

PERANCANGAN SISTEM OTOMATIS UPDATE PADA APLIKASI DESKTOP ABIOS

Karto Iskandar

Jurusan Teknik Informatika, Fakultas Ilmu Komputer, Bina Nusantara University
Jln. K.H. Syahdan No. 9 Kemanggisan, Palmerah, Jakarta Barat 11480
karto_i@binus.edu

ABSTRACT

Unlike web applications easier to update the latest version, desktop applications more difficult and must involve the user in doing so. It is caused by a desktop application is an application that is installed in the computer user. The purpose of this research is to design an automatic system updates on a desktop application, an example case: Application Binus International Operational Support (ABIOS). This research used literature study and system design. In desktop applications, often there is update the latest applications that are not known to the user who sometimes fatal and disrupt business operations. Generally, developer will inform the changes version to user that they can update the application. In an update of applications, should be done by the system automatically, not manually by users. Once in a while, the user background is not from computer base. After doing the research, it can be concluded that the system automatically updates the application has benefits to users in obtaining information regarding the latest version, and can assist in automatically update the latest application is based on computerization. For further development of this system is expected to operate on multi platforms and or mobile applications.

Keywords: *system, automatic update, application desktop, update application, ABIOS*

ABSTRAK

Berbeda dengan aplikasi web yang lebih mudah dalam update versi terbaru, aplikasi desktop lebih sulit dan harus melibatkan user dalam melakukannya. Hal tersebut disebabkan karena Aplikasi desktop adalah aplikasi yang terinstal didalam komputer user. Tujuan penelitian ini adalah merancang sistem otomatis update pada aplikasi desktop, contoh khusus: Application BiNus International Operational Support (ABIOS). Penelitian ini menggunakan metode studi pustaka dan perancangan sistem. Pada aplikasi desktop, sering ada update aplikasi terbaru yang tidak diketahui oleh user yang terkadang berakibat fatal dan mengganggu bisnis berjalan. Umumnya pengembang aplikasi menginformasikan adanya perubahan versi tersebut kepada user agar user dapat melakukan update aplikasi. Dalam melakukan update aplikasi, seharusnya dapat dilakukan oleh sistem secara otomatis guna mempermudah user dengan tidak melakukan tahapan-tahapan update aplikasi secara manual yang sulit diingat oleh user yang umumnya bukan berlatar belakang dari komputer. Setelah melakukan penelitian, dapat disimpulkan bahwa sistem otomatis update aplikasi memiliki manfaat terhadap user dalam mendapatkan informasi adanya versi terbaru, dan dapat membantu dalam otomatis update aplikasi terbaru tersebut berbasis komputerisasi. Untuk pengembangan lebih lanjut diharapkan sistem ini dapat beroperasi pada multi platform dan ataupun aplikasi mobile.

Kata kunci: *sistem, otomatis update, aplikasi desktop, update aplikasi, ABIOS*

PENDAHULUAN

Dalam proses Rekayasa Piranti Lunak, pergerakan perubahan software sejalan dengan Proses Bisnis terasa sangat cepat, terutama untuk bisnis yang prosesnya terus berkembang. Software aplikasi yang digunakan untuk menunjang proses bisnis dapat kita bedakan antara desktop aplikasi dan *web* aplikasi. Aplikasi untuk proses operasional lebih cocok untuk dijalankan dalam model desktop, karena disebabkan alasan kinerja dan keamanan. Sedangkan aplikasi web memiliki sejumlah kelemahan terutama dalam hal keamanan (Satoto, 2009). Perbaikan dan penambahan pada sistem aplikasi terutama model desktop terkadang hanya di ketahui oleh pihak IT dan user terkait, tanpa diketahui user bagian lain. Masalah ini cukup mengganggu, karena user tidak tahu dan sulit mendapatkan informasi versi terbaru dari aplikasi yang dipakai. Proses ataupun cara umum yg dipakai untuk menginformasikan adanya versi terbaru tersebut adalah, pihak IT yang menghubungi user-user terkait yang memakai aplikasi tersebut. Masalah menjadi besar seiring dengan jumlah user yang bertambah.

Universitas Bina Nusantara adalah universitas yang berlokasi di Indonesia dan menjalankan proses bisnisnya dengan dukungan banyak aplikasi dan teknologi maju. Banyak Sistem Informasi yang dikembangkan oleh tim IT Directorate yang digunakan oleh Binus baik itu merupakan desktop aplikasi maupun web aplikasi. Pemilihan desktop ataupun web aplikasi tentunya disesuaikan dengan kebutuhan user dan pemakaian aplikasi tersebut. Salah satu aplikasi desktop yang digunakan adalah *Application BiNus International Operational Support* (ABIOS). ABIOS adalah aplikasi desktop yang digunakan Universitas Bina Nusantara khususnya area The Joseph Wibowo Center (JWC). Aplikasi ABIOS digunakan untuk membantu proses administrasi operasional pada program Binus International. Untuk user dengan unit area yang tersebar luas harus menggunakan satu sistem aplikasi yang terintegrasi untuk meningkatkan integritas / *real time* dan kualitas (Choldun, 2006).

Untuk mendukung proses operasionalnya, di JWC terbagi menjadi beberapa area dengan *job desk* dan tanggung jawab yang berbeda. Semua user operasional di JWC tersebut menggunakan satu aplikasi yang telah terintegrasi dalam ABIOS. Untuk meningkatkan kualitas ABIOS, sering dilakukan perbaikan dan perubahan disesuaikan dengan perubahan proses bisnisnya, seperti menurut Saputri (2010). Perubahan ini harus diinformasikan ke semua user pengguna dan mereka harus mengupdate aplikasi. Perbedaan versi aplikasi dapat berdampak serius terhadap data operasional dan akurasi proses. Akan terasa lebih nyata dampaknya, jika aplikasi tersebut mengolah data penting, seperti nilai, ijazah, ataupun keuangan. Karena versi aplikasi yang berbeda tentunya terdapat proses yang berbeda didalamnya.

Dua masalah utama disini adalah, bagaimana menginformasikan kepada user bahwa ada versi aplikasi terbaru dan bagaimana user mengupdate aplikasi. Masalah pertama muncul karena disebabkan jumlah user yang banyak dan tidak teralokasi pada satu area saja. Selain itu, staff operasional juga memiliki shift jam kerja yang tidak sama, ada yang shift pagi, siang dan sore. Masalah kedua muncul karena disebabkan tidak semua user berlatar belakang dari IT atau tidak mengerti teknis komputer. Masalah kedua ini merupakan masalah klasik yang muncul disebabkan oleh ketidakpahaman user akan cara update aplikasi yang sifatnya terlalu teknis IT. Sedangkan seperti kita ketahui, tidak semua user operasional memiliki *background* IT. Disisi lain staff IT tidak bisa selalu ada dan siap untuk membantu staff operasional dalam update aplikasi dikarenakan banyaknya tugas staff IT. Oleh karena itu alangkah baiknya jika sistem aplikasi yang dipakai staff operasional dapat mengupdate dirinya sendiri secara pintar, otomatis dan mudah. Tujuan dari penelitian ini adalah (1) melakukan analisis terhadap perubahan proses pengupdatean aplikasi di sisi user, sehingga user tidak perlu dibebankan dengan proses teknis IT; (2) Merancang aplikasi kecil yang pintar dan dapat menginformasikan adanya versi terbaru pada aplikasi kepada user; (3) Merubah aplikasi berjalan dengan menambahkan macro untuk cek versi setiap aplikasi mulai dijalankan; (4) Merancang aplikasi kecil yang mengupdate aplikasi utama.

METODE

Metode penelitian yang digunakan adalah menggunakan metode studi pustaka dan perancangan sistem. Pada metode studi pustaka, dilakukan pengumpulan bahan-bahan pustaka baik yang dilakukan di perpustakaan maupun pencarian melalui internet. Bahan-bahan yang dikumpulkan ada yang bersumber dari *textbook* ataupun jurnal yang dengan topik terkait. Metode ini berguna dalam membantu memperdalam pembahasan materi, pembuatan sistem dan penyusunan karya tulis.

Pada metode perancangan sistem, dilakukan proses-proses pengembangan sistem berdasarkan teori rekayasa piranti lunak dengan rincian proses sebagai berikut. Analisis proses berjalan dilakukan guna menemukan kesulitan user dan menemukan pola kerja untuk proses baru yang lebih mempermudah user. Setelah mengetahui proses berjalan, beserta kesulitan-kesulitan yang biasa ditemui user, penulis melakukan studi pustaka dan metode baru melalui internet guna menyelesaikan masalah yang diketemukan. Metode perancangan sistem dilakukan untuk mengubah sistem berjalan dengan melakukan tambahan proses dan aplikasi untuk mempermudah user melakukan update aplikasi berjalan. Pada metode perancangan ini juga dikaitkan dengan teori Interaksi Manusia Komputer (IMK) dalam perancangan *User Interface* yang baik sehingga sistem akan mudah digunakan oleh user. Diakhir proses, dilakukan ujicoba terhadap sistem berjalan, yaitu *Application BiNus International Operational Support* (ABIOS), dan meminta user untuk melakukan testing sebelum sistem otomatis update ini diimplementasi

Aplikasi desktop

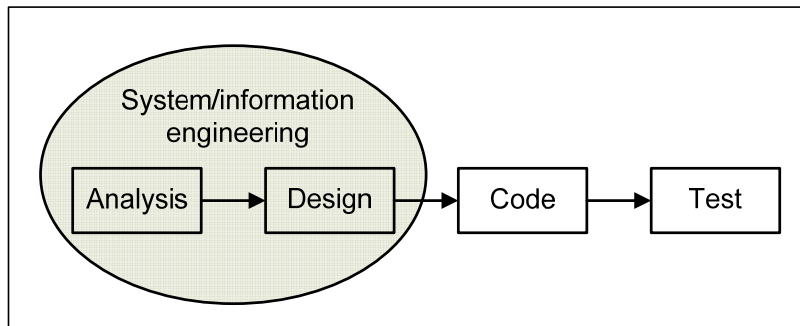
Aplikasi desktop merupakan aplikasi yang tersedia pada komputer client dan dikontrol penggunaannya oleh user. Aplikasi desktop dijalankan pada drive lokal drive dan tidak memerlukan jaringan atau untuk berfungsi dengan baik, meskipun jika memerlukan koneksi jaringan aplikasi ini juga berfungsi dengan baik.

Pada komputer, sebuah *desktop environment* (DE) (lingkungan meja) umumnya merujuk ke gaya grafis antarmuka pengguna (GUI=*Graphical User Interface*) yang memperlihatkan kiasan meja kerja di layar komputer modern. DE adalah alternatif yang paling populer sebagai pengganti antarmuka baris perintah (CLI=*command line interface*) yang saat ini umumnya digunakan terbatas oleh para pakar komputer. Sebuah DE biasanya terdiri dari ikon, sistem jendela (*window*), *toolbar*, *folder*, latar belakang (*wallpaper*), dan *desktop widgets*. Sistem operasi Microsoft Windows dan Mac OS X masing-masing mempunyai DE tersendiri.

Rekayasa Piranti Lunak (RPL)

Menurut pressman (2001) yang dimaksud dengan piranti lunak adalah (1) kumpulan instruksi (program komputer) yang jika dieksekusi akan menyediakan fungsi dan dayaguna yang diinginkan; (2) kumpulan struktur data yang memungkinkan program untuk memanipulasi informasi dengan memadai; dan (3) kumpulan dokumen yang menggambarkan operasi dan penggunaan program. Rekayasa piranti lunak adalah pembuatan dan penggunaan prinsip keahlian teknik untuk mendapatkan piranti lunak yang ekonomis yang handal dan bekerja secara efisien pada mesin nyata.

Proses Rekayasa Piranti Lunak secara linear menurut Pressman (2001) dengan model *waterfall*, model linear ini menyusulkan sebuah pendekatan yang sistematis dan sekuensial terhadap pengembangan piranti lunak.



Gambar 1 Proses Rekayasa *Software*

Tahapan dalam rekayasa piranti lunak meliputi proses analisa, desain, code dan testing. Masing-masing proses tersebut saling terkait satu sama lainnya, dapat dilihat pada gambar 1. Menurut Pressman (2001), proses sistem atau informasi *engineering* terjadi pada tahap analisa dan desain.

Interaksi Manusia dan Komputer (IMK)

Menurut Shneiderman (2004), IMK adalah disiplin ilmu yang berhubungan dengan perancangan, evaluasi, dan implementasi sistem komputer interaktif untuk digunakan oleh manusia serta studi fenomena besar yang berhubungan dengannya. Perancangan dan evaluasi itu berkaitan dengan *user interface*. *User interface* adalah bagian sistem komputer yang memungkinkan manusia berinteraksi dengan komputer yang bertujuan agar sistem komputer dapat digunakan oleh *user*. Tujuan timbulnya IMK adalah agar tercipta fungsional yang sesuai pada sistem tersebut. Suatu sistem diharapkan dapat berjalan sesuai dengan kebutuhan *user*-nya. Sistem yang memungkinkan terjadinya IMK menuntut adanya keandalan, ketersediaan, keamanan, dan integritas data. IMK yang baik adalah interaksi yang memperhatikan Keseragaman, Terintegritas, Konsistensi, dan Portabilitas.

Menurut Shneiderman (2004), lima faktor manusia terukur (*five measureable human factors*) dalam merancang suatu tampilan yang *user friendly* adalah Waktu belajar, Kecepatan kinerja, Tingkat kesalahan, Daya ingat, dan Kepuasan subjektif. Prinsip IMK adalah Kondisi dan aksi alternatif harus dapat terlihat oleh *user*. Harus ada konseptual yang baik dan gambaran sistem yang konsisten. *Interface* yang dibuat harus memiliki pemetaan yang baik, yang mencakup hubungan tiap tingkatnya. *User* harus mendapatkan *feedback* yang terus-menerus. Konsistensi yang harus dicapai menurut Shneiderman adalah bahwa suatu urutan aksi haruslah tersusun dengan urutan, dapat diduga, dijelaskan dengan beberapa aturan sehingga mudah dipelajari dan diingat oleh *user* serta mudah bagi *user* untuk menyesuaikan diri.

HASIL DAN PEMBAHASAN

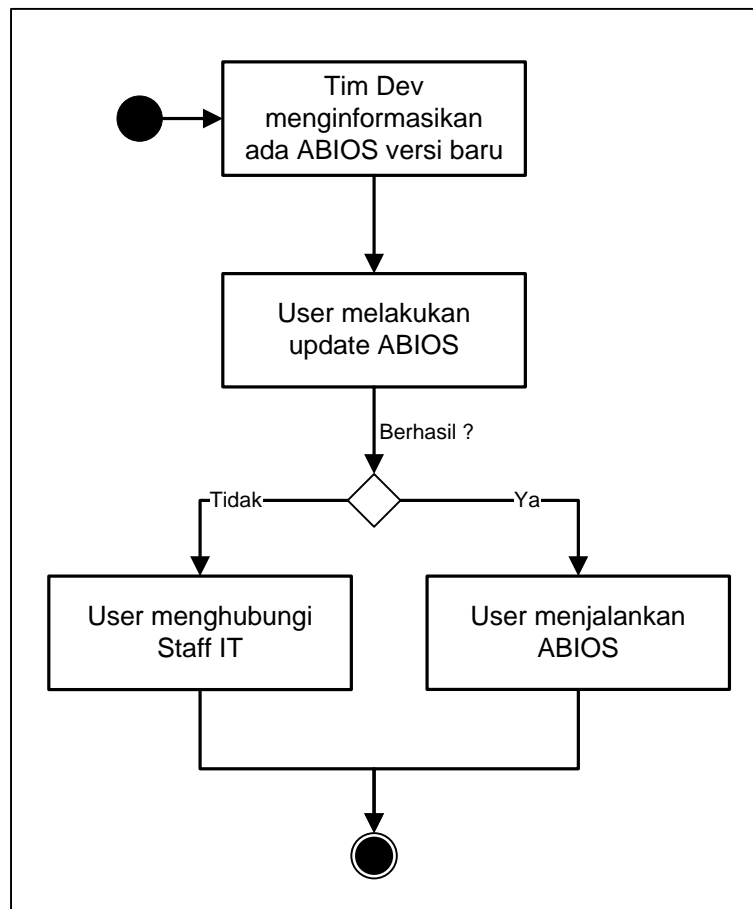
Pada pembahasan ini, penulis akan menjabarkan Sistem update aplikasi yang berjalan pada sistem aplikasi di JWC, staf operasional. Penulis akan menjelaskan rincian tahapan yang terjadi, guna mempermudah ilustrasi, juga akan disertakan tampilan layar serta tahapan tahapan yang dilalui staf operasional.

Analisis Sistem Berjalan

Saat ini, pengembangan aplikasi yang dipakai di JWC dilakukan oleh Direktorat IT. Untuk

perubahan, penambahan ataupun perbaikan aplikasi BIOS, dimulai dari staff operation melakukan permintaan untuk perubahan sistem berjalan. Kemudian staff Direktorat IT akan melakukan analisa dan mempelajari perubahan tersebut. Setelah ada rancangan solusi, maka Direktorat IT akan mempresentasikan kepada user, negosiasi dilakukan sampai menemukan sebuah design yang disepakai oleh kedua belah pihak.

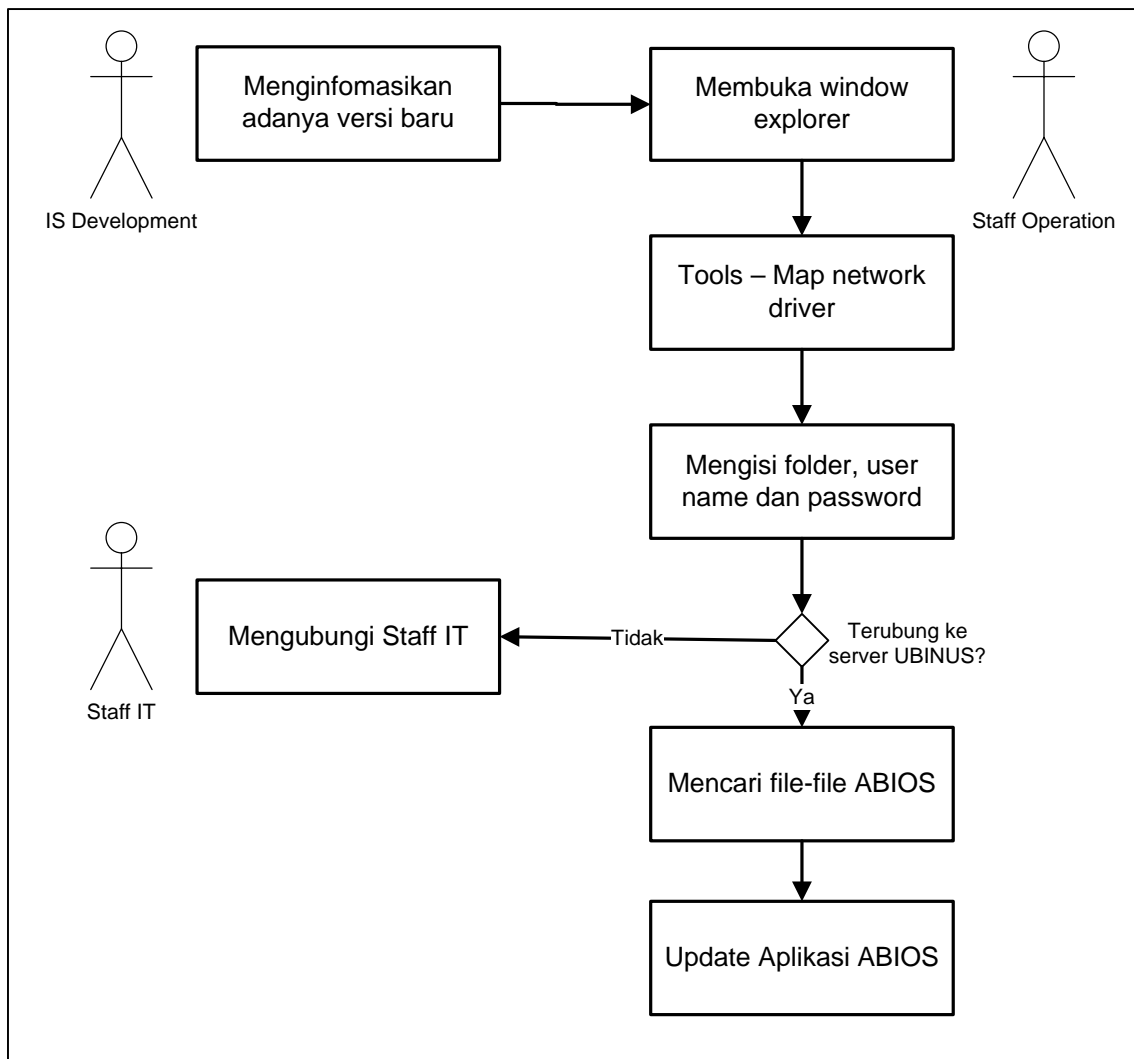
Proses dilanjutkan dengan Direktorat IT mengubah aplikasi BIOS (tahap coding) kemudian dilanjutkan pada tahap internal testing. Sampai pada kondisi dimana sudah tidak ada error, maka BIOS akan di serahkan ke user untuk melakukan beta testing (testing oleh user). Setelah tidak ada error lagi, baru aplikasi akan di-*deploy* disertai dengan penutupan projek.



Gambar 2 Proses update aplikasi (sistem berjalan)

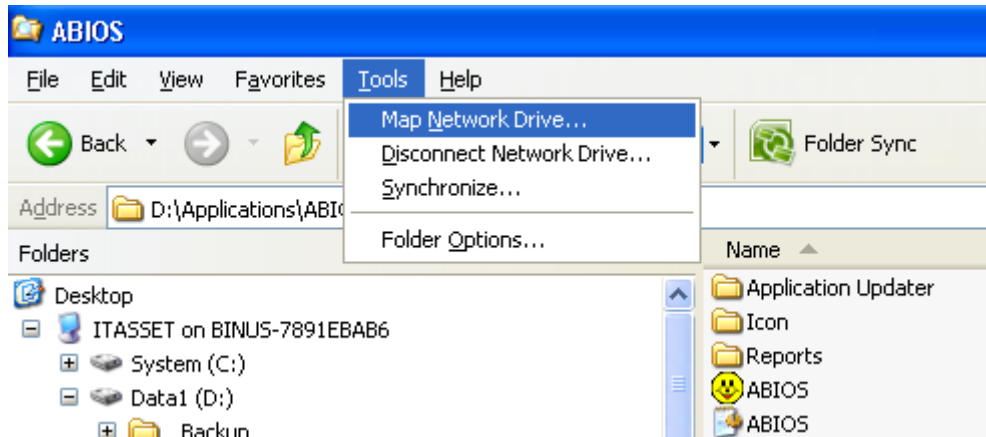
Proses *deploy* disini adalah proses update aplikasi BIOS dengan meng-copy file exe ataupun komponen lain dalam aplikasi BIOS ke komputer lokal user. Proses dapat dilihat pada Gambar 2. Untuk menjembatani antara user dengan direktorat IT, maka dibantu dengan adanya file server yang berisi file aplikasi BIOS untuk versi terbarunya. Saat ini diberi nama: UBINUS dengan folder appscenter. Untuk mengakses folder ini, telah dibuat kesepakatan sebelumnya untuk menggunakan user dan password tertentu. Kesepakatan ini telah diketahui oleh bagian Network, IS Development dan user. Adapun tahapan yang dilakukan user adalah sebagai berikut: (1) membuka window explorer. (2) Lakukan map network driver melalui menu tools—*map network driver*; (3) user mengisi folder, *user name* dan *password* sesuai kesepakatan (user sering lupa); (4) setelah komputer user terhubung dengan server UBINUS, maka user mencari file BIOS di folder yang telah disepakai oleh user dan IS Development; (5) user melakukan update aplikasi dengan meng-copy file-file tersebut ke komputer

lokal (user harus menutup aplikasi berjalan); (6) setelah selesai melakukan update, baru user dapat memakai aplikasi BIOS dengan versi terbaru. Detail proses dapat dilihat pada Gambar 3.



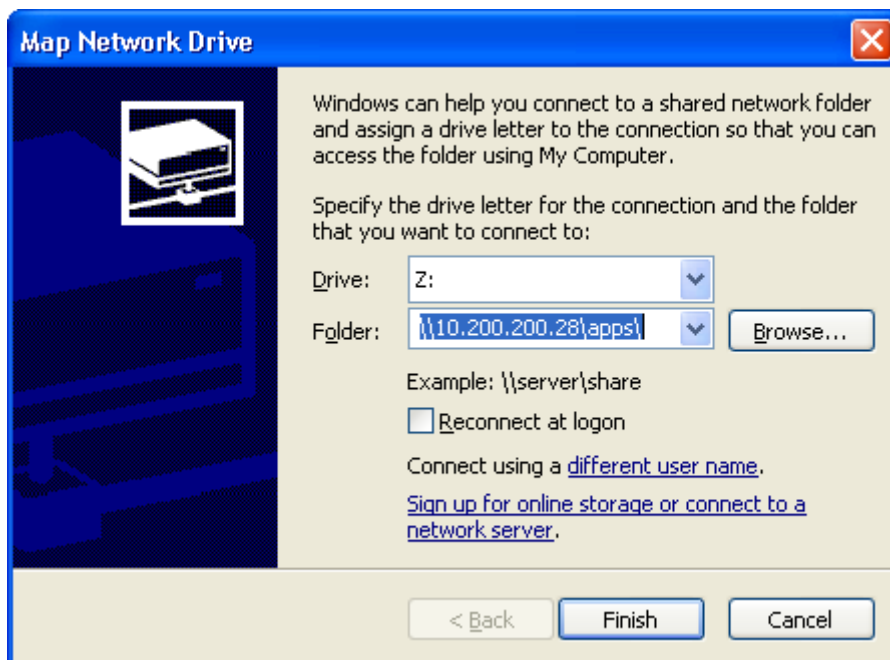
Gambar 3 Proses *detail update* aplikasi (sistem berjalan)

Urutan proses diatas, dirasakan amat mudah oleh *staff IT*, namun tidak demikian oleh mereka *staff operation* yang tidak memiliki *background IT*. Kendala utama yang sering terjadi adalah user lupa akan tahapan, lupa akan alamat folder, lupa user name ataupun password. Sehingga user menemukan kesulitan dan kembali menghubungi *staff IT*. Kendala lainnya adalah, user tidak tau kapan adanya perubahan versi pada aplikasi BIOS, dan tidak mengetahui perubahan apa saja yang terjadi pada versi baru tersebut. Oleh karena itu, untuk mempermudah staf operasional, maka dilakukan perubahan proses update aplikasi dengan penambahan makro pada aplikasi BIOS dan aplikasi update aplikasi.



Gambar 4 Menu Map Network Drive

Gambar 4 adalah menu yang harus dipilih user pada saat melakukan tahap Map Network Driver. User sering lupa akan tahap ini dan lupa berada dimana menu ini.



Gambar 5 Tampilan Map Network Drive

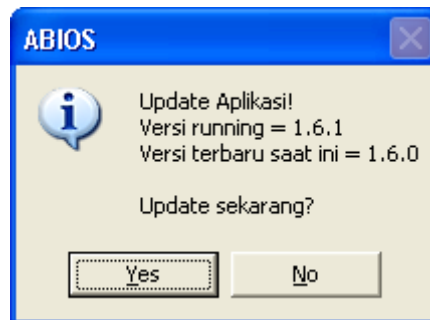
Gambar 5 adalah tampilan Windows setelah user memilih menu *Map Network Driver*, pada tahap ini user harus mengisi folder, dan kembali user sering lupa. Pada Windows ini, user harus mengklik '*different user name*' dan mengisi *username* dan *password*, dan kembali user sering lupa harus melakukan tahap ini, juga lupa *username* dan *password*.

User yang berhasil melakukan *Map Network Driver* maka komputer user akan terhubung dengan server UBINUS. Selanjutnya user harus mencari file ABIOS di folder yang telah disepakati oleh user dan IS Development. Pada tahap ini user juga sering lupa berada dimana folder file ABIOS tersebut. Selanjutnya user melakukan update aplikasi dengan mengcopy file-file tersebut ke komputer lokal (user harus menutup aplikasi ABIOS yang sedang berjalan). Terkadang user lupa dan tidak menutup aplikasi ABIOS, dan akibatnya user tidak bisa copy ke komputer lokal. Setelah selesai melakukan update, baru user dapat memakai aplikasi ABIOS dengan versi terbaru

Mengevaluasi proses update aplikasi diatas, dapat kita lihat banyaknya proses yang harus diingat oleh user. Tidak semua proses tersebut diingat dengan baik oleh user yang bukan berlatar belakang IT. Banyak user yang menemukan kendala dalam update aplikasi dan meminta bantuan staff IT. Terkadang karena kebutuhan yang penting dan mendesak staf operasional sangat terganggu dengan aplikasi ABIOS yang belum *ter-update*.

Analisis Sistem yang Diusulkan

Sistem yang diusulkan dibuat mendekati proses bisnis yang ada sekarang, hal ini dilakukan agar user tidak terganggu dan dipermudah dengan proses update aplikasi. Semua proses update aplikasi tersebut dilakukan oleh sistem secara komputerisasi. Sistem yang diusulkan ini juga sangat mudah dipakai dengan tidak memerlukan beban ingatan user sesuai pedoman IMK.



Gambar 6 Tampilan perbedaan versi ABIOS

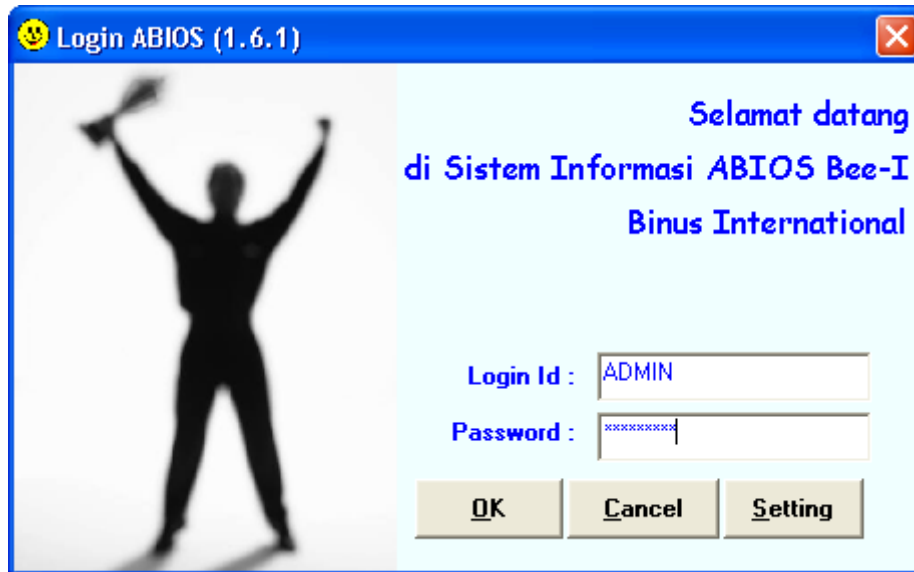
Pada sistem baru, ketika dijalankan, maka sistem akan memeriksa pada database, versi terakhir yang direlease. Kemudian akan dilakukan pemeriksaan versi tersebut pada versi dirinya. Perbandingan dilakukan oleh sistem sehingga user tidak perlu cek secara manual. Jika ditemukan bahwa versi dirinya sama dengan versi terakhir, maka ABIOS akan jalan normal seperti biasa. Namun jika tidak, maka ABIOS akan menginformasikan temuannya tersebut kepada user, tampilan layar dapat dilihat pada Gambar 6. Pada tahap ini user juga ditanyakan apakah user ingin melakukan update aplikasi atau tetap menjalankan ABIOS dengan versi lama.

Pada rancangan database, tabel utama yang dipakai adalah tabel `tabel_versi_aplikasi` dengan detail rancangan tabel dapat dilihat pada tabel 1. Selebihnya, menggunakan tabel yang sudah ada.

Tabel 1 `tabel_versi_aplikasi`

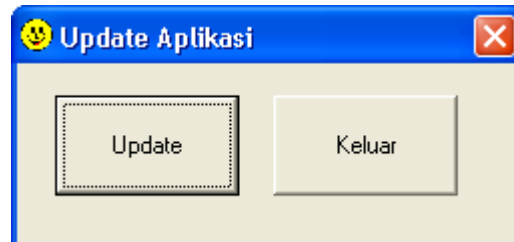
Field	Type Data	Size	Description
<code>stsrc</code>	<code>char</code>	1	Status record
<code>appname</code>	<code>char</code>	30	Nama aplikasi, untuk kasus ini berisi "ABIOS"
<code>versi</code>	<code>varchar</code>	10	Versi aplikasi
<code>keterangan</code>	<code>varchar</code>	300	Keterangan perubahan atau perbaikan versi
<code>TglRelease</code>	<code>datetime</code>	8	Tanggal perubahan versi

Primary Key: `appname`, `versi`



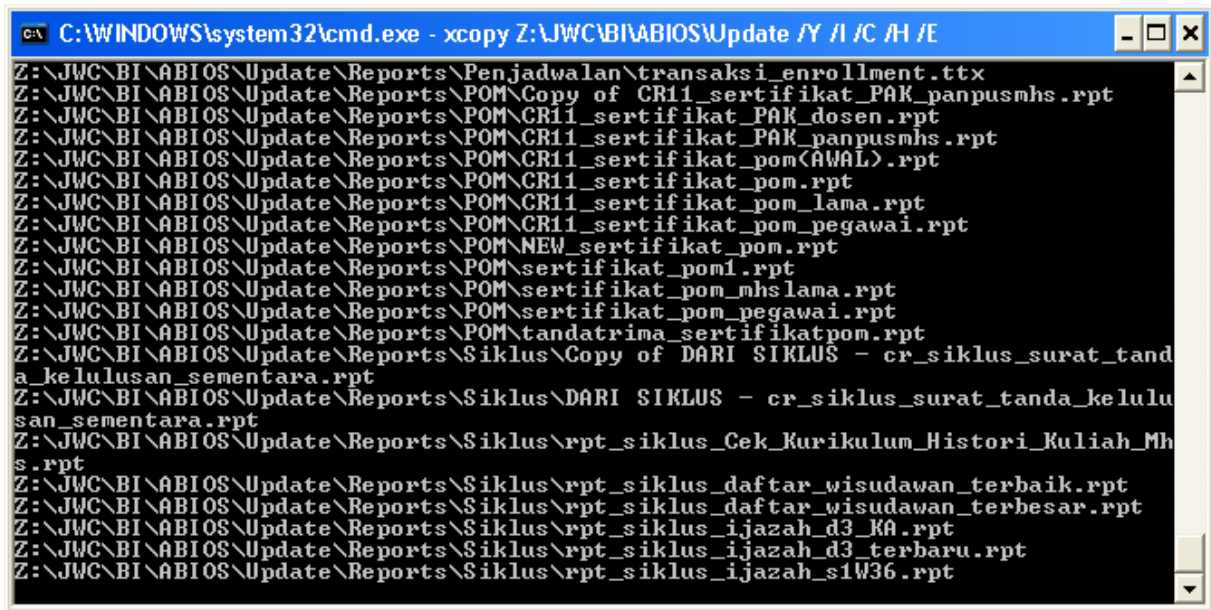
Gambar 7 Tampilan layar login ABIOS

Jika user memutuskan untuk tidak melakukan update aplikasi, dan tetap menjalankan aplikasi ABIOS versi lama, maka user tetap bisa menjalankan aplikasi ABIOS versi lama tersebut. Pada Gambar 6, jika user memilih 'No' maka aplikasi ABIOS akan berjalan seperti biasa dengan menampilkan layar login seperti terlihat pada Gambar 7. Pada layar login ini juga dapat dilihat versi aplikasi ABIOS yang berjalan.



Gambar 8 Tampilan konfirmasi update aplikasi pada ABIOS

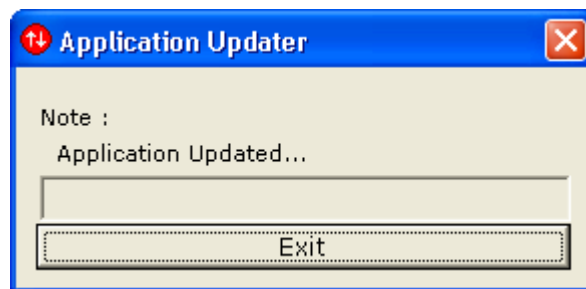
Setelah login, user tetap dapat melakukan update aplikasi dengan menjalankan menu Update Aplikasi. Tampilan layar setelah user memilih menu tersebut dapat dilihat pada Gambar 8. Pada layar ini, jika user benar ingin melakukan update aplikasi, maka user pilih "Update", proses yang jalan sama dengan jika user memilih "Yes" pada Gambar 6. Proses update aplikasi, beberapa proses manual akan dilakukan secara otomatis oleh sistem. Pada Gambar 3, proses Membuka window explorer, Map network driver, mencari file-file ABIOS, dan mengcopy file ke local user dilakukan oleh sistem secara komputerisasi. jika user tidak ingin melakukan update aplikasi, maka user pilih "Keluar" dan sistem tidak jadi di-*update*.



Gambar 9: Tampilan proses update aplikasi pada BIOS

Pada Gambar 8, jika user memilih update, maka sistem akan melakukan proses update aplikasi BIOS. Proses copy dilakukan dengan menggunakan sintak Disk Operating System (DOS) dengan tampilan pada Gambar 9. Sintak DOS yang dipanggil dalam MS Visual Studio VB 6 adalah sebagai berikut:

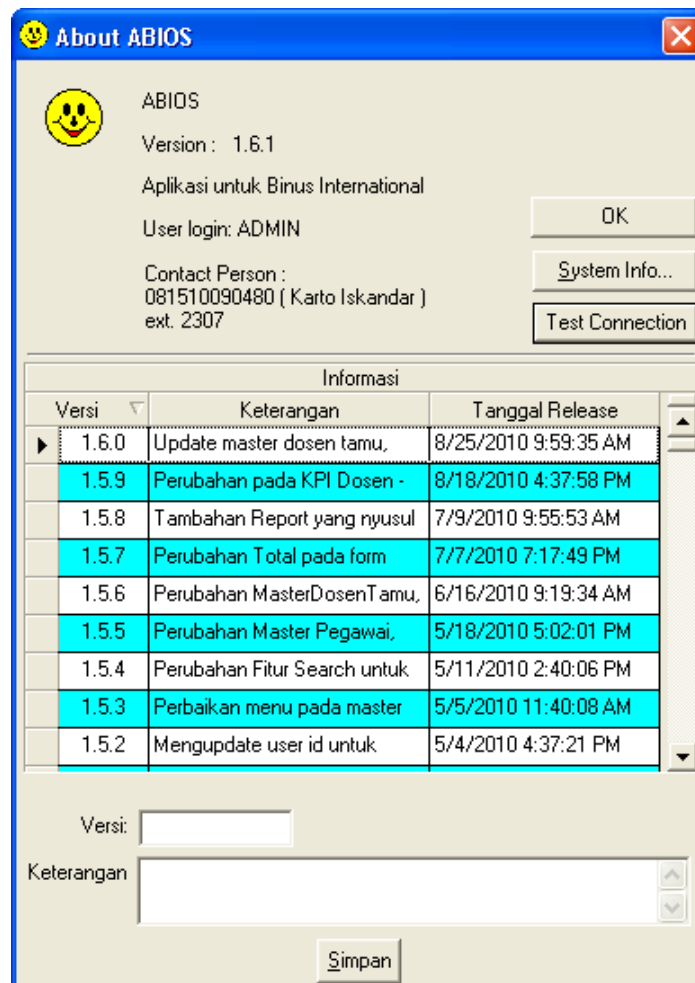
```
Dim wshShell As Object
Dim result As Long
Set wshShell = CreateObject("WScript.shell")
prgProgress.Value = prgProgress.Value + 1
result = wshShell.run("cmd /k cd " & App.Path & " & xcopy " &
    cmbDriveLetter.Text & sSource & " /Y /I /C /H /E & exit", 1, True)
lblStatus.Caption = "Application Updated..."
DoEvents
success = CancelConnection(Left(cmbDriveLetter.Text, 2), 1)
```



Gambar 10 Tampilan informasi update aplikasi pada BIOS berhasil

Setelah proses update aplikasi berhasil, maka layar pada Gambar 9 akan otomatis tertutup. Kemudian diganti dengan tampilan informasi bahwa update aplikasi telah berhasil. Tampilan konfirmasi ini dapat dilihat pada Gambar 10. *Note: Application Updated*, berarti update aplikasi berhasil. Gambar 10 ini juga akan tampil jika proses update aplikasi gagal, dengan note yang berbeda,

diantaranya: *Not Connected check network*, *Error update*: pesan kesalahan, Tidak bisa mapping ke UBINUS. Untuk keluar dari Sistem Otomatis Update ini, user pilih “Exit”, maka layar ini akan tertutup. Selanjutnya user dapat menjalankan aplikasi ABIOS dengan versi terbaru.



Gambar 11 Tampilan informasi histori perubahan versi ABIOS

User dapat mengetahui tambahan fitur baru atau perbaikan *error* apa yang terdapat pada versi ABIOS terbaru. Setelah user berhasil masuk ke aplikasi ABIOS, untuk mengetahui informasi perubahan versi dan perbaikan apa saja, maka user dapat memilih menu About ABIOS. Tampilan layar About ABIOS ini dapat dilihat pada Gambar 11. Pada layar ini, user dapat mengetahui versi yang berjalan, *contact person* yang dapat dihubungi jika perlu bantuan pada aplikasi ABIOS, dan histori perubahan versi aplikasi ABIOS lengkap dengan informasi perubahan dan tanggal jam updatenya. Untuk versi-versi tertentu yang pada pengembangannya dibutuhkan komponen baru, update aplikasi ini tidak dapat berjalan, hal ini disebabkan perlunya di install ulang aplikasi ABIOS pada komputer user.

Implementasi dan Evaluasi

Sistem Otomatis *Update* pada aplikasi ABIOS ini telah berhasil dibuat dan telah lolos tahap testing (alpha dan beta) . Sistem ini telah diimplementasikan pada beberapa aplikasi di Binus, salah satunya adalah aplikasi ABIOS. User yang menggunakan ABIOS tidak merasa terganggu dengan

adanya sistem otomatis update ini, malah terbantu dan dapat memotong banyak proses manual yang mengganggu dalam proses update aplikasi.

Sistem Otomatis Update ini tidak dapat berjalan jika sistem jaringan dari kampus JWC ke kampus Syahdan bermasalah. Hal ini disebabkan Sistem Otomatis Update ini melakukan proses2 yang menggunakan resource jaringan, diantaranya mapping network dan copy file melalui jaringan. Hal lainnya adalah server UBINUS berlokasi di kampus Syahdan. Disamping masalah network, untuk *performance* selama implementasi tidak ditemukan masalah karena pada topologi network JWC—Syahdan sangat baik dengan pita lebar dan aturan topologi protokol yang baik.

SIMPULAN

Adapun simpulan yang diperoleh dari pembuatan sistem otomatis update ini adalah sebagai berikut. Menginformasikan kepada user, bahwa ada update aplikasi versi terbaru pada saat user menjalankan aplikasi desktop. Mempermudah user dalam mengupdate aplikasi desktop yang terinstal di PC mereka. Kelebihan dari sistem otomatis update ini adalah user dapat dengan cepat mengetahui informasi adanya versi terbaru dari aplikasi yang dipakai. User dapat mengetahui detail informasi mengenai tambahan fitur ataupun perbaikan modul pada versi terbaru, beserta histori perubahan tiap-tiap versinya. User dapat mengupdate aplikasi desktop versi terbaru dari *source* server dengan mudah dan cepat. Server *source* aman, karena hak akses ke server tersebut hanya melalui aplikasi, tidak secara manual. Adapun saran yang dapat diberikan untuk pengembangan lebih lanjut dari sistem otomatis update adalah sebagai berikut. Sistem ini telah dibuat secara *independent*, dalam arti berdiri sendiri, sehingga dapat dipakai dalam update aplikasi desktop manapun yang berjalan dalam lingkungan Operating System keluarga Windows. Untuk kedepannya, dapat dikembangkan sistem otomatis update untuk multi platform seperti Linux ataupun Macintosh dan pada aplikasi *mobile*.

DAFTAR PUSTAKA

- Choldun, I. (2006). *Perancangan Sistem Informasi Akademik dengan Mengimplementasikan ERP*. Prosiding Konferensi Nasional Teknologi Informasi & Komunikasi untuk Indonesia. Institut Teknologi Bandung.
- Pressman, R. S. (2001). *Software Engineering A Practitioner's Approach* (5th ed.). Singapore: McGraw Hill.
- Satoto, K. I. (2009). *Analisis Keamanan Sistem Informasi Akademik Berbasis Web Di Fakultas Teknik Universitas Diponegoro*. Seminar Nasional Aplikasi Sains dan Teknologi. ISSN: 1979-911X: 175 – 186.
- Saputri, A. W. (2010). *Analisis Total Quality Management dalam Proses Perencanaan Strategis Sistem Informasi*. Seminar Nasional Aplikasi Teknologi Informasi. ISSN: 1907-5022: B-44.
- Shneiderman, B. (2004). *Designing the User Interface: Strategies for Effective Human Computer Interaction* (4th ed.). California: Addison Wesley Longman Publishing.