

PENGEMBANGAN APLIKASI *SECURE MESSAGE* BERBASIS ANDROID

Udin Sidik Sidin¹⁾

Abstrak: Penelitian ini adalah penelitian pengembangan perangkat lunak yang bertujuan untuk membangun aplikasi *secure message* berbasis android. Aplikasi ini dikembangkan dengan menggunakan model *prototyping*. Model *prototyping* terdiri dari beberapa tahapan-tahapan: pengumpulan kebutuhan, membangun *prototyping*, mengkodekan sistem, pengujian sistem. Hasil yang diperoleh dalam penelitian ini adalah aplikasi *secure message* berbasis android yang dapat meningkatkan kinerja bagi pengguna android. Hasil pengujian dengan menggunakan *blackbox*, *whitebox* dan validasi *expert* maka aplikasi ini layak untuk digunakan. Tanggapan pengguna (*user*) terhadap aplikasi ini berada pada kategori baik.

Kata Kunci : *Secure message service*, Android, Algoritma RC6

PENDAHULUAN

Perkembangan teknologi telekomunikasi yang ada pada saat ini mampu menciptakan berbagai macam perangkat keras yang dapat digunakan untuk mengirim atau menerima informasi dengan cepat dan mudah. Penggunaan *handphone* sebagai *device* akses informasi telah berkembang pesat pada era ini. Terlebih lagi, banyak aplikasi *mobile* yang diciptakan, membuat informasi-informasi yang dibutuhkan mudah untuk diperoleh.

Menurut Defni (2014), perangkat *mobile* saat ini yang disebut dengan *smartphone* memiliki fitur dari teknologi terbaru untuk menjalankan berbagai fungsi layaknya sebuah komputer biasanya, menjadi alasan utama kenapa teknologi *smartphone* menjadi lebih diminati dan menyebabkan produsen *smartphone* menjadi sangat kompetitif di pasaran. Adapun beberapa sistem yang saat ini populer digunakan pada perangkat *smartphone* seperti IOS, *BlackBerry*, *WindowsPhone* dan juga android.

Sistem operasi pada *smartphone* android menjadi yang paling diunggulkan oleh para pengguna dan juga produsen *smartphone* karena fiturnya yang sangat menarik. Android yang merupakan salah satu sistem operasi *opensource* memiliki banyak peminat sehingga memunculkan juga banyak pengembang (*developer*), sehingga android mendukung perkembangan yang cepat, karena seperti *opensource* lainnya android membuka *code* sumbernya secara gratis untuk dikembangkan oleh para *developer*.

Mengenai keamanan pada layanan SMS, Defni (2014) menjelaskan bahwa pengamanan atau *security* dari layanan penyedia SMS juga sangat perlu diperhatikan karena penggunaannya yang sangat dominan pada sebuah perangkat telepon dan juga memiliki celah keamanan yang besar yang mungkin menyebabkan datanya diketahui oleh pihak yang tidak dikehendaki, seperti pihak operator memang terkadang menjanjikan kepada pelanggan bahwasanya keamanan akan dijamin dan tidak akan terjadi penyadapan

¹⁾ adalah dosen Fakultas Teknik Universitas Negeri Makassar, Jl. AP. Pettarani, Makassar

data oleh pihak yang tidak berhak, namun pada kenyataannya data yang dikirimkan melalui *service* SMS bisa saja akan tersadap atau diketahui dengan gampang saat data SMS tersimpan pada data *center* sebuah *provider*. Oleh karena itu sebaiknya *user* terlebih dahulu menggunakan aplikasi yang bisa membuat data terenkripsi sebelum dikirimkan melalui *service* SMS sehingga *provider* hanya akan meneruskan pesan yang sudah dienkripsi pada perangkat *user* pengirim.

Metode enkripsi yang dapat digunakan untuk aplikasi *mobile* adalah RC6 (Rivest Code 6). RC 6 merupakan pengembangan dari algoritma sebelumnya yaitu RC5. Algoritma RC6 dilengkapi dengan beberapa parameter, sehingga dituliskan sebagai RC6-w/t/b. Parameter w merupakan ukuran kata dalam satuan bit, parameter merupakan bilangan bukan negatif yang menunjukkan banyaknya iterasi selama proses enkripsi dan parameter b menunjukkan ukuran kunci enkripsi dalam *byte*.

Penelitian penggunaan metode RC6 pernah dilakukan oleh Arfiandinata (2015). Menggunakan metode enkripsi dan dekripsi algoritma RC6 namun aplikasi ini tidak menyimpan pesan (masih bersifat *inbox* dan *outbox*) dan aplikasi belum dapat mengirimkan langsung kunci bersamaan dengan pesan teksnya. Selanjutnya Mangundap (2015) juga menggunakan algoritma CR6 untuk membuat aplikasi *secure message* berbasis android namun pesan yang dikirim hanya data text berupa huruf, angka dan symbol sehingga untuk pengiriman gambar belum bisa

dilakukan. Aplikasi tersebut juga tidak kompatibel dengan android versi Jelly Bean 4.2 keatas. Berdasarkan uraian tersebut penulis bermaksud untuk mengembangkan aplikasi keamanan pengiriman pesan melalui SMS untuk mengenkripsi data yang berjalan pada sistem operasi android, sehingga diharapkan pemilik *handphone* yang berbasis android dapat melakukan pertukaran data yang lebih aman.

Layanan yang digunakan pada aplikasi SMS bawaan ponsel masih banyak digunakan oleh setiap *user*, dan bukan merupakan jalur yang aman dalam pertukaran informasi. Pesan yang dikirim menggunakan aplikasi SMS bawaan ponsel masih berupa teks terbuka yang belum *terproteksi* (terlindungi), selain itu pengiriman SMS yang dilakukan tidak sampai ke penerima secara langsung, akan tetapi pengiriman SMS harus lewat *Short Message Service Center* (SMSC) yang berfungsi mencatat komunikasi yang terjadi antara pengirim dan penerima. Dengan tersimpannya SMS pada SMSC, maka seorang operator dapat memperoleh informasi atau membaca SMS di dalam SMSC tersebut. Maka dibutuhkan suatu metode dan aplikasi yang dapat mempertimbangkan dengan melakukan enkripsi terhadap pesan SMS. Enkripsi adalah proses mengubah suatu pesan asli yang disebut *plaintext* (pesan yang dapat dibaca) menjadi sebuah sandi atau kode yang tidak terbaca yang disebut *ciphertext* (pesan yang terkunci) dan tidak dapat dimengerti, untuk mengembalikan pesan ke bentuk asli

seperti semula diperlukan suatu proses yang disebut dekripsi.

Enkripsi yang umum digunakan yaitu menggunakan algoritma enkripsi dan kunci yang dapat diubah-ubah sesuai kesepakatan untuk meningkatkan keamanan. Teknik ini disebut sebagai algoritma kunci simetris (*symetric key*) yaitu suatu enkripsi dengan menggunakan kunci yang sama untuk melakukan proses enkripsi dan dekripsi. Contoh algoritma kunci simetris yaitu Rivest Code 6 (RC6) yang dirancang oleh Ronald L Rivest, M.J.B. Robshaw, R. Sidney dan Y.L. Yin, algoritma ini merupakan pengembangan dari

algoritma sebelumnya yaitu RC5. Perangkat lunak yang akan dibangun merupakan perangkat lunak yang diterapkan pada telepon selular yang bersistem operasi android dan memiliki fungsi untuk melakukan enkripsi dan dekripsi. Perangkat lunak dapat melakukan pengiriman dan penerima pesan. Pengguna akan berinteraksi dengan perangkat lunak melalui *user interface* yang disediakan oleh perangkat lunak, pengguna memasukkan data dengan menggunakan *keypad* (simbol) yang dimiliki oleh telepon selular. Pesan yang telah dibuat dikirimkan ke telepon selular lain melalui jaringan SMS.



Gambar 1 Cara kerja SMS
(Nazruddin Safaat H, 2015)

PENGERTIAN ANDROID

Android merupakan sebuah sistem operasi yang berbasis *linux* untuk telepon selular seperti telepon pintar dan komputer, adapun pengertian android menurut para ahli sebagai berikut :

1) Safaat (2011) mengatakan android adalah sebuah sistem operasi untuk perangkat *mobile* berbasis *linux* yang mencakup sistem operasi, *middleware*, dan aplikasi. Android menyediakan *platform* terbuka

bagi para pengembang untuk menciptakan aplikasi mereka.

- 2) Menurut Yosef Murya (2014), "android adalah sistem operasi berbasis *linux* yang digunakan untuk telepon selular (*mobile*) seperti telepon pintar (*smartphone*) dan komputer tablet.
- 3) Menurut Hermawan (2011), android merupakan OS (*Operating System*) *Mobile* yang tumbuh ditengah OS lainnya yang berkembang dewasa ini. OS lainnya seperti

Windows Mobile, i-Phone OS, Symbian, dan masih banyak lagi. Akan tetapi, OS yang ada ini berjalan dengan memprioritaskan aplikasi inti yang dibangun sendiri tanpa melihat potensi yang cukup besar dari aplikasi pihak ketiga.

Berdasarkan beberapa pengertian diatas dapat disimpulkan bahwa android merupakan sistem operasi *open source* yang dikembangkan oleh perusahaan ternama Google *inc* yang digunakan untuk *smartphone* dan *tablet PC* namun saat ini beberapa perangkat lain juga sudah mulai menggunakan sistem operasi ini antara lain yaitu jam tangan pintar (*smartwatch*), televisi pintar (*smart TV*), bahkan beberapa mobil mewah.

Menurut Safaat (2011), android dapat dikatakan sebagai *platform* masa depan karena memiliki 3 kriteria, yakni:

- 1) Lengkap (*complete platform*), para desainer dapat melakukan pendekatan yang komprehensif ketika mereka sedang mengembangkan *platform* android. Android merupakan sistem operasi yang aman dan banyak menyediakan *tools* dalam membangun *software* dan memungkinkan untuk peluang pengembangan aplikasi.
- 2) Terbuka (*open source platform*), *platform* android disediakan melalui lisensi *open source*. Pengembangan dapat dengan bebas untuk peluang mengembangkan aplikasi. Android sendiri menggunakan *kernel 2.6*.

3) *Free (free platform)*, android adalah *platform* atau aplikasi yang bebas untuk *developer*. Adapun kelebihan menggunakan android yaitu:

- a) Tidak terdapat lisensi atau biaya royalti untuk dikembangkan pada *platform* android.
- b) Tidak diperlukan keanggotaan.
- c) Tidak diperlukan biaya pengujian.
- d) Tidak diperlukankontrak.

Android dapat didistribusikan dan diperdagangkan dalam bentuk apapun. Android *Software Development Kit* (SDK) merupakan *tools* bagi para *programmer* yang ingin mengembangkan aplikasi berbasis Google android. Android *Software Development Kit*(SDK) mencakup seperangkat alat pengembangan yang komprehensif. Android *Software Development Kit* (SDK)terdiri dari *debugger*, *libraries*, *handset emulator*, dokumentasi, contoh kode, dan *tutorial*. Saat ini android sudah mendukung arsitektur x86 pada Linux (distribusi Linux apapun untuk *desktop* modern), Mac OS X 10.4.8 atau lebih, Windows XP atau Vista. Persyaratan mencakup JDK, Apache Ant dan Python 2.2 atau yang lebih baru. IDE yang didukung secara resmi adalah Eclipse 3.2 atau lebih dengan menggunakan *pluginAndroid Development Tools* (ADT), dengan ini pengembang dapat menggunakan teks editor untuk mengedit file Java dan XML serta menggunakan peralatan *command line* untuk menciptakan, membangun, melakukan *debug* aplikasi android dan pengendalian perangkat android

(misalnya, *reboot*, menginstal paket perangkat lunak dengan jarak jauh).

PENGETIHAN ALGORITMA RC6

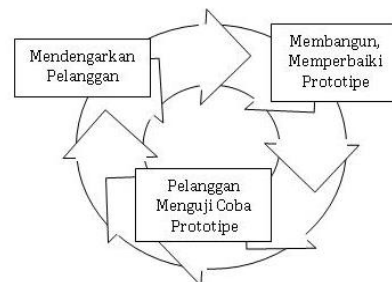
Algoritma RC6 merupakan algoritma sederhana, fungsi yang digunakan merupakan fungsi yang sederhana dan hanya mengandalkan prinsip *iterated cipher* untuk keamanan. Tampilan hasil enkripsi dan data hasil enkripsi yang diterima harus diperhatikan, hal ini dikarenakan pada data hasil enkripsi, setiap karakternya akan memiliki panjang 8 bit. Dengan demikian dalam perancangan algoritma RC6 pada SMS karakter-karakter yang akan dienkripsi diubah kedalam nilai ASCII, di mana nilai karakter dalam table ASCII ditambah *table* karakter *special* adalah 0 sampai dengan 255, artinya satu karakter ASCII akan diwakili oleh 8 bit, dimana $2^8 = 256$. Sehingga, dalam 1 blok plainteks (32 bit) akan menyimpan 4 karakter dan setiap kali iterasi, maka akan diambil 16 karakter dari plainteks.

Panjang plain teks atau panjang kunci kurang dari 16 karakter, maka akan dilakukan *padding*, yaitu dengan menambah bit "0" (nol) di akhir teks, sehingga panjang teks mencukupi 116 karakter. Layar pada sebagian besar telepon selular hanya dapat menampilkan karakter dengan panjang 7 bit dan pesan yang telah terenkripsi akan berbentuk binary, sehingga layar tidak akan menampilkan dengan semestinya. Oleh karena itu, pada aplikasi yang akan dibangun, untuk menampilkan pesan yang telah terenkripsi, ditambahkan informasi karakter yang terdapat pesan tersebut dengan format heksadesimal agar dapat

ditampilkan dilayar dan informasinya lebih terbaca. Algoritma RC6 yang akan digunakan dalam aplikasi SMS yang akan dibangun dengan w sebesar 32 bit, r sebesar 20 kali putaran dan panjang kunci beragam lebih dari 1 karakter (8 bit).

METODE PENELITIAN

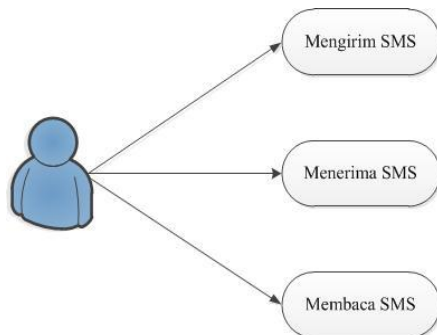
Penelitian ini adalah penelitian pengembangan perangkat lunak yang bertujuan untuk mengembangkan aplikasi *secure message* berbasis android menggunakan algoritma RC6. Adapun tahapan penelitian dalam pengembangan sistem ini yaitu dengan menggunakan model pengembangan yang terdapat pada metode pengembangan SDLC, yaitu *prototyping*. *Sistem Development Life Cycle* (SDLC) atau siklus hidup pengembangan sistem adalah kerangka kerja (*framework*) yang berisi proses-proses sekuensial di mana sistem informasi dikembangkan (Turban, 2003), sedangkan model *prototyping* adalah proses pembuatan model sederhana untuk *softwarefinal* yang mengizinkan pengguna memiliki gambaran dasar tentang program serta melakukan pengujian awal. Berikut ini tahapan-tahapan dalam model *prototyping*:



Gambar 2. Tahapan Prototyping

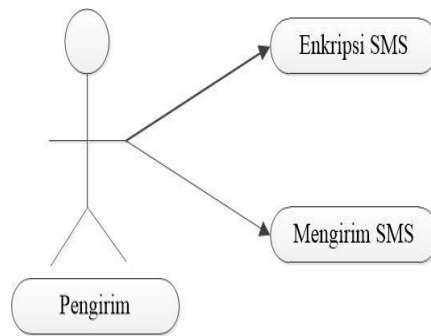
PERANCANGAN USE CASE DIAGRAM

Use case diagram adalah suatu gambaran dari beberapa atau semua *use case*, dan interaksi diantara komponen-komponentersebut yang memperkenalkan suatu sistem yang akandibangun atau dikembangkan dengan tujuan untuk menjelaskan bagaimana langkah-langkah yang seharusnya dikerjakan oleh sistem. *Use case diagram* menjelaskan manfaat suatu sistem jika dilihat menurut pandangan orang yang berada di luar sistem. Diagram ini menunjukkan fungsionalitas suatu sistem atau kelas dan bagaimana sistem tersebut berinteraksi dengan dunia luar.



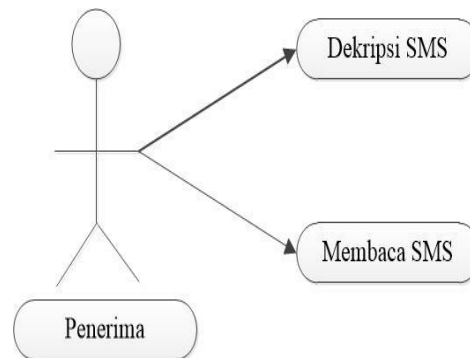
Gambar 3
Use case utama

Gambar 3.2 adalah *use case* utama, pengguna mengetik pesan pada teks box yang nantinya akan diambil karakter-karakter yang ada didalamnya. Kemudian pesan akan diterima oleh nomor yang dituju. Penerima dapat membaca pesan secara normal jika memiliki aplikasi yang sama.



Gambar 4
Use case kirim

Gambar 4 adalah *use case* kirim, sebelum mengirim pesan maka pengirim mengenkripsi pesan kemudian menulis pesan, setelah menulis pesan maka pengirim dapat mengirimkan pesan tersebut.



Gambar 5
Use case terima

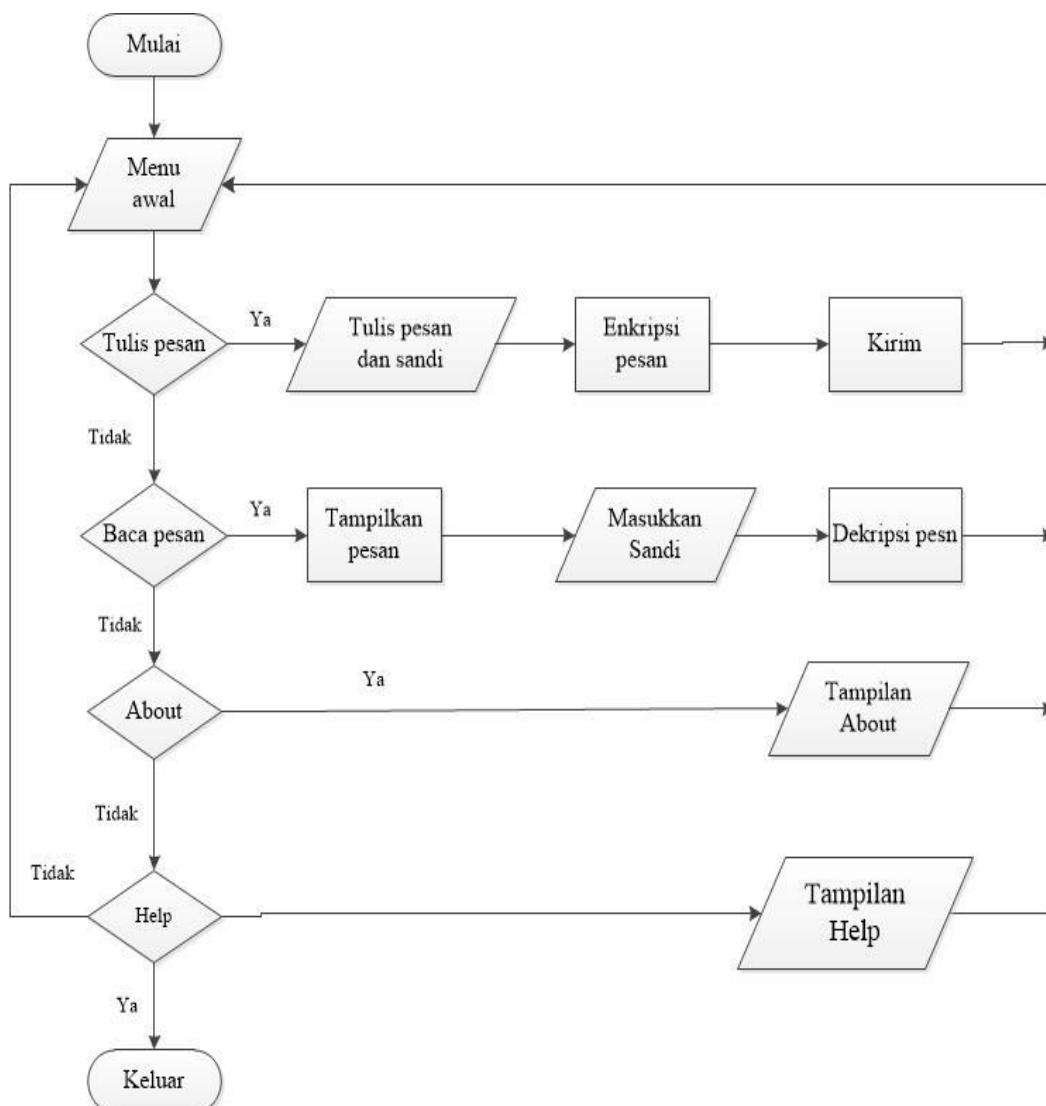
Gambar 5 adalah *use case* terima, pesan dapat dibaca ketika pesan tersebut sudah didekripsi oleh penerima pesan yang memiliki aplikasi yang sama.

PERANCANGAN FLOWCHART SISTEM

Flowchart adalah suatu bagan dengan simbol-simbol tertentu yang menggambarkan urutan proses secara mendetail dan hubungan antara suatu

proses (instruksi) dengan proses lainnya dalam suatu program. *Flowchart* berfungsi menganalisis alur sistematika program. Gambar 6 adalah *flowchart* aplikasi SMS, Pada halaman awal atau menu awal dari program *user* hanya akan melihat empat buah tombol yaitu tulis pesan, baca pesandan *about*. Pada tombol “Tulis Pesan” berfungsi untuk menampilkan layer baru yang

berguna untuk menulis *plain text*, aplikasi akan menjalankan fungsi untuk menghasilkan sebuah *chipertext* yang akan dikirim ke penerima SMS, kemudian pada tombol ”baca pesan” akan mengaktifkan fungsi untuk melihat kotak masuk dari pesan yang terdapat pada inbox *smartphone user*.

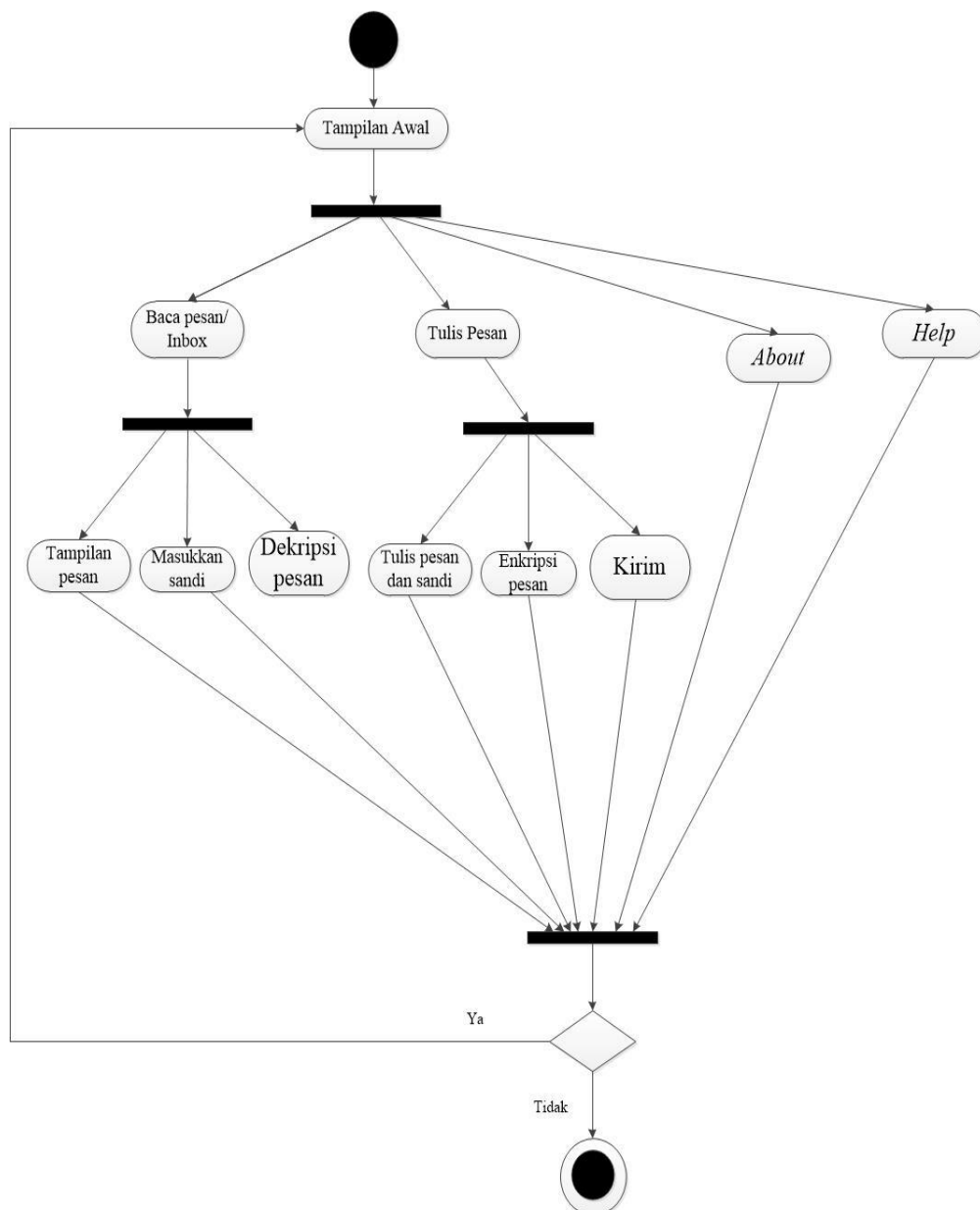


Gambar 6
Flowchart Aplikasi SMS

PERANCANGAN ACTIVITY DIAGRAM

Activity diagram menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, *action* yang

terjadi, dan bagaimana mereka berakhir. *Activity diagram* juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi. Dari *use case diagram* diatas akan dijelaskan *activity diagram*, sebagai berikut:



Gambar 7
Activity diagram aplikasi SMS

PENGUJIAN SISTEM

Pengujian sistem dalam penelitian ini dilakukan dalam beberapa tahap yaitu sebagai berikut :

a. Metode *black box*

Black box testing adalah pengujian yang dilakukan hanya mengamati hasil eksekusi melalui data uji dan memeriksa fungsional dari perangkat lunak. Jadi dianalogikan seperti kita melihat suatu kotak hitam, kita hanya bisa melihat penampilan luarnya saja, tanpa tau ada apa dibalik bungkus hitamnya. Sama seperti pengujian *black box*, mengevaluasi hanya dari tampilan luarnya(interface nya).

b. Pengujian *white box*

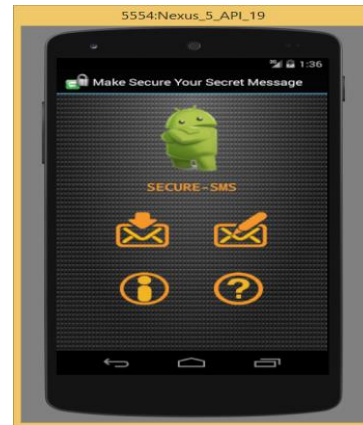
Pengujian *white box* adalah pengujian yang didasarkan pada pengecekan terhadap detail perancangan, menggunakan struktur kontrol dari desain program secara procedural untuk membagi pengujian ke dalam beberapa kasus pengujian.

HASIL DAN PEMBAHASAN

Struktur aplikasi *secure message* berbasis Android, terdiri dari beberapa menu yang memiliki fungsi tersendiri. Menu-menu tersebut akan tampil sesuai yang diinginkan setelah *user* menggunakan aplikasi tersebut.

Halaman utama merupakan halaman awal yang terdapat 4 tombol yang memiliki fungsi masing-masing yaitu, tombol tulis pesan untuk menuju *activity* tulis pesan. Tombol kotak masuk berfungsi menuju *activity* *inbox*. Tombol *help* berfungsi untuk panduan penggunaan aplikasi. Sedangkan tombol *about* berfungsi untuk identitas pengembang aplikasi.

Tampilannya ditunjukkan pada gambar 8



Gambar 8
Tampilan awal

a. *Form* tulis pesan

Pada *form* tulis pesan ini pengguna diharapkan untuk memasukkan beberapa data sub-sub menu yang ada, berikut adalah sub menu dan fungsinya gambar 9.



Gambar 9
Tampilan tulis pesan

b. *Form* baca pesan

Pada *form* baca pesan ini pengguna akan menerima pesan yang dalam kondisi masih terenkripsi, berikut

adalah sub tampilan baca pesan gambar 10.



Gambar 10
Tampilan baca pesan

1. **Pengujian sistem**
 - a. Pengujian *Blackbox*

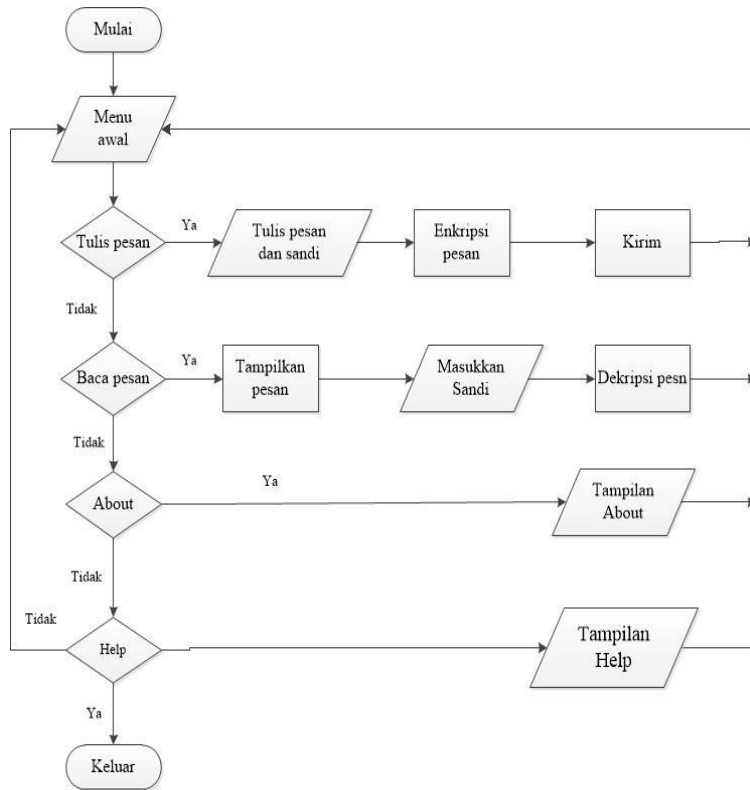
Aplikasi yang dibangun ini diuji dan dianalisa agar penerapan teori ke dalam program dapat sejalan. Sehingga jika di cek dengan baik secara manual menghasilkan presentase kemungkinan dan hasil yang tidak berbeda. Daftar *use case*, metode pengujian dan kriteria evaluasi hasil pengujian pada *user* dapat dilihat pada gambar tabel 1

- b. Pengujian *Blackbox* dan *White Box*

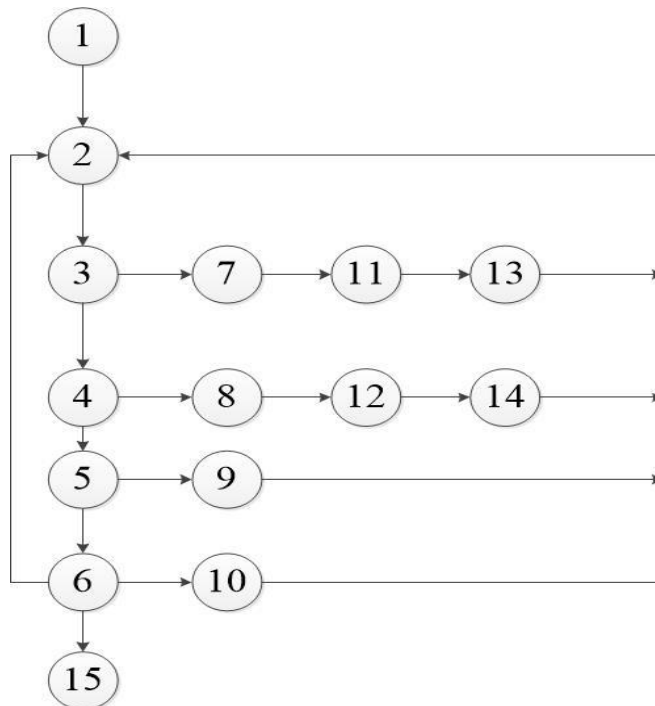
Teknik pengujian ini merupakan pengujian terhadap cara kerja perangkat lunak itu sendiri yaitu basis *path* (prosedur programnya) atau proses *looping* (perulangan), dengan proses pengujian *black box* dan *white box*.

Tabel 1
Hasil Uji Coba *Blackbox*

Skenario Uji	Hasil Yang Diharapkan	Kesimpulan
Menampilkan Beranda Aplikasi	Tampil Beranda Aplikasi	Berhasil
Tekan Tombol Menulis Pesan	Tampil Foam Untuk menulis pesan	Berhasil
Tekan kolom untuk nomor tujuan	Tampilkan nomor tujuan yang akan dikirimkan pesan	Berhasil
Tekan kolom kunci	Tampilkan foam untuk tulis kunci pesan	Berhasil
Tekan tombol pengirim pesan	Tampilan untuk untuk mengirim pesan	Berhasil
Tekan tombol baca pesan	Tampil Pesan yang akan di baca/di buka	Berhasil
Tekan Menu About	Tampil Tentang Developer Aplikasi	Berhasil
Tekan tombol menu Help	Tampil petunjuk penggunaan aplikasi	Berhasil



Gambar 11
Flowgraph aplikasi



Gambar 12
Flowchart aplikasi

Proses perhitungan pada gambar 11 dan 12 *flowchart* dan *flowgraph* adalah :

1) Flowgraph mempunyai 6 *region*

2) Untuk menghitung

Cyclomatic Complexity $V(G)$

E = Jumlah *Edge* (panah) = 19

N = Jumlah *Node* (lingkaran)

= 15

$V(G)$

= $E - N + 2$

= $19 - 15 + 2$

= 6

Path-path yang terdapat pada *flowgraph* yaitu :

1) Path 1 : 1, 2, 3, 4, 5, 6, 15

2) Path 2 : 1, 2, 3, 4, 5, 6, 2

3) Path 3 : 1, 2, 3, 7, 11, 13, 2

4) Path 4 : 1, 2, 3, 4, 8, 12, 14, 2

5) Path 5 : 1, 2, 3, 4, 5, 9, 2

6) Path 6 : 1, 2, 3, 4, 5, 6, 10, 2

Hasil dari pengujian diperoleh 5 *region* yaitu R1, R2, R3, R4, R5, R6 artinya terdapat 6 daerah

yang dibatasi oleh *edge* dan *node*. Kemudian diperoleh 5 *Cyclomatic Complexity* $V(G)$, artinya terdapat 6 kondisi pada program yang menghubungkan *node* awal dengan *node* akhir. Selanjutnya terdapat 6 *path* (jalur) yang menjelaskan setiap alurnya menuju menu yang dipilih. Berdasarkan rekapitulasi hasil perhitungan diatas diperoleh hasil *Cyclomatic Complexity* (CC) = 6, *Region* (R) = 6, dan *Independent Path* (IP) = 6. Karena *Cyclomatic Complexity* (CC), *Region* (R), *Independent Path* (IP) menunjukkan hasil yang sama, maka pengujian ini terbebas dari kesalahan logika.

TANGGAPAN USER

Menganalisis sistem ini penulis membagikan 32 lembar penilaian kepada responden. Berdasarkan hasil analisis deskriptif, diperoleh kecenderungan data seperti pada tabel 2

Tabel 2

Hasil analisis deskriptif

Deskripsif	Nilai
Mean	76.62
Median	77.00
Mode	80
Std. Deviation	3.883
Variance	15.081
Kurtosis	-1.129
Std. Error of Kurtosis	.809
Range	12
Minimum	70
Maximum	82
Sum	2452

Berdasarkan tabel 2, diketahui rata-rata (*mean*) sebesar 80,503; Median sebesar 77.00; Modus sebesar 80; Nilai Maksimum 82; Nilai Minimum 70 ; dan standar

deviasi sebesar 3.883. Berdasarkan nilai deskriptif diatas, tanggapan pengguna (*user*) terhadap sistem aplikasi *secure message* berbasis android ditujukan pada tabel 3.

Tabel 3
Distribusi Frekuensi

No.	Kategori	Interval	FA	PR (%)	PK (%)
1.	Sangat Baik	80,503 Ke atas	12	37,5	37,5
2.	Baik	76,62 – 80,503	9	28,13	65,63
3.	Cukup Baik	72,737 – 76,62	6	18,75	84,38
4.	Kurang Baik	Di bawah 72,737	5	15,63	100
Jumlah			32	100	

Berdasarkan tabel distribusi frekuensi dapat dilihat bahwa 12 orang atau 37,5% berada pada kategori sangat baik, 9 orang atau 28,13% berada pada kategori baik, 6 orang atau 18,75 % berada pada kategori cukup baik dan 5 orang atau 15,63% berada pada kategori kurang baik terhadap aplikasi *secure message*.

VALIDASI SISTEM

Validasi sistem aplikasi *secure message* berbasis Android adalah proses untuk menilai kelayakan sistem sebelum digunakan aplikasi *secure message* berbasis Android. Pengujian ini dilakukan oleh pakar yang telah diberikan kepercayaan dalam memvalidasi aplikasi *secure message* berbasis Android dengan hasil persentase kelayakan sebagai berikut :

Tabel 4
Hasil Persentase Kelayakan Expert

Expert	Skor Perolehan	Skor Maksimal	Presentase Kelayakan
<i>Expert 1</i>	28	32	87,5%
<i>Expert 2</i>	26	32	81,25%

Berdasarkan tabel diatas menunjukkan hasil dari *Expert* pertama adalah 87,5%, dan hasil *Expert* kedua adalah 81,25%. Maka sesuai dengan table deskripsi data validator ahli mediamaka dapat disimpulkan bahwa *Expert* pertama dan kedua menunjukkan bahwa aplikasi *secure message* berada pada kategori Sangat Layak/sangat baik.

Perancangan aplikasi *Secure message* berbasis android menggunakan metode *prototyping*, tahap yang dilakukan yaitu mendengarkan pelanggan, membangun *prototipe*, mengujicoba sistem. Perancangan aplikasi ini menggunakan *android studio* untuk pengkodean sistem yang berjalan pada Windows 8.1. Adapun berepa

tampilan icon dirancang menggunakan button.

Pengujian aplikasi ini melalui beberapa tahap yaitu pengujian *white box*, *black box*, validasi ahli (*expert*), dan tanggapan pengguna. Berdasarkan tahapan tersebut aplikasi ini layak untuk diimplementasikan karena pada tahapan pertama terlihat dari hasil pengujian *white box* yang telah dilakukan dimana *Cyclomatic Complexity* (CC), *Region* (R), dan *Independent Path* (IP) menunjukkan hasil yang sama, jika ketiga nilai tersebut di atas sama, maka sistem dikatakan dapat berjalan dengan baik sesuai dengan kriteria rancangan sebelumnya.

Tahapan kedua yaitu *black box testing* dimana peneliti melakukan uji coba dengan mencoba semua menu, tombol, dan fungsi lainnya apakah berjalan sesuai dengan yang diharapkan. Uji coba juga dilakukan kepada responden terhadap aplikasi, dengan membagikan angket kepada responden untuk mengetahui pendapat responden terhadap aplikasi. Berdasarkan tabel distribusi frekuensi dan histogram dapat dilihat bahwa 12 orang atau 37,5% berada pada kategori sangat baik, 9 orang atau 28,13% berada pada kategori baik, 6 orang atau 18,75 % berada pada kategori cukup baik dan 5 orang atau 15,63% berada pada kategori kurang baik terhadap aplikasi *secure message*.

KESIMPULAN

Berdasarkan hasil penelitian dan pembahasan yang telah diuraikan sebelumnya, maka dapat diambil kesimpulan bahwa aplikasi *secure*

message berbasis android dapat dikembangkan dengan menggunakan model pengembangan *prototyping*. Pesan yang dikirimkannya data text berupa huruf, angka *symbol* sehingga pengiriman gambar belum bisa dilakukan.

DAFTAR PUSTAKA

- Abinaufal Reza. 12 April 2014. *Penelitian Ilmiah Tentang Aplikasi*. (<http://rezaabinaufal.blogspot.co.id/2014/04/12/peneitian-ilmiah-tentang-aplikasi.html>, diakses 31 Januari 2016).
- Aep. 06 Desember 2009. *Sekilas Tentang Telekomunikasi Selular*. (<https://anantoep.wordpress.com/2009/12/16/sekilas-tentang-sistem-komunikasi-seluler/>, diakses 04 Februari 2016).
- Aellyas, 03 Septeember 2011. *Konsep Dasar Komunikasi Selular*. (<https://aellyas.wordpress.com/2011/09/03/konsep-dasar-komunikasi-seluler/>, diakses 04 Februari 2016).
- Arfiandinata, Dwi Nanda. 2015. *Secure Text Message Dengan Metode Enkripsi Dekripsi Algoritma RC6 Berbasis Android*. STMK MDP, Palembang. Diakses pada 15 february 2016, dari <http://mdp.ac.id>
- Arifanto, Rifay. 31 Maret 2014. *Pengertian perbedaan white box*. (<http://rivayiarifanto.blogspot.co.id/2014/03/pengertian-perbedaan-white-box-dan.html>, diakses 04 Februari 2016).

- Anggraini Erianti, 19 April 2015. *Perkembangan Telekomunikasi di Indonesia*. (<http://eriantianggraini.blogspot.co.id/2015/04/perkembangan-telekomunikasi-di-indonesia.html>, diakses 04 Februari 2016).
- Ariyus, Dony. 2008. *Pengantar Ilmu Kriptografi*. Yogyakarta: ANDI
- Baraza, Haidi. 06 Juli 2013. *Pengertian Android*. (<https://haidibarasa.wordpress.com/2013/07/06/pengertian-android-sdk-software-development-kit/>, diakses 31 Januari 2016).
- Binanto, Iwan. 2005. *Konsep Dasar Program*. Yogyakarta : PT. Elex Media Komputindo.
- Defni, 2014. *Enkripsi SMS (Short Message Service) Pada Telepon Selular Berbasis Android Dengan Metode RC6*. Jurnal Momentum Vol. 16 No. 1. Halaman 63-73.
- Febrian, Jack. 2004. *Istilan Teknologi Informasi*. Bandung : Informatika
- Hartono M, Jogyanto (1993). *Konsep Dasar Pemrograman Bahasa C*. Yogyakarta : Andi
- Hermawan Susanto, Stephanus. 2011. *Mudah Membuat Aplikasi Android*. Yogyakarta : C.V ANDI OFFSET.
- Herdi, Hafizh. 09 september 2014. *Belajar Membuat Aplikasi Android Menggunakan android Studio*. <http://www.twoh.co/2014/09/belajar-membuat-aplikasi-android-menggunakan-android-studio/>, diakses 29 mei 2016).
- Indra, Muhammad 2014. *Implementasi Algoritma RC6 Untuk Dekripsi dan Enkripsi SMS Berbasis Android*, STMIK AMIKOM, Yogyakarta, Diakses pada September 2014, dari <http://repository.amikom.ac.d>
- Kurniawan, Yusuf, Ir. 2004. *Kriptografi Keamanan Internet dan Jaringan Komunikasi*. Bandung : Informatika
- Mangundap, Rionald Ricardo. 2015. *Aplikasi Secure Message Menggunakan Algoritma CR6 Berbasis Android*. e-Jurnal Spirit Pro Patria Vol. 1 No. 2. Halaman 96-110.
- Muttaqien, Ikhwanul. 2014. *Sejarah Perkembangan Aplikasi Mobile*, (On line), (<https://prezi.com/ng0df3j1zvd/sejarah-perkembangan-aplikasi-mobile/>, diakses 22 Juli 2015).
- Murya, Yosef. 2014. *Pemrograman Android Black Box*. Jakarta: Jasakom.
- Oktafiana, Gian Dwi. 2015. *Pengembangan Media Pembelajaran Berbasis Android dalam Bentuk Buku Saku Digital Untuk Mata Pelajaran Akuntansi Kompetensi Dasar Membuat Ikhtisar Siklus Akuntansi Perusahaan Jasa di Kelas XI MAN 1 Yogyakarta Tahun ajaran 2014-2015*. Yogyakarta : Universitas Negeri Yogyakarta. Diakses 15 Februari 2015.
- Rickyanto, Isak. 2003. *Dasar Pemrograman Berorientasi*

- Objek dengan Java 2: JDK 1.4.* Yogyakarta : Andi Offset.
- Rizky Dhanta. 2009. *Pengantar Ilmu Komputer.* Surabaya: INDAH.
- Safaat, H. Nazruddin. 2011. *Android Pemrograman Aplikasi Mobile Smartphone dan Tablet PC Berbasis Android.* Bandung : Informatika Bandung.
- Safaat H Nazruddin. 2015. *Implementasi dan Pengembangan Aplikasi Mobile Berbasis Android.* Bandung : Informatika Bandung.
- Salahuddin, M dan Rosa, A.S. 2013. *Modul Pembelajaran Rekayasa Perangkat Lunak (Terstruktur & Berorientasi Objek.* Bandung: Modula
- Sugiyono. 2005. *Pemrograman Terstruktur.* Kuningan : Panji Gumilang Press.
- Sutarman. 2009. *Pengantar Teknologi Informasi.* Yogyakarta : PT. Bentang Pusaka.
- No name. 2013. *Skripsi informatikan bab 1.* (<http://www.slideshare.net/slayemail/skripsi-informatika-bab-1-revisi-20-agustus-2013>, diakses 11 februari)
- Winarno, Edy, Ali Zaki, SmitDev Community. 2012. *Membuat Sendiri Aplikasi Android untuk Pemula.*Jakarta: PT Elex Media Komputindo
- Yasin, V. 2012. *Rekayasa Perangkat Lunak Berorientasi Objek Pemodelan, Arsitektur Dan Perancangan (Modeling, Architecture And Design).*Jakarta: Mitra Wacana Media.
- Yuswanto. 2008. *Algoritma dan Pemrograman dengan Visual Basic.Net 2005.* Jakarta: