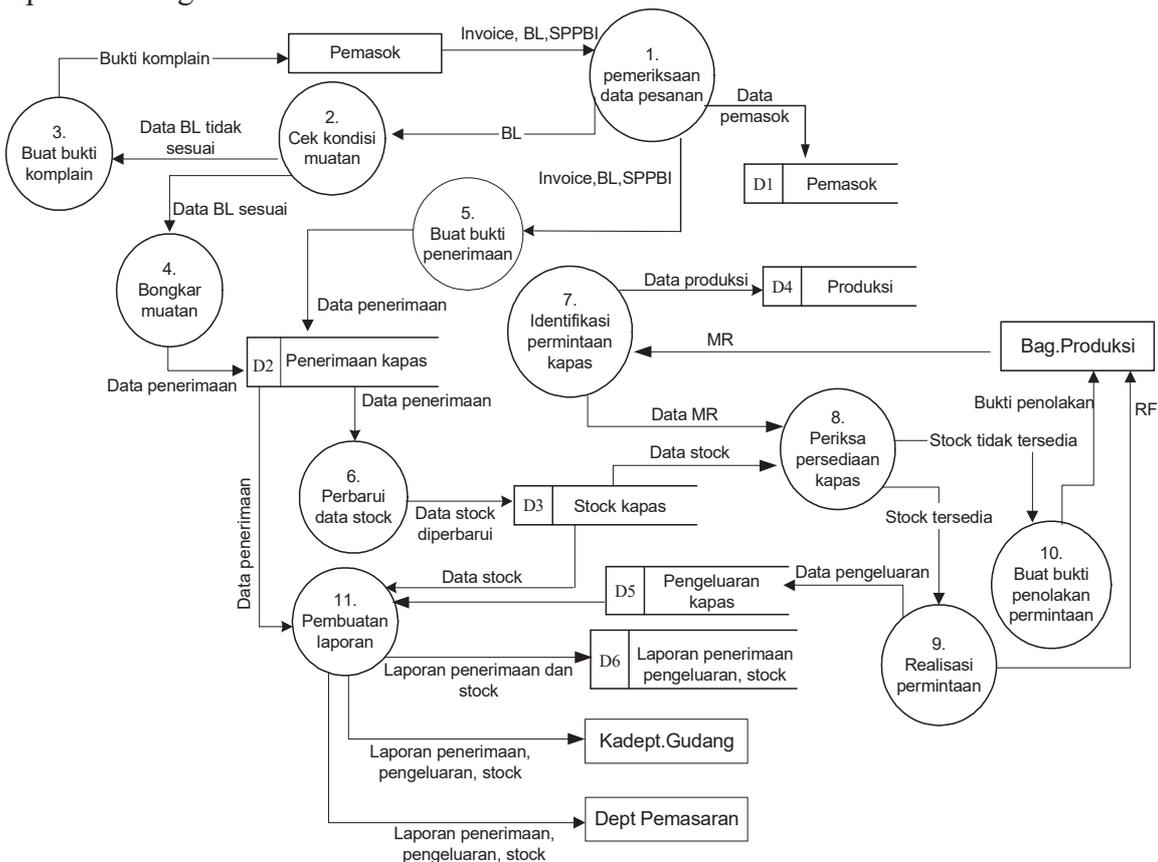
5.1 Data Flow Diagram (DFD)

Untuk menggambarkan sistem sebagai jaringan kerja antar fungsi yang berhubungan satu sama lain dengan aliran dan penyimpanan data, berikut *Data Flow Diagram* level 0



Gambar 7. DFD Level 0 Web Aplikasi

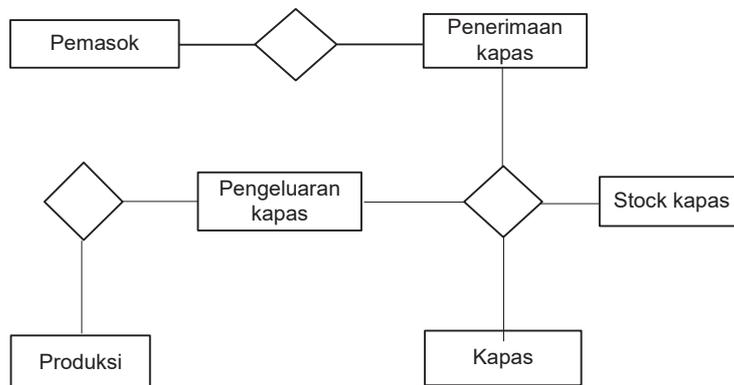
Selanjutnya disampaikan dalam bentuk gambar Data Flow Diagram Logika Level 1 Web Aplikasi sebagai berikut :



Gambar 8. DFD Logika Level 1 Web Aplikasi

5.2 Entity Relationship Diagram (ERD)

Adapun Entity Relationship Diagram nya adalah sebagai berikut :



Gambar 9. ERD Web Aplikasi

5.3 Kodifikasi Pemasok dan Produksi

Kode digunakan untuk mengklasifikasikan data, memasukan data kedalam komputer dan untuk mengambil bermacam-macam informasi yang berhubungan dengannya. Kode dapat dibentuk dari kumpulan angka, huruf dan karakter-karakter khusus. Tipe dari kode yang dapat digunakan di dalam sistem informasi diantaranya kode mnemonik (*mnemonic code*), kode urut (*sequential code*), kode blok (*block code*), kode group (*group code*) dan kode desimal (*decimal code*).

Kode Pemasok digunakan tipe kode mnemonik untuk pengkodean tabel pemasok pada *field* kode_pemasok. Kode mnemonik dibuat dengan dasar singkatan atau mengambil sebagian karakter dari *item* yang akan mewakili dengan kode ini. Berikut contoh penggunaan dari kode_pemasok :

- AUS → Australia
- IND → India
- PKT → Pakistan

Kebaikan dari kode ini adalah mudah diingat dan kelemahannya adalah kode dapat menjadi terlalu panjang.

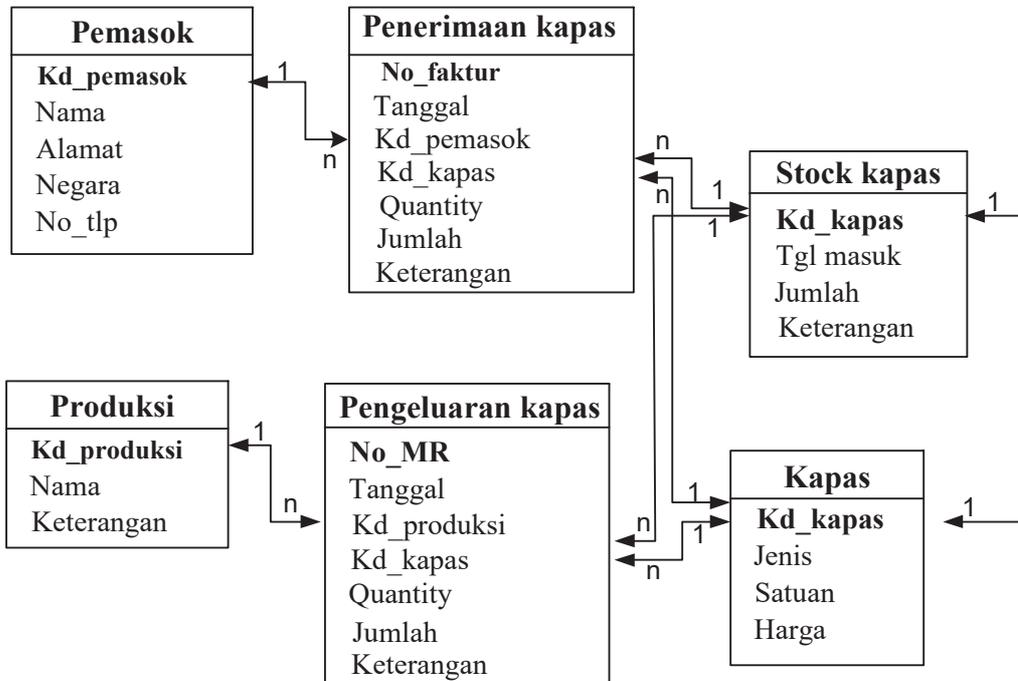
Untuk kode produksi digunakan kode group. Kode group merupakan kode yang berdasarkan *field-field* dan tiap-tiap field kode mempunyai arti. Contoh penggunaan kode produksi adalah sebagai berikut :

- SP-01 → produksi *spinning* (pemintalan) dengan no urut unit produksi 01.
- WV-01 → produksi *Weaving* dengan no urut unit produksi 01.

Dua digit pertama menunjukkan nama dari bagian produksi, dan dua digit terakhir merupakan no urut dari unit produksi. Kelebihan dari kode ini adalah nilai kode mempunyai arti, mudah diperluas, dapat ditambah atau dibuang sebagian serta dapat menunjukkan jenjang dari data.

5.4 Relasi antar File

Keterhubungan antar file dilakukan untuk mengorganisasikan file dalam rangka menghindari data item yang berulang-ulang (redudansi), karena file yang berisikan group data item yang berulang-ulang akan menyebabkan penyimpangan data yang sama berulang ulang di belakang file, sehingga menyebabkan pemborosan ruang penyimpanan, inkonsistensi data juga biaya pengaksesan menjadi tinggi. Dalam keterhubungan antar file ini di tentukan suatu field yang akan dijadikan suatu field kunci (*key field*).



Gambar 10. Relasi antar File

5.5 Kamus Data

Kamus data memiliki fungsi yang sama dalam pemodelan sistem yaitu membantu pelaku sistem untuk mengerti aplikasi secara detail, dan mengorganisasi semua elemen data yang digunakan dalam sistem secara presisi sehingga pemakai dan penganalisa sistem punya dasar pengertian yang sama tentang masukan, keluaran, penyimpanan dan proses.

Tabel 1. Kamus data

Tabel	Atribut	Tipe data	Panjang	1. Keterangan
Pemasok	Kd_pemasok	VarChar	3	Not null (primarykey)
	Nama	VarChar	50	Not Null
	Alamat	VarChar	50	Not Null
	Negara asal	VarChar	20	Null
Stock	No. Tlp	VarChar	20	Null
	Kd_kapas	VarChar	3	Not null (primarykey)
	Tgl_masuk	Date	Default	Format : mm:dd:yy

	Jumlah	int		Jumlah stock
	Keterangan	VarChar	100	Null
Penerimaan kapas	No. Faktur	VarChar	10	Not null (primary key)
	Tanggal	Date		Format: mmddyy
	Kd_pemasok	VarChar	10	NotNull (foreignkey)
	Kd_kapas	VarChar	100	Not Null
	Quantity	int	11	Not null
	Jumlah	Double		Not null
	Keterangan	Varchar	100	null
Produksi	Kd_produksi	VarChar	10	Not null (primary key)
	Nama	VarChar	50	Not null
	Keterangan	VarChar	100	null
Pengeluaran_kapas kapas	No_mr	VarChar	10	Not null (primarykey)
	Tanggal	Date		Format : mmddyy
	Kd_produksi	Varchar	10	Not null (foreignkey)
	Kd_kapas	VarChar	100	Not null
	Quantity	Int	11	Not null
	Jumlah	Double		Not null
	Keterangan	Varchar	100	null
Kapas	Kd_kapas	int	3	Not null (primary key)
	Jenis	VarChar	10	Not null
	Satuan	VarChar	10	Not null
	Harga	Double		Not null

5.6 Spesifikasi Proses

Untuk mendeskripsikan proses yang terjadi dalam level paling dasar dalam DFD, penulis menggunakan *Process Specification* yang berfungsi mendeskripsikan apa yang dilakukan ketika masukan ditransformasi menjadi keluaran. Model inilah yang menjelaskan pola kerja dari setiap proses. Berikut adalah *Process Specification* dari sistem.

a) Proses Pencatatan Penerimaan Kapas

Jenis proses : Entri.

Fungsi : untuk mencatat data penerimaan kapas.

Tempat penyimpanan : data store penerimaan kapas.

Perintah:

1. Simpan : menyimpan data penerimaan kapas.
2. Batal : membatalkan data yang telah dimasukan.
3. Edit : memperbaiki, menambah dan menghapus data.
4. View data : menampilkan data yang telah dimasukan.
5. Cari : pencarian data dengan menliskan No.Faktur

Validasi : sesuai dengan kamus data pada tabel kamus data diatas.

Gambar 11. Form Penerimaan Kapas

b) Proses Pencatatan Pengeluaran Kapas

Jenis proses : Entri.

Fungsi : untuk mencatat data pengeluaran kapas.

Tempat penyimpanan : data store pengeluaran kapas.

Perintah:

1. Simpan : menyimpan data pengeluaran kapas.
2. Batal : membatalkan data yang telah dimasukkan.
3. Edit : memperbaiki, menambah dan menghapus data.
4. View data : menampilkan data yang telah dimasukkan.
5. Cari : pencarian data dengan menuliskan No.Faktur

Validasi : sesuai dengan kamus data pada tabel kamus data diatas.

Gambar 12. Form Pengeluaran Kapas

c) Proses input data pemasok

Jenis proses : entri

Tempat penyimpanan : data store pemasok

Perintah :

1. Simpan : menyimpan data pemasok.
2. Batal : membatalkan data yang telah dimasukan.
3. Edit : mengkoreksi, menambah dan menghapus data
4. Cari : pencarian data dengan menuliskan kode pemasok
5. View data : menampilkan data pemasok.

Validasi : sesuai dengan kamus data pada tabel kamus data diatas.

Gambar 13. Form Data Pemasok

d) Proses input data produksi

Jenis proses : Entri

Tempat penyimpanan : data store produksi

Perintah :

1. Simpan : menyimpan data produksi.
2. Batal : membatalkan data yang telah dimasukan.
3. Edit : mengkoreksi, menambah dan menghapus data.
4. Cari : pencarian data dengan menuliskan kode produksi.
5. View data : menampilkan data produksi.

Validasi : sesuai dengan kamus data pada tabel kamus data diatas.

Gambar 14. Form Data Produksi

e) Proses input data kapas

Jenis proses :entri

Tempat penyimpanan :data store kapas

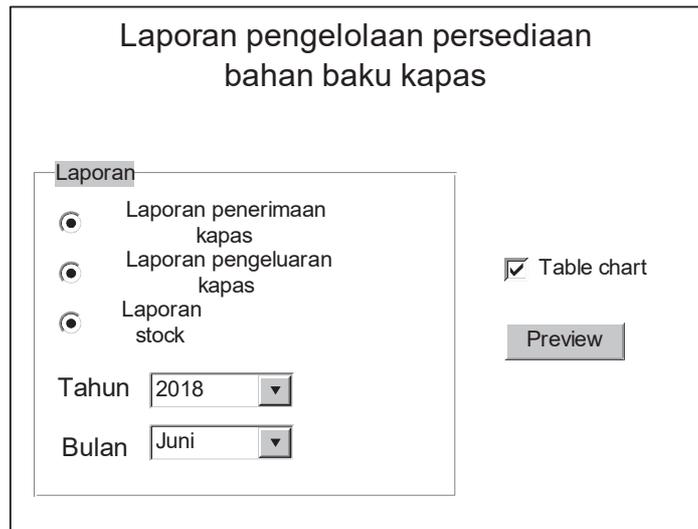
Perintah :

1. Simpan : menyimpan data kapas.
2. Batal : membatalkan data yang telah dimasukan.
3. Edit : mengkoreksi, menambah dan menghapus data.
4. Cari : pencarian data kapas dengan memasukan data
5. View data : menampilkan data kapas.

Validasi : sesuai dengan kamus data pada tabel kamus data diatas.

Gambar 15. Form Data Kapas

a) Proses Pembuatan Laporan



Gambar 16. Menu Laporan

- a. Jenis proses : report
- Fungsi : pembuatan laporan penerimaan kapas.
- Perintah :
- Preview : menampilkan laporan kelayar.
- Sumber data : Penerimaan kapas, pemasok, kapas

Sumber data : Penerimaan kapas, pemasok, kapas.

LAPORAN PENERIMAAN KAPAS
Per 01-01-2018 s/d 31-07-2018

No Faktur	Tanggal	Kode Pemasok	Kode Kapas	Quantity	Jumlah	Keterangan

Kepala Gudang Kadept. Gudang Kepala Bagian

Gambar 17. Tampilan Laporan Penerimaan Kapas

- b. Jenis proses : report
- Fungsi : pembuatan laporan pengeluaran kapas.
- Perintah:

Sumber data : 1. Preview : menampilkan laporan kelayar.
: Pengeluaran kapas, Produksi, Kapas.

LAPORAN PENGELUARAN KAPAS
Per 01-07-2018 s/d 31-08-2018

No MR	Tanggal	Kode Produksi	Kode Kapas	Quantity	Jumlah	Keterangan

Kepala Gudang Kadept. Gudang Kepala Bagian
.....

Gambar 18. Tampilan Laporan Pengeluaran Kapas

c. Jenis Proses : report
Fungsi : pembuatan laporan stok kapas.
Perintah :
1. Preview : menampilkan laporan kelayar.
Sumber data : kapas, stock

LAPORAN STOK KAPAS
5 AGUSTUS 2018

Kode Kapas	Jenis	Harga	Jumlah	Tanggal Masuk

Kepala Gudang Kadept. Gudang Kepala Bagian
.....

Gambar 19. Tampilan Laporan Stok Kapas

6. Kesimpulan

Kebutuhan informasi mengenai data *stock* bahan baku kapas, data penerimaan kapas dan data pengeluaran kapas oleh pengguna sistem yang lebih dari satu user dengan lokasi yang berbeda dapat diatasi dengan penggunaan aplikasi web berbasis *client/server* dengan arsitektur sistem model 3 *tier*, sehingga aksesibilitas informasi yang disediakan sistem ini menjadi lebih baik.

Pengembangan serta pendistribusian program di setiap *client* lebih mudah dilakukan karena tidak diperlukannya pengembangan dan pendistribusian program di setiap *client* sehingga menghemat waktu dan biaya serta fleksibilitas sistem dengan menggunakan aplikasi web berbasis *client/server* dengan arsitektur sistem model 3 *tier* dapat terjaga karena dapat digunakan oleh setiap spesifikasi sistem yang berbeda.

Daftar Pustaka

- [1] Assauri S, (1978). *Management Produksi*, Lembaga Penerbit Fakultas Ekonomi Universitas Indonesia
- [2] Anhar. (2010), *Panduan Menguasai PHP & MySQL Secara Otodidak*. Jakarta:Mediakita.
- [3] E. Herjanto. (2007), *Manajemen Operasi*. Edisi ketiga. Jakarta: Grasindo.
- [4] Jogiyanto (2005). *Analisis dan Desain Sistem Informasi: Pendekatan Terstruktur, Teori dan Praktik Aplikasi Bisnis*. Yogyakarta. Andi
- [5] Komputer, Wahana (2012). *Membangun Web Interaktif dengan Adobe Dreamweaver CS5.5, PHP dan MySQL*. Semarang. Andi
- [6] Madcoms (2012). *JavaScript Untuk Membangun Website Profesional*. Yogyakarta: ANDI.
- [7] Petroustos E, 2000 *Mastering Database Programming With Visual Basic 6*. Sybex
- [8] Pressman S Roger, 2001 *Software Engineering : A Practitioner's Approach*. McGraw-Hill.
- [9] V.Yasin, (2012), *Rekayasa Perangkat Lunak Berorientasi Objek Pemodelan 3 dan Perancangan (Modelling, Architecture and Design)*, Jakarta: Mitra Wacana Media, 2012. hlm. 267-279.