

## BAB 2

### KAJIAN PUSTAKA

#### 2.1 Retensi

Menurut (Anggraeni, 2016), Retensi adalah tindakan penahanan dimana ini merupakan kemampuan untuk melanjutkan kerja individu yang berkualitas, yaitu perawat dan penyedia layanan kesehatan lainnya / perusahaan asosiasi yang dinyatakan akan meninggalkan organisasi. Dampak dari tindakan ini adalah untuk menjaga stabilitas dan meningkatkan kualitas layanan serta mengurangi biaya bagi organisasi. Strategi retensi adalah program dan manfaat yang dirancang untuk meningkatkan perekrutan dan retensi karyawan. Untuk meningkatkan retensi perawat di rumah sakit diperlukan beberapa hal antara lain: seleksi calon perawat secara profesional, disepakati mengenai imbalan jasa yang akan diterima, kelanjutan pendidikan yang akan diperoleh, mengikutsertakan sertifikasi kekhususan, peningkatan komunikasi dengan perawat, dan pemberian waktu yang fleksibel bagi setiap perawat. Untuk mendukung retensi perawat di rumah sakit George Washington diperlukan penambahan insentif, perubahan jadwal kerja dari yang waktunya panjang menjadi lebih pendek, peningkatan promosi/karir dan peningkatan infra struktur berupa kultur organisasi yang memberikan kepuasan dan penghargaan atas peningkatan kerja perawat. Hal-hal yang dipertanyakan tersebut diatas adalah hal yang berkaitan dengan sistem retensi, penyusutan, dan pemusnahan berkas rekam medis. Retensi berarti menyimpan. Jadi sistem retensi adalah sistem yang mengatur jangka waktu penyimpanan berkas rekam medis (bukan sistem yang mengatur tata cara pemusnahan rekam medis). Peraturan Menteri Kesehatan (Permenkes) 269/Menkes/Per/III/2008 tentang Retensi dalam bab IV pasal 8 mengatur bahwa :

1. Rekam medis pasien rawat inap di rumah sakit wajib disimpan sekurang-kurangnya untuk jangka waktu 5 (lima) tahun dihitung dari tanggal terakhir pasien berobat atau dipulangkan;
2. Setelah batas waktu 5 (lima) tahun sebagaimana dimaksud pada ayat (1) dilampaui, rekam medis dapat dimusnahkan, kecuali ringkasan pulang dan persetujuan tindakan medik.
3. Ringkasan pulang dan persetujuan tindakan medik sebagaimana dimaksud pada ayat (2) hanya disimpan untuk jangka waktu 10 (sepuluh) tahun dihitung dari tanggal dibuatnya ringkasan tersebut.

## **2.1 Pengertian Rumah Sakit**

Rumah sakit adalah institusi pelayanan kesehatan yang menyelenggarakan pelayanan kesehatan prorangan secara parnipurna yang menyediakan pelayanan rawat inap, rawat jalan dan gawat darurat yang merupakan institusi kesehatan professional yang pelayanannya diselenggarakan oleh dokter, perawat, dan tenaga ahli lainnya. Di dalam Rumah Sakit terdapat banyak aktivitas dan kegiatan yang berlangsung secara berkaitan (Junaidi, 2019). Kegiatan-kegiatan tersebut menjadi bagian dari tugas serta fungsi Rumah Sakit, yaitu:

1. Memberi pelayanan medis
2. Memberi pelayanan penunjang medis
3. Memberi pelayanan kedokteran kehakiman
4. Memberi pelayanan medis khusus
5. Memberi pelayanan rujukan kesehatan
6. Memberi pelayanan kedokteran gigi
7. Memberi pelayanan sosial
8. Memberi penyuluhan kesehatan
9. Memberi pelayanan rawat jalan, rawat inap, rawat darurat dan rawat intensif.
10. Memberi pendidikan medis secara umum dan khusus
11. Memberi fasilitas untuk penelitian dan pengembangan ilmu kesehatan dan
12. Membantu kegiatan penyelidikan epidemiologi

## **2.2 Jenis-jenis rumah sakit**

Menurut (Anggraeni, 2016), Jenis-jenis Rumah Sakit di Indonesia secara umum ada lima, yaitu Rumah Sakit Umum, Rumah Sakit Khusus atau Spesialis, Rumah Sakit Pendidikan dan Penelitian, Rumah Sakit Lembaga atau Perusahaan, dan Klinik. Berikut penjelasan dari lima jenis Rumah Sakit tersebut:

1. Rumah Sakit Umum  
Rumah Sakit Umum, biasanya Rumah Sakit Umum melayani segala jenis penyakit umum, memiliki institusi perawatan darurat yang siaga 24 jam (Ruang gawat darurat). Untuk mengatasi bahaya dalam waktu secepat-cepatnya dan memberikan pertolongan pertama. Di dalamnya

juga terdapat layanan rawat inap dan perawatan intensif, fasilitas bedah, ruang bersalin, laboratorium, dan sarana-prasarana lain

2. Rumah Sakit Khusus atau Spesialis  
Rumah Sakit Khusus atau Spesialis dari namanya sudah tergambar bahwa Rumah Sakit Khusus atau Rumah Sakit Spesialis hanya melakukan perawatan kesehatan untuk bidang-bidang tertentu, misalnya, Rumah Sakit untuk trauma (trauma center), Rumah Sakit untuk Ibu dan Anak, Rumah Sakit Manula, Rumah Sakit Kanker, Rumah Sakit Jantung, Rumah Sakit Gigi dan Mulut, Rumah Sakit Mata, Rumah Sakit Jiwa,
3. Rumah Sakit Bersalin dan lain-lain  
Rumah Sakit Pendidikan dan Penelitian, Rumah Sakit ini berupa Rumah Sakit Umum yang terkait dengan kegiatan pendidikan dan penelitian di Fakultas Kedokteran pada suatu Universitas atau Lembaga Pendidikan Tinggi
4. Rumah Sakit Lembaga atau Perusahaan  
Rumah sakit ini adalah Rumah Sakit yang didirikan oleh suatu lembaga atau perusahaan untuk melayani pasien-pasien yang merupakan anggota lembaga tersebut
5. Klinik  
Merupakan tempat pelayanan kesehatan yang hampir sama dengan Rumah Sakit, tetapi fasilitas medisnya lebih

### **2.3 Rekam Medis**

Rekam medis adalah berkas yang berisikan catatan dan dokumen tentang identitas pasien, pemeriksaan, tindakan dan pelayanan lain yang telah diberikan kepada pasien. Pelayanan rekam medis merupakan salah satu pelayanan penunjang medis di rumah sakit yang menjadi dasar penilaian mutu pelayanan medik rumah sakit. Rekam medis pasien berisi informasi tentang catatan dan dokumen tentang identitas pasien, pemeriksaan, pengobatan, tindakan dan pelayanan lain yang telah diberikan kepada pasien (Mavkren, 2013).

Tujuan dari rekam medis adalah untuk menunjang tercapainya tertib administrasi dalam rangka upaya peningkatan pelayanan kesehatan di Rumah Sakit. Berdasarkan keputusan direktur jendral pelayanan medik nomor 78/ pelayanan Medik/ RS UMDIK/ YMU/ 1991 tentang petunjuk pelaksanaan penyelenggaraan

Rekam Medis di Rumah sakit, maka dengan ini rekam medis dapat dipakai sebagai berikut :

1. Sumber informasi medis pasien yang berobat di rumah sakit yang berguna untuk keperluan pengobatan dan pemeliharaan kesehatan pasien
2. Alat komunikasi antara dokter dengan dokter lain, antara dokter dengan paramedic lain dalam usaha memberikan pelayanan pengobatan dan perawatan.
3. Bukti tertulis tentang pelayanan yang telah diberikan oleh rumah sakit
4. Alat untuk analisa dan evaluasi terhadap kualitas pelayanan yang diberikan oleh rumah sakit.
5. Alat untuk melindungi kepentingan hokum bagi semua pasien, dokter, tenaga kesehatan lainnya serta rumah sakit itu sendiri.
6. Alat untuk penelitian dan pendidikan
7. Alat untuk perencanaan dan sumber daya
8. Alat unuk kepentingan lain yang ada kaitannya dengan rekam medis.

Kegunaan rekam medis:

1. Aspek Medis  
Berkas rekam medis terletak pada dokter yang merawat, tahap memperdulikan ada tidaknya bantuan yang diberikan kepadanya dalam melengkapi rekam medis oleh staf lain dirumah sakit.
2. Aspek Hukum  
Dokumen rekam medis secara umum untuk melindungi kepentingan hokum bagi pasien, dokter tenaga kesehatan lainnya di rumah sakit.
3. Aspek keuangan  
Tindakan-tindakan yang terekam dalam DRM dapat digunakan sebagai bukti untuk transaksi keuangan
4. Aspek Penelitian  
Rekam medis secara umum bisa digunakan sebagai bahan analisa penelitian, dan bahan evaluasi terhadap kualitas pelayanan yang diberikan kepada pasien.
5. Aspek Pendidikan  
Berkas rekam medis mempunyai nilai pendidikan, karena isinya menyangkut data atau informasi tentang perkembangan kronologis dan kegiatan pelayanan medis yang diberikan kepada pasien, informasi tersebut dapat digunakan sebagai bahan atau referensi pengajaran di bidang profesi.

6. Aspek Dokumentasi  
Berkas Rekam Medis mempunyai nilai dokumentasi karena isinya menyangkut sumber ingatan yang harus di dokumentasikan dan dipakai sebagai bahan pertanggung jawaban dan laporan rumah sakit.
7. Aspek Administrasi  
Sebagai bukti tertulis atas segala tindakan pelayanan, perkembangan penyakit dan pengobatan pasien selama berkunjung/dirawat di rumah sakit.

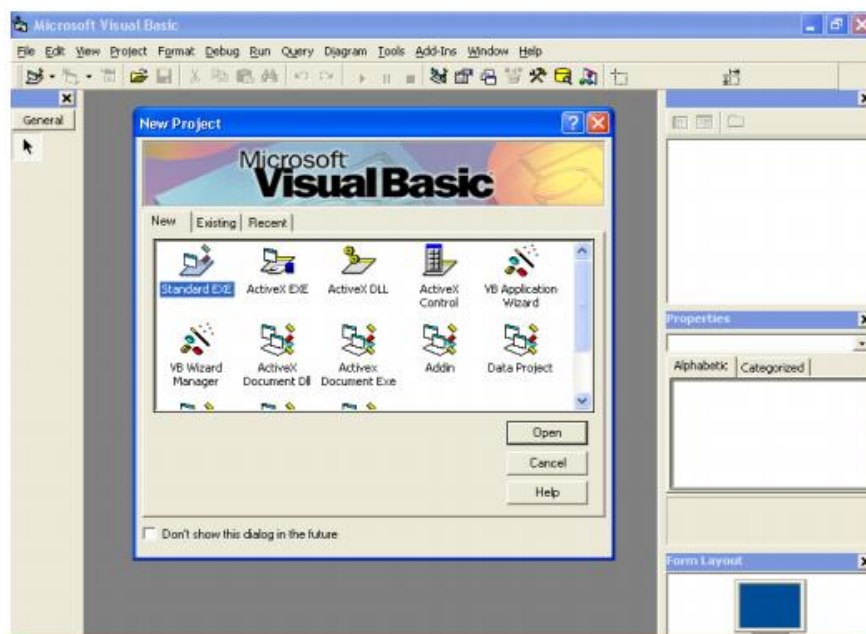
#### **2.4 Visual Basic 6**

Menurut (Pusparani, 2019), Visual Basic adalah salah satu bahasa pemrograman komputer. Bahasa pemrograman adalah perintah yang dimengerti oleh komputer untuk melakukan tugas-tugas tertentu. Bahasa pemrograman Visual Basic, yang dikembangkan oleh Microsoft sejak tahun 1991, merupakan pengembangan dari pendahulunya yaitu bahasa pemrograman BASIC (Beginner's All-purpose Symbolic Instruction Code) yang dikembangkan pada era 1950-an. Visual Basic merupakan salah satu Development Tool yaitu alat bantu untuk membuat berbagai macam program komputer, khususnya yang menggunakan sistem operasi Windows. Visual Basic merupakan salah satu bahasa pemrograman komputer yang mendukung object (Object Oriented Programming = OOP).

Aplikasi adalah suatu unit perangkat lunak yang dibuat untuk melayani kebutuhan akan beberapa aktivitas. Aplikasi akan menggunakan sistem operasi (OS) komputer dan aplikasi lainnya yang mendukung Apl. Istilah ini mulai perlahan masuk ke dalam istilah Teknologi Informasi semenjak tahun 1993. Secara historis, aplikasi adalah software yang dikembangkan oleh sebuah perusahaan. Bahasa pemrograman Visual Basic 6.0 dapat digunakan untuk menyusun dan membuat program aplikasi pada sistem operasi windows. Program aplikasi dapat berupa program database, program grafis dan lain sebagainya. Didalam Visual Basic 6.0

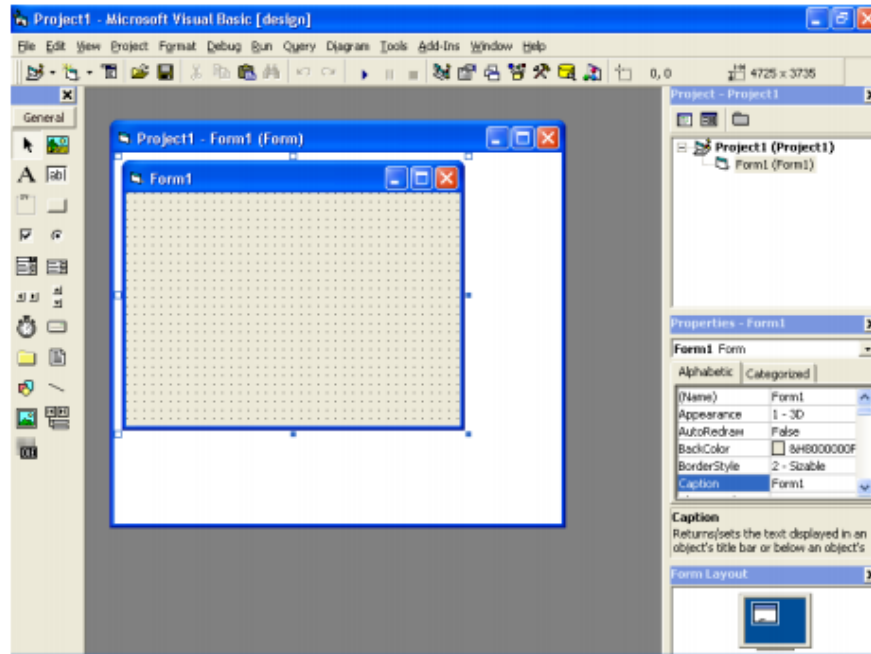
terdapat komponen - komponen yang sangat membantu dalam pembuatan program aplikasi. Dalam pembuatan program aplikasi pada Visual Basic 6.0 dapat didukung oleh software seperti Microsoft Access, Microsoft Exel, Seagate Crystal Report, dan lain sebagainya. Untuk dapat menyusun dan membuat suatu program aplikasi dari VB 6.0, tentunya harus mengetahui fasilitas – fasilitas yang disediakan agar proses penyusunan dan pembuatan program tersebut berjalan dengan baik.

Dalam pembuatan program aplikasi pada VB 6.0, langkah pertama adalah dengan membuat sebuah project. Adapun caranya dapat dilakukan dengan beberapa cara, diantaranya mengklik Start | program | Microsoft Visual Basic 6.0 | Microsoft Visual Basic 6.0. Cara lainnya adalah dengan membuka Windows Explorer dengan mengklik kanan Start dan mengklik Explore. Setelah explorer muncul klik akses program files\ Microsoft Visual Studio\ VB98 dan klik ganda vb6.exe. Selanjutnya akan terlihat tampilan seperti pada Gambar 2.1.



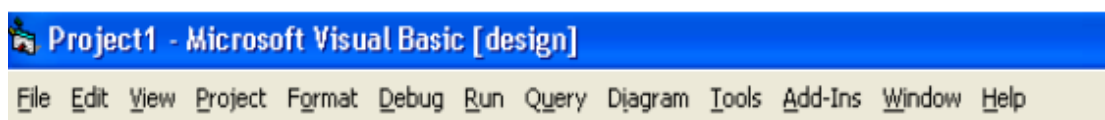
Gambar 2. 1 Tampilan Awal VB 6.0

Kemudian pilih Standart EXE lalu klik open maka akan terlihat tampilan from project seperti pada Gambar 2.2.



Gambar 2. 2 From Project

Menu bar merupakan salah satu fasilitas yang dapat digunakan untuk membantu user dalam membuat program aplikasi pada Visual Basic. Ada tigabelas menu yang dapat digunakan dan masing – masing mempunyai fungsi yang berbeda. Untuk menggunakan fasilitas menu, klik menu yang dipilih dan selanjutnya mengklik submenu yang akan digunakan.



Gambar 2. 3 Menu Bar

Toolbar berfungsi sama dengan menu, hanya saja berbeda tampilan. Pada toolbar cukup mengklik icon yang ingin digunakan yang terdapat pada toolbar. Jumlah icon

pada toolbar dapat diatur dengan mengklik Menu View | toolbars. Selanjutnya ada pilihan menambah toolbar, diantaranya Debug, Edit, Form editor, Standart, dan Customize. Pada submenu Customize terdapat pilihan untuk mengatur toolbar yang akan digunakan. Tampilan salah satu toolbar terlihat seperti pada Gambar 2.4.



Gambar 2. 4 Toolbar

Toolbox merupakan tempat icon – icon untuk objek yang akan dimasukkan dalam form pada pembuatan program aplikasi. Secara default pada toolbox hanya terdapat objek - objek seperti Gambar 2.5



Gambar 2. 5 Toolbox

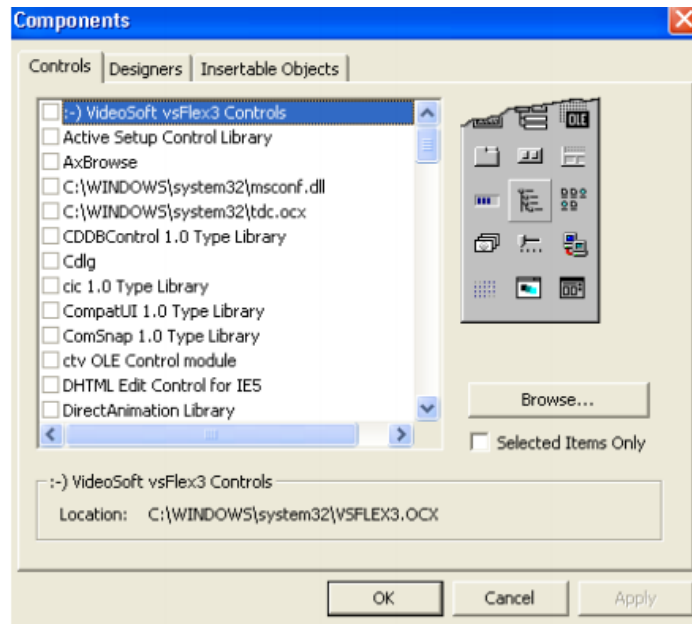


Secara garis besar fungsi dari masing-masing control tersebut adalah sebagai berikut :

1. Pointer  
Pointer bukan merupakan suatu kontrol tapi icon ini digunakan ketika memilih kontrol yang sudah berada pada form.
2. Label  
Label adalah kontrol yang digunakan untuk menampilkan text, yang tidak dapat diperbaharui.
3. Frame  
Frame adalah kontrol yang digunakan untuk mengidentifikasi sebuah group pengontrolan.
4. Check Box  
Check Box adalah kontrol yang digunakan untuk memilih satu atau beberapa check Box secara bersamaan.
5. Combo Box  
Combo Box adalah kontrol yang digunakan untuk mengetikkan pilihan atau untuk memilih item lewat Drop-Down List.
6. Horizontal Scroll Bar  
Horizontal Scroll Bar adalah kontrol yang digunakan untuk memungkinkan pemakai untuk memilih suatu objek selama dalam jangkauan Horizontal Objek.
7. Timer  
Timer adalah kontrol yang digunakan untuk mengoperasikan waktu kejadian pada rutin program termasuk internal waktu.
8. Directory List  
Box Directory List Box adalah kontrol yang digunakan untuk menampilkan daftar directory pada drive terpilih sehingga dapat dipilih sebuah Directiry dan path.
9. Shape  
Shape adalah kontrol yang digunakan untuk membentuk objek dua dimensi, bujur sangkar, lingkaran, empat persegi panjang, ellips.
10. Image  
Image adalah kontrol yang digunakan untuk menampilkan gambar bitmabs, windows, metafile, dan icon.
11. OLE  
OLE adalah kontrol yang digunakan untuk memungkinkan pemakai untuk menempelkan suatu objek dari aplikasi visual basic ke aplikasi yang mendukung OLE.
12. Data List dan Data Combo  
Data List dan Data Combo adalah kontrol yang digunakan untuk menampilkan data dalam sebuah Drop-Down List Box.
13. SSTab  
SSTab adalah objek berbentuk lembaran-lembaran, setiap lembaran berisi kelompok informasi.

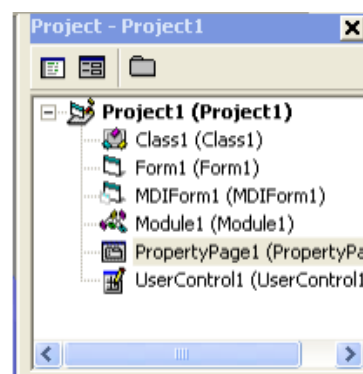
14. Picture Box  
Picture Box adalah kontrol yang akan menampilkan file gambar, dengan format Bitmaps, icon, Gif, Jpeg, dan sebagainya
15. TextBox  
TextBox adalah kontrol yang digunakan untuk menempatkan teks dalam form dan pemakai dapat mengedit teks tersebut.
16. Command Button  
Command Button adalah kontrol yang digunakan untuk memilih satu atau beberapa check Box secara bersamaan.
17. Option Button  
Option Button sering digunakan lebih dari satu sebagai pilihan terhadap beberapa option yang hanya dapat dipilih satu.
18. List Box  
List Box mengandung sejumlah item dan pemakai dapat memilih lebih dari satu
19. Vertikal Scroll Bar  
Vertical Scroll Bar adalah kontrol yang memungkinkan pemakai untuk memilih suatu objek dan digunakan untuk membentuk scroll bar berdiri sendiri.
20. Drive List Box  
Drive List Box digunakan untuk menampilkan daftar drive pada komputer pemakai dan memungkinkan untuk memilih sebuah drive.
21. File List Box  
File List Box digunakan untuk menampilkan daftar file pada directory terpilih dan memungkinkan untuk memilih sebuah drive.
22. line  
line adalah kontrol yang memungkinkan pemakai untuk membuat garis lurus.
23. Data dan Adodc  
Data dan Adodc digunakan untuk menampilkan database pada suatu form.
24. Data Grid  
Data Grid digunakan untuk menampilkan data dalam bentuk grid seperti di worksheet excel.

Tapi objek – objek pada toolbox dapat ditambah sesuai dengan kebutuhan user dalam pembuatan program aplikasinya. Cara menambah objek adalah dengan klik kanan pada toolbox, lalu pilih components, sehingga akan muncul tampilan seperti Gambar 2.6. selanjutnya pilih komponen - komponen kontrol / objek yang dibutuhkan dengan menklik /menceklis pada kotak pilihan dan klik OK.



Gambar 2. 6 Components

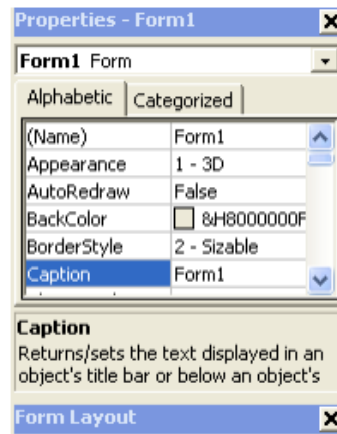
Project Explorer merupakan tempat untuk melihat daftar form, modules, dan design dengan mengklik kanan pada bagian project explorer dan pilih add, lalu pilih yang akan ditambah. Tampilan project explorer seperti pada gambar 2.7.



Gambar 2. 7 Project Explorer

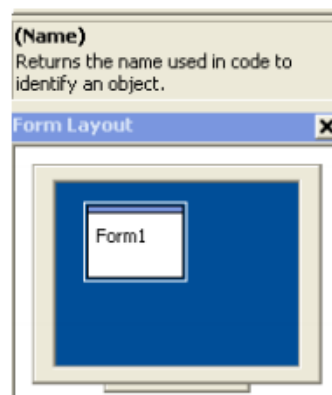
Properties Windows merupakan tempat yang digunakan untuk mengatur properti dari setiap objek kontrol. Pada properti windows ini semua objek control dapat

diatur sesuai dengan program aplikasi yang akan dibuat. Tampilan properties tampak seperti gambar 2.8.



Gambar 2. 8 Properties Windows

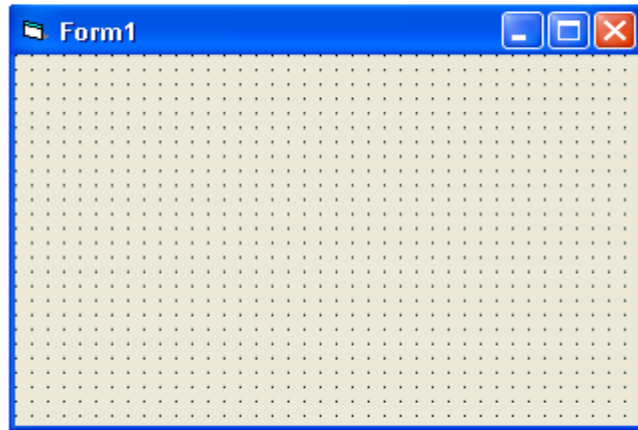
Form layout windows merupakan tempat untuk melihat posisi tampilan form saat dieksekusi atau program dijalankan. Untuk mengubah posisi tampilan saat dijalankan, klik pada form layout window dan atur sesuai dengan keinginan. Tampilan form layout windows seperti gambar 2.9.



Gambar 2. 9 Form Layout

Kontrol – kontrol pada toolbox akan diletakkan disini sesuai dengan rancangan program aplikasi. Untuk menampilkan form objek ini, klik ganda pada icon / nama

form pada jendela project explorer atau dengan klik kanan pada icon / nama pilih view object. Contoh tampilan form seperti gambar 2.10.

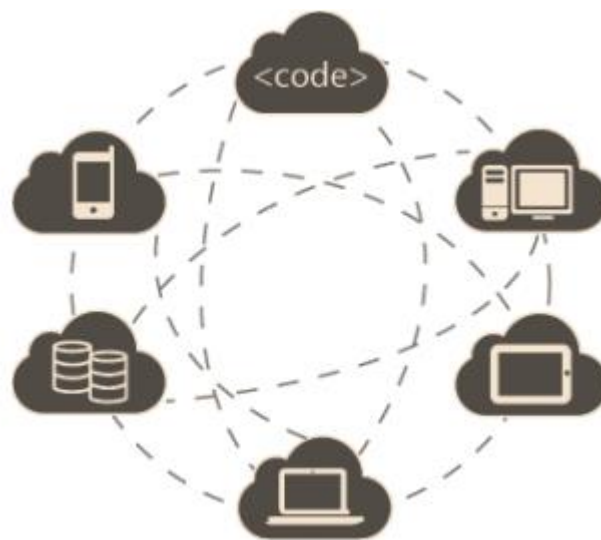


Gambar 2. 10 Form Layout

## 2.5 *Cloud computing*

*Cloud computing* adalah sebuah model yang memungkinkan untuk ubiquitous (Di manapun dan kapanpun), Nyaman, On-demand akses jaringan ke sumber daya komputasi (contoh: jaringan, server, storage, aplikasi, dan layanan) yang dapat dengan cepat dirilis atau ditambahkan. *Cloud computing* sebagai suatu layanan teknologi informasi yang dapat dimanfaatkan oleh pengguna dengan berbasis jaringan/internet. Dimana suatu sumber daya, perangkat lunak, informasi dan aplikasi disediakan untuk digunakan oleh komputer lain yang membutuhkan. *Cloud computing* mempunyai dua kata “Cloud” dan “Computing”. Cloud yang berarti internet itu sendiri dan Computing adalah proses komputasi. Konsep *Cloud computing* biasanya dianggap sebagai internet. Karena internet sendiri digambarkan sebagai awan (Cloud) besar (biasanya dalam skema jaringan, internet dilambangkan sebagai awan) yang berisi sekumpulan komputer yang saling

terhubung. *Cloud computing* datang sebagai sebuah evolusi yang mengacu pada konvergensi teknologi dan aplikasi lebih dinamis. Dimana terdapat perubahan besar memiliki implikasi yang menyentuh hampir setiap aspek komputasi. Untuk end user, Komputasi awan menyediakan sarana untuk meningkatkan layanan baru atau mengalokasikan sumber daya komputasi lebih cepat, Berdasarkan kebutuhan bisnis.



Gambar 2. 11 Gambaran umum *Cloud computing*

Empat Model pengembangan cloud.

1. Public Cloud Jenis Cloud ini diperuntukkan untuk umum oleh penyedia layanannya.
2. Private Cloud Merupakan infrastruktur layanan Cloud, yang dioperasikan hanya untuk sebuah organisasi tertentu. Infrastruktur Cloud itu bisa saja dikelola oleh sebuah organisasi itu atau oleh pihak ketiga. Lokasinya pun bisa on-site ataupun off-site. Biasanya organisasi dengan skala besar saja yang mampu memiliki/mengelola private Cloud ini.

3. Community Cloud Dalam model ini, sebuah infrastruktur Cloud digunakan bersama-sama oleh beberapa organisasi yang memiliki kesamaan kepentingan, misalnya dari sisi misinya, atau tingkat keamanan yang dibutuhkan, dan lainnya.
4. Hybrid Cloud yang menggabungkan baik public dan private. Untuk jenis ini, infrastruktur Cloud yang tersedia merupakan komposisi dari dua atau lebih infrastruktur Cloud (private, community, atau public). meskipun secara entitas mereka tetap berdiri sendiri, tapi dihubungkan oleh suatu teknologi / mekanisme yang memungkinkan portabilitas data dan aplikasi antar Cloud itu. Misalnya, mekanisme loadbalancing yang antar Cloud, sehingga alokasi sumberdaya bisa dipertahankan pada level yang Optimal.

## **2.6 Sejarah dan Perkembangan *Cloud computing***

Ide awal dari *cloud computing* pada tahun 1960-an, saat John McCarthy, pakar komputasi MIT yang dikenal juga sebagai salah satu pionir intelegensi buatan, menyampaikan visi bahwa “suatu hari nanti komputasi akan menjadi infrastruktur publik, seperti halnya listrik dan telepon”. Namun baru di tahun 1995 lah, Larry Ellison, pendiri Oracle, memunculkan ide “Network Computing” sebagai kampanye untuk menggugat dominasi Microsoft yang saat itu merajai desktop computing dengan Windows 95-nya. Larry Ellison menawarkan ide bahwa sebetulnya user tidak memerlukan berbagai software, mulai dari Sistem Operasi dan berbagai software lain, dijejalkan ke dalam PC Desktop mereka. PC Desktop bisa digantikan oleh sebuah terminal yang langsung terhubung dengan sebuah

server yang menyediakan environment yang berisi berbagai kebutuhan software yang siap diakses oleh pengguna.

Kehadiran konsep ASP (Application Service Provider) di akhir era 90-an. Seiring dengan semakin meningkatnya kualitas jaringan komputer, memungkinkan akses aplikasi menjadi lebih cepat. Hal ini ditangkap sebagai peluang oleh sejumlah pemilik data center untuk menawarkan fasilitasnya sebagai tempat „hosting“ aplikasi yang dapat diakses oleh pelanggan melalui jaringan komputer. Dengan demikian pelanggan tidak perlu investasi di perangkat data center. Hanya saja ASP (Application Service Provider) ini masih bersifat “private”, di mana layanan hanya dikastemisasi khusus untuk satu pelanggan tertentu, sementara aplikasi yang di sediakan waktu itu umumnya masih bersifat client-server. Kehadiran berbagai teknik baru dalam pengembangan perangkat lunak di awal abad 21, terutama di area pemrograman berbasis web disertai peningkatan kapasitas jaringan internet, telah menjadikan situs-situs internet bukan lagi berisi sekedar informasi statis. Tapi sudah mulai mengarah ke aplikasi bisnis yang lebih kompleks. Popularitas *Cloud computing* semakin menjulang saat di awal 2000-an, Marc Benioff ex VP di Oracle, meluncurkan layanan aplikasi CRM (Customer Relationship Management) dalam bentuk Software as a Service, Salesforce.com. Pada tahun 2005, mulai muncul inisiatif yang didorong oleh nama-nama besar seperti Amazon.com yang meluncurkan Amazon EC2 (Elastic Compute Cloud), Google dengan Google App Engine-nya, tak ketinggalan raksasa biru IBM meluncurkan Blue Cloud Initiative. Semua inisiatif ini masih terus bergerak, dan bentuk *Cloud computing* pun masih terus mencari bentuk terbaiknya, baik dari sisi praktis maupun dari sisi akademis.



Di Indonesia yang menawarkan layanan komputasi awan salah satunya ialah PT Telekomunikasi Indonesia, Layanan yang ditawarkan berupa aplikasi berbasis Software as a Service (SaaS) dan Infrastrukture as a Service (IaaS). PT Telkom (Telekomunikasi Indonesia) menawarkan layanan e-Office on Demand untuk kebutuhan kolaborasi atau korespondensi di dalam suatu perusahaan atau organisasi. Teknologi informasi dalam negeri masih kurang “peminat” terhadap *Cloud computing* ini, Beberapa faktor penyebabnya, di antaranya:

1. Penetrasi infrastruktur internet yang bisa dibilang masih terbatas, bandwidth masih terbatas
2. Tingkat kematangan pengguna internet, yang masih menjadikan media internet utamanya sebagai media hiburan atau sosialisasi.
3. Tingginya investasi yang dibutuhkan menyediakan layanan cloud ini, karena harus merupakan kombinasi antara infrastruktur jaringan, hardware dan software sekaligus.

## **2.7 *System Development Life Cycle (SDLC)***

*Software Development Life Cycle* atau SDLC merupakan sebuah proses pengembangan software yang digunakan untuk mengembangkan sebuah sistem informasi. SDLC mencakup kebutuhan (*requirement*), validasi, pelatihan, kepemilikan (*userownership*) dari sebuah sistem informasi yang diperoleh melalui investigasi, analisis, desain, implementasi, dan perawatan software. SDLC merupakan pendekatan sistematis untuk memecahkan masalah yang terdiri dari beberapa tahapan.

*Mode Waterfall* merupakan salah satu model dari *Software Development Life Cycle*. Mode waterfall merupakan dasar dari aktivitas proses yang terdiri dari spesifikasi, pengembangan, validasi, evolusi dan semua proses itu direpresentasikan dalam tahapan proses yang terpisah seperti spesifikasi kebutuhan, perancangan perangkat lunak, implementasi, pengujian, pemeliharaan dan sebagainya. Tahapan-tahapan dari Waterfall Model mencakup serangkaian kegiatan pengembangan yang mendasar. Tahapan tersebut adalah:

1. *Requirements Analysis and Definition*

Penentuan layanan sistem, batasan sistem dan juga tujuan ditetapkan setelah melakukan konsultasi dengan pengguna sistem (user). Hasil dari konsultasi tersebut kemudian didefinisikan lebih detail dan digunakan untuk spesifikasi sistem yang akan dibuat.

2. *System and Software Design*

Proses perancangan sistem ini membagi kebutuhan sistem menjadi dua bagian yaitu piranti lunak dan perangkat keras. Proses ini bertujuan untuk membangun arsitektur sistem secara keseluruhan. Perancangan piranti lunak dilakukan berdasarkan rangkuman identifikasi kebutuhan dan gambaran dasar sistem piranti lunak serta hubungan diantara keduanya yang telah dibahas pada tahap pertama.

3. *Implementation and Unit Testing*

Pada tahap ini, perancangan piranti lunak yang telah dilakukan pada tahap selanjutnya diimplementasikan menjadi sebuah program atau unit program. Pengujian ini berfungsi untuk memastikan apakah setiap unit program sesuai dengan spesifikasi yang telah dibahas pada tahap pertama.

4. *Integration and System Testing*

Tahap ini adalah penggabungan setiap unit program yang sudah dibuat. Setelah digabungkan menjadi sebuah sistem yang utuh dan lengkap, pengujian dilakukan untuk memastikan bahwa kebutuhan piranti lunak telah terpenuhi. Setelah pengujian selesai, sistem baru dikirimkan ke user.

5. *Operation and Maintenance*

Secara umum, tahap ini merupakan fase yang paling lama dilakukan. Sistem yang telah dikirimkan kemudian di-install dan digunakan untuk kepentingan yang sebenarnya. Proses maintenance terdiri dari pengkoreksian error yang tidak ditemukan pada tahap-tahap awal pengembangan, peningkatan implementasi dari unit sistem dan peningkatan layanan yang diberikan oleh sistem seiring ditemukannya kebutuhan baru.

## 2.8 Black Bock

Pengujian menggunakan sekumpulan aktifitas validasi, dengan pendekatan black box testing. Menurut (Rosa, 2011), black box testing adalah menguji perangkat lunak dari segi spesifikasi fungsional tanpa menguji desain dan kode program. Pengujian dimaksudkan untuk mengetahui apakah fungsi-fungsi, masukan, dan keluaran dari perangkat lunak sesuai dengan spesifikasi yang dibutuhkan. Pengujian kotak hitam dilakukan dengan membuat kasus uji yang bersifat mencoba semua fungsi dengan memakai perangkat lunak apakah sesuai dengan spesifikasi yang dibutuhkan.

Kasus uji yang dibuat untuk melakukan pengujian black box testing harus dibuat dengan kasus benar dan kasus salah. Menurut (Pressman, 2010), black box testing juga disebut pengujian tingkah laku, memusat pada kebutuhan fungsional perangkat lunak. Teknik pengujian black box memungkinkan memperoleh serangkaian kondisi masukan yang sepenuhnya menggunakan semua persyaratan fungsional untuk suatu program. Beberapa jenis kesalahan yang dapat diidentifikasi adalah fungsi tidak benar atau hilang, kesalahan antar muka, kesalahan pada struktur data (pengaksesan basis data), kesalahan performasi, kesalahan inisialisasi dan akhir program.

1. Equivalence Partitioning Equivalence Partitioning merupakan metode black box testing yang membagi domain masukan dari program kedalam kelas-kelas sehingga test case dapat diperoleh. Equivalence Partitioning berusaha untuk mendefinisikan kasus uji yang menemukan sejumlah jenis kesalahan, dan mengurangi jumlah kasus uji yang harus dibuat. Kasus uji yang didesain

untuk Equivalence Partitioning berdasarkan pada evaluasi dari kelas ekuivalensi untuk kondisi masukan yang menggambarkan kumpulan keadaan yang valid atau tidak. Kondisi masukan dapat berupa spesifikasi nilai numerik, kisaran nilai, kumpulan nilai yang berhubungan atau kondisi boolean. Kesetaraan kelas dapat didefinisikan menurut panduan berikut (Pressman, 2001) :

1. Jika masukan kondisi menentukan kisaran, satu sah dan dua diartikan tidak valid kesetaraan kelas.
2. Jika masukan membutuhkan nilai, kondisi tertentu satu sah dan dua tidak valid kesetaraan kelas diartikan.
3. Jika masukan kondisi menentukan anggota dari set, satu sah dan satu tidak valid kesetaraan kelas diartikan.
4. Jika kondisi yang input, boolean satu sah dan satu tidak valid kelas diartikan.

Sebagai contoh, pemeliharaan data untuk aplikasi bank yang sudah diotomatisasikan. Pemakai dapat memutar nomor telepon bank dengan menggunakan mikro komputer yang terhubung dengan password yang telah ditentukan dan diikuti dengan perintah-perintah. Data yang diterima adalah :

1. Kode Area : kosong atau 3 digit.
2. Prefix : 3 digit atau tidak diawali 0 atau 1
3. Suffix : 4 digit
4. Password : 6 digit alfanumerik
5. Perintah : check, deposit, dll

Selanjutnya kondisi masukan digabungkan dengan masing-masing data elemen, dapat ditentukan sebagai berikut :

1. Kode Area : kondisi masukan, Boolean-kode area mungkin ada atau tidak. Kondisi masukan, kisaran nilai ditentukan antara 200-999.
2. Prefix : kondisi masukan kisaran lebih besar 200 atau tidak diawali 0 atau 1.
3. Suffix : kondisi masukan nilai 4 digit.
4. Password : kondisi masukan Boolean-pw mungkin diperlukan atau tidak. Kondisi masukan nilai dengan 6 karakter string.
5. Perintah : kondisi masukan diatur dengan berisi perintah-perintah yang telah didefinisikan.

Menerapkan pedoman untuk derivasi kelas kesetaraan, uji kasus untuk setiap masukan domain item data dapat dikembangkan dan dilaksanakan. Uji kasus dipilih sehingga jumlah terbesar dari atribut dari kelas kesetaraan tersebut dilakukan sekaligus. Beberapa kata kunci dalam pengujian perangkat lunak yang dapat diperhatikan, yaitu (Simarmata, 2009) :

1. Dinamis

Pengujian perangkat lunak dilakukan pada masukan yang bervariasi. Masukan ini ditentukan sebelum pengujian dilakukan dengan batasan yang disesuaikan dengan kemampuan perangkat lunak. Masukan tidak harus sesuatu yang dimungkinkan terjadi pada penggunaan program lebih lanjut, melainkan meliputi keseluruhan batasan yang dapat dijangkau perangkat lunak dan dilakukan pemercontohan (sampling) secara acak untuk proses pengujian.

## 2. Terbatas

Meskipun pengujian dilakukan pada perangkat lunak sederhana sehingga rumit sekalipun, pengujian dilakukan dengan memenuhi batasan-batasan tertentu sesuai dengan kemampuan program. Batasan ini juga diberlakukan pada masukan-masukan yang dipilih untuk pengujian. Tidak semua kemungkinan masukan diujikan pada perangkat lunak karena akan memakan waktu yang cukup panjang mengingat begitu banyaknya kemungkinan yang bisa terjadi. Untuk mengatasi hal ini, pemilihan masukan-masukan pada proses pengujian secara acak yang diperkirakan mampu memenuhi kebutuhan pengujian perangkat lunak akan dilakukan.

## 3. Tertentu

Pengujian dilakukan dengan batasan tertentu disesuaikan dengan harapan pada fungsi, respon, dan karakteristik perangkat lunak tersebut. Batasan tersebut akan disesuaikan dengan teknik-teknik pengujian yang ada. Pemilihan kriteria pengujian yang paling tepat merupakan hal yang kompleks. Dalam praktiknya, analisis risiko pengujian dan pengalaman terhadap pengujian-pengujian sejenis akan diperlukan.

## 4. Harapan

Kata kunci ini memiliki keadaan-keadaan yang diharapkan, baik berupa respon sistem terhadap masukan maupun karakteristik responnya. Dalam hal ini, batasan-batasan hasil pengujian yang

diharapkan harus ditentukan. Dengan demikian, dapat diketahui apakah perangkat lunak tersebut telah memenuhi hasil pengujian yang diharapkan atau memerlukan pembenahan kembali, baik berupa perbaikan maupun pengembangan perangkat lunak