

PENERAPAN SISTEM PERSONALISASI AKUN PADA ADMISI ONLINE BINUS UNIVERSITY

Karto Iskandar; Jefriyanto; Halena Giovanni; Candra Gunawan

Computer Science Department, School of Computer Science, Binus University
Jl. K.H. Syahdan No. 9, Palmerah, Jakarta Barat 11480
karto_i@binus.edu; jefri.08@gmail.com; halquelin@gmail.com; hell_buster90@yahoo.com

ABSTRACT

This research purpose is to make easy the registration process of new students by personalizing the account page on admission online. Therefore, each user can manage their own needs and recognize the steps that must be taken to be a BINUS student. Analyses are performed towards the current working system and issues related to the system. Further is performing finding references related to the problem through literature study. Then, creating a UML diagram of the system design, user interface and database. Next is the programming phase and evaluation of the application . The results obtained is an online admission application with a personalized login page. It can be used to make easy the registration of new student or related issue at the many institutions of Binus. It is suggested to create the mobile version of online admission for the future development.

Keywords: *sytem information development, account personalisation, online admission*

ABSTRAK

Penelitian ini bertujuan untuk mempermudah proses pendaftaran mahasiswa baru dengan cara personalisasi akun pendaftar di admisi online. Maka dari itu, pengguna dapat mengatur tampilan sendiri sesuai kebutuhan dan mengetahui langkah-langkah yang harus ditempuh untuk menjadi mahasiswa BINUS. Metode penelitian dilakukan dengan cara analisis terhadap cara kerja dan wawancara pada permasalahan sistem yang berjalan. Selanjutnya melakukan studi pustaka untuk mencari referensi yang berhubungan dengan permasalahan tersebut. Kemudian melakukan perancangan sistem berupa diagram UML, user interface dan database. Setelah itu adalah tahap pemrograman dan melakukan uji coba terhadap aplikasi yang selesai dibuat. Hasil yang dicapai berupa aplikasi admisi online dengan halaman login dari yang dipersonalisasi. Hasil dari perancangan ini dimanfaatkan untuk kemudahan pendaftaran mahasiswa baru atau setingkat di berbagai instansi BINUS. Kesimpulan dari penelitian ini adalah tampilan halaman yang dapat dipersonalisasi akan mempermudah pendaftar dalam proses pendaftaran mahasiswa baru pada Admisi Online BINUS University. Saran untuk pengembangan ke depannya, pendaftaran admisi online dapat dilakukan dalam versi mobile.

Kata kunci: *pengembangan sistem informasi, personalisasi akun, admisi online, BINUS*

PENDAHULUAN

Perkembangan Teknologi Informasi pada saat ini semakin berkembang dengan pesat. Hal ini ditandai dengan adanya peningkatan penggunaan internet dari tahun ke tahun. Dengan adanya internet, penyebaran informasi saat ini terjadi dengan sangat cepat. Sekarang ini, banyak Perguruan Tinggi yang telah menggunakan internet misalnya dalam hal pendaftaran calon mahasiswa baru.

Salah satu Perguruan Tinggi yang mengimplementasikan internet pada pendaftaran calon mahasiswa baru adalah BINUS University. BINUS University menyadari pentingnya proses marketing dan admisi sebagai ujung tombak dalam persaingan pada era globalisasi ini guna menuju visinya sebagai *world class Univerisity*. Di dalam BINUS University sistem pendaftaran *online* ini dapat dilakukan pada *website* admisi *online* dengan alamat <http://www.binus.ac.id/daftaronline> dan diatur oleh staff admisi BINUS melalui aplikasi *Back end*.

Permasalahan yang sering muncul pada aplikasi web internet adalah tidak adanya bantuan ketika pengguna tidak tahu harus melakukan apa terhadap sistem aplikasi. Pengguna sering bingung dan tidak tahu sekarang mereka sudah sampai pada phase mana dari keseluruhan phase yang ada untuk mendaftar sebagai mahasiswa BINUS. Untuk proses pendaftaran *online* pada BINUS University terdapat banyak proses-proses penting yang harus diketahui oleh pendaftar, seperti: informasi program yang dibuka, informasi harga yang harus dibayar, informasi kapan pendaftar harus menghadiri test ujian masuk, informasi hasil test, pendaftaran ulang sampai informasi kapan mulai kuliah.

Peneliti mengembangkan admisi *online* melalui inovasi sesuai dengan perkembangan teknologi saat ini. Inovasi yang dilakukan yaitu dengan mengusulkan implementasi personalisasi halaman setelah *login*. Personalisasi menurut Bonett (2001) adalah proses pengumpulan informasi dari *user* selama terjadinya interaksi, yang kemudian digunakan untuk memberikan konten dan layanan khusus sesuai dengan kebutuhan user. Tujuannya adalah untuk meningkatkan *user experience* dalam hal pelayanan.

Personalisasi dilakukan dengan memanfaatkan tools *droptings*. Menurut Zabir (2007, p4), *droptings* adalah sebuah web portal yang mengizinkan pengguna untuk mengontrol apa yang mereka ingin letakkan pada halaman *website* mereka. Dengan demikian, informasi yang ditampilkan benar-benar relevan dengan pendaftar. *Droptings* dapat mengambil konten dari berbagai sumber yang berbeda seperti layanan penyedia foto online dari Flickr, berita dari CNN, laporan cuaca dari Weather.com, dan lain sebagainya. Intinya, *droptings* menciptakan interaksi pada *website* ke tahapan selanjutnya dengan menggunakan teknologi Ajax.

Selain itu, terdapat juga perubahan dimana tampilan dan fungsi setiap informasi lebih dipersonalisasi. Perubahan tersebut menggunakan tampilan *drag and drop*. Komponen yang dirancang menggunakan gaya "*drag and drop*" memiliki banyak keuntungan termasuk peningkatan efisiensi dalam hal eksekusi dan penggunaan kembali. Selama beberapa tahun, *drag and drop* telah beranjak dari fitur yang dingin menjadi sebuah potongan yang dibutuhkan dari hampir semua antarmuka pengguna (Jalender, 2010). Tampilan *drag and drop* terdiri atas kumpulan *widget-widget*. Sebuah *widget* menurut Zabir (2007, p2) adalah potongan elemen yang berbeda dan tidak saling berhubungan pada sebuah halaman *website* untuk menampilkan fungsi tertentu, *user interface* sendiri, dan terdiri dari fitur-fitur. Contoh *widget* adalah *to-do lists*, buku alamat, daftar kontak, RSS feeds, jam, kalender, playlists, daftar perubahan harga saham, laporan cuaca, laporan lalu lintas, kamus, permainan, atau hampir semua hal yang dapat dibayangkan yang dapat dipaketkan dan ditampilkan ke dalam *website*. Pada lingkungan korporasim, *widget* dapat berhubungan dengan sistem internal. Sebagai contoh sebuah *widget expense tracker* dapat berinteraksi secara langsung dengan sistem

akunting internal. Dengan adanya personalisasi, pendaftar dapat mengatur halaman sesuai dengan keinginan mereka sehingga informasi yang akan ditampilkan lebih informatif. Selain dapat mengatur halaman sendiri, di dalam halaman personalisasi terdapat langkah-langkah yang harus dilakukan oleh calon pendaftar untuk mempermudah proses menjadi calon mahasiswa BINUS.

METODE

Penelitian akan dilakukan dengan menggunakan dua metodologi utama. Pertama adalah analisis, dilakukan dengan tahapan sebagai berikut: (1) survei atau observasi atas sistem yang sedang berjalan, dilakukan dengan meminta informasi dan data mengenai kondisi, cara kerja dan permasalahan sistem pada pihak IT Directorate Binus yang menangani langsung kebutuhan admisi *online*; (2) wawancara kepada pihak IT Directorate divisi marketing mengenai detail kondisi, cara kerja, dan permasalahan sistem pada pihak IT Directorate Binus yang menangani langsung kebutuhan admisi *online*; (3) studi pustaka melalui buku-buku dan jurnal ilmiah yang berkaitan dengan topik yang dibahas.

Metode kedua adalah perancangan, meliputi beberapa proses. Pertama adalah pembuatan model proses. Metode yang digunakan untuk sistem admisi *online* ini adalah model *waterfall*. Hal ini dikarenakan tahap demi tahap yang dilalui harus menunggu selesainya tahap sebelumnya dan berjalan berurutan. Model proses terdiri dari tahapan sebagai berikut: (1) *system information engineering and modeling* – kebutuhan dari keseluruhan sistem admisi *online* didapatkan melalui wawancara kepada pihak IT Directorate Binus; (2) *software requirement analysis* – kebutuhan dari admisi *online* difokuskan pada penambahan instansi Binus yang baru, proses pengisian formulir calon mahasiswa baru dan implementasi personalisasi pada halaman setelah *login*; (3) *design* – pada proses ini dilakukan perancangan *database* yang mengacu pada Connolly dan Begg (2004) dan juga perancangan *user interface* untuk memenuhi kebutuhan sistem yang akan dibuat; (4) *coding* – bahasa pemrograman yang digunakan pada sistem admisi *online* adalah ASP .NET C# 3.5 dengan DBMS Microsoft SQL Server 2005. Untuk personalisasi, tools yang digunakan adalah tools Droptings; (5) *testing verification* – pada tahap testing/ verification, langkah pertama yang dilakukan berupa membuat sistem registrasi instansi Binus yang baru dengan sistem CMS. Lalu mengisi formulir yang sudah dipisahkan berdasarkan prioritas, baik sebelum *login* ataupun setelah *login*. Terakhir, melakukan testing halaman yang telah mengimplementasikan personalisasi.

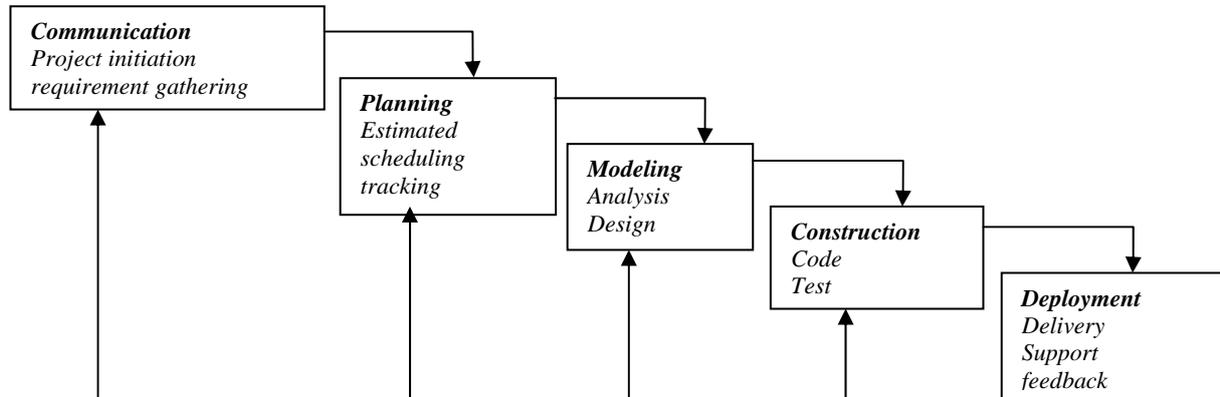
Proses selanjutnya adalah notasi perancangan. Notasi yang digunakan adalah UML 2.0 yang mengacu pada standar perancangan Roger S. Pressman.

Software Development

System Development Life Cycle (SDLC) dalam rekayasa sistem dan rekayasa perangkat lunak, adalah proses pembuatan dan perubahan sistem serta model dan metodologi yang digunakan untuk mengembangkan sistem-sistem tersebut. Metode Perancangan SDLC yang digunakan dalam penulisan ini adalah Model Proses *Waterfall*. Model *waterfall* (Gambar 1) adalah proses pengembangan software sekuensial, dimana kemajuan dipandang sebagai terus mengalir ke bawah (seperti air terjun) melalui tahapan konsepsi, inisiasi, analisis, desain, konstruksi, pengujian dan pemeliharaan (Pressman, 2006, p79).

Communication – permodelan ini diawali dengan komunikasi dan kolaborasi dengan konsumen (*stakeholders*) untuk mencari kebutuhan dari keseluruhan sistem yang akan diaplikasikan ke dalam bentuk *software*. Hal ini sangat penting, mengingat *software* harus dapat berinteraksi dengan

elemen-elemen yang lain seperti *hardware*, *database*, dsb. Tahap ini sering disebut dengan *Project Definition*.



Gambar 1 Waterfall model

Planning – proses ini menetapkan rencana untuk pengerjaan *software* yang meliputi: tugas-tugas teknis yang akan dilakukan, resiko yang mungkin terjadi, sumber-sumber yang dibutuhkan, hasil yang akan dibuat, dan jadwal pengerjaan.

Modeling – proses ini meliputi pembuatan model yang memungkinkan pengembang dan konsumen untuk lebih memahami kebutuhan perangkat lunak dan desain yang akan mencapai kebutuhan tersebut.

Construction – proses ini merupakan proses gabungan dari *coding* dan *testing*. Untuk dapat dimengerti oleh mesin computer, maka desain tadi harus diubah bentuknya menjadi bentuk yang dapat dimengerti oleh mesin, yaitu ke dalam bahasa pemrograman melalui proses *coding*. Tahap ini merupakan implementasi dari tahap design yang secara teknis nantinya dikerjakan oleh *programmer*. Sedangkan *testing* adalah sesuatu yang dibuat haruslah diujicobakan. Demikian juga dengan *software*. Semua fungsi-fungsi *software* harus diujicobakan, agar *software* bebas dari error, dan hasilnya harus benar-benar sesuai dengan kebutuhan yang sudah didefinisikan sebelumnya.

Deployment – pemeliharaan suatu *software* diperlukan, termasuk di dalamnya adalah pengembangan, karena *software* yang dibuat tidak selamanya hanya seperti itu. Ketika dijalankan mungkin saja masih ada error kecil yang tidak ditemukan sebelumnya, atau ada penambahan fitur-fitur yang belum ada pada *software* tersebut. Pengembangan diperlukan ketika adanya perubahan dari eksternal perusahaan seperti ketika ada pergantian sistem operasi, atau perangkat lainnya.

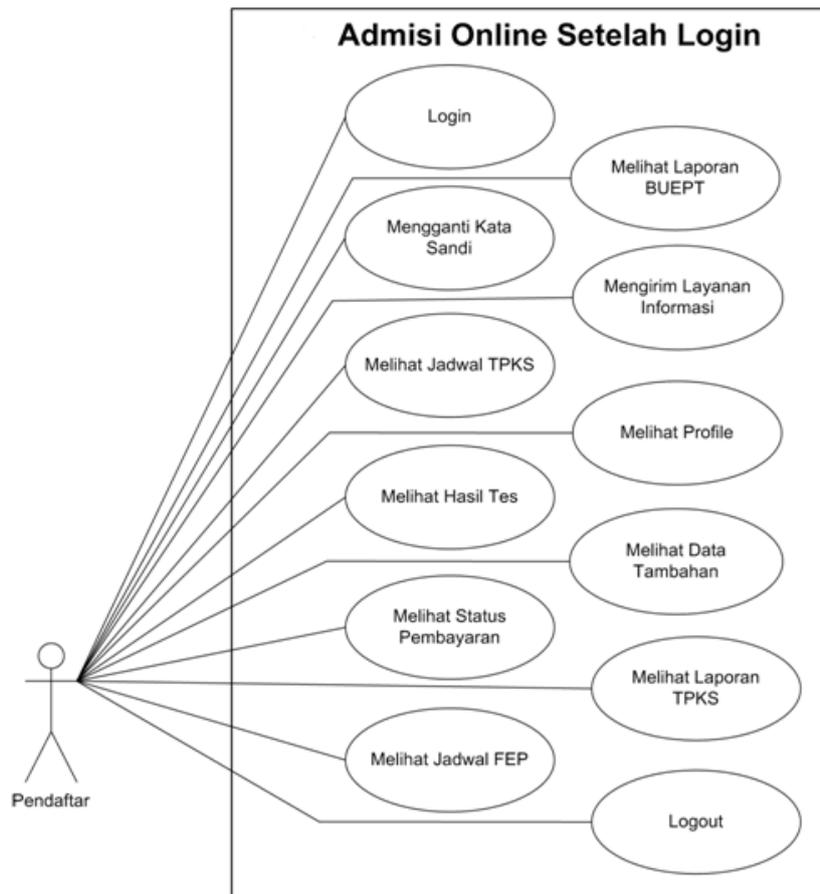
HASIL DAN PEMBAHASAN

Admisi *online* Binus University yang merupakan wadah pendaftaran *online* bagi calon mahasiswa Binus saat ini telah memiliki bagian *front end* dan *back end*. Bagian *front end* menampilkan beragam informasi yang dibutuhkan oleh calon pendaftar serta proses pendaftaran *online* itu sendiri. Bagian *back end* digunakan untuk mengatur data pendaftaran.

Saat ini pendaftar dapat masuk pada halaman setelah *login* apabila telah melakukan pembayaran formulir. Setelah *login*, beragam informasi dapat diperoleh pendaftar, yaitu: Change Password, Jadwal TKPS, View Result, Payment Status, Pre University, Print Briefing Card,

Information Service, View Profile, Request Change Data, Additional Data, TKPS Report, BUEPT Report.

Pada *use case* diagram (Gambar 2) admisi *online* setelah *login*, yang bertindak sebagai aktor adalah pendaftar. Pendaftar yang *login* dapat melihat laporan BUEPT, mengganti kata sandi, mengirim layanan informasi, melihat jadwal TPKS, melihat profile, melihat hasil tes, melihat data tambahan, melihat status pembayaran, melihat laporan TPKS, melihat jadwal FEP, dan melakukan *logout*.



Gambar 2 Use case diagram admisi *online* setelah *login* (sistem berjalan)

Pada halaman setelah *login*, terdapat fitur *timeline registration* yang memberikan informasi terbaru mengenai tahapan pendaftar untuk menjadi mahasiswa Binus. Tahapan tersebut terdiri dari: (1) *online registration* – berisi tanggal pengisian formulir *online* dari pendaftar; (2) *entrance test* – jadwal untuk mengikuti ujian. *Widget* yang digunakan adalah *Entrance Test Schedule*; (3) *first payment* – jadwal pembayaran biaya kuliah. *Widget* yang berhubungan adalah *Payment Status*; (4) *reregistration* – jadwal pengenalan seputar jurusan yang diambil. *Widget* yang berhubungan adalah *View Test Result*; (5) *briefing schedule* – jadwal pengenalan seputar jurusan yang diambil. *Widget* yang berhubungan adalah *FEP Schedule*.

Pada *use case* diagram (Gambar 3), yang bertindak sebagai aktor adalah pendaftar. Aktor dapat menampilkan *widget information service*, mengisi biodata ayah, menampilkan *widget parental information father*, mengisi biodata ibu, menampilkan *widget parental information guardian*, mengisi biodata wali, menampilkan *widget parental information mother*, melihat *timeline registration*, menampilkan *widget payment status*, menampilkan *widget BUEPT report*, menampilkan *widget TPKS*

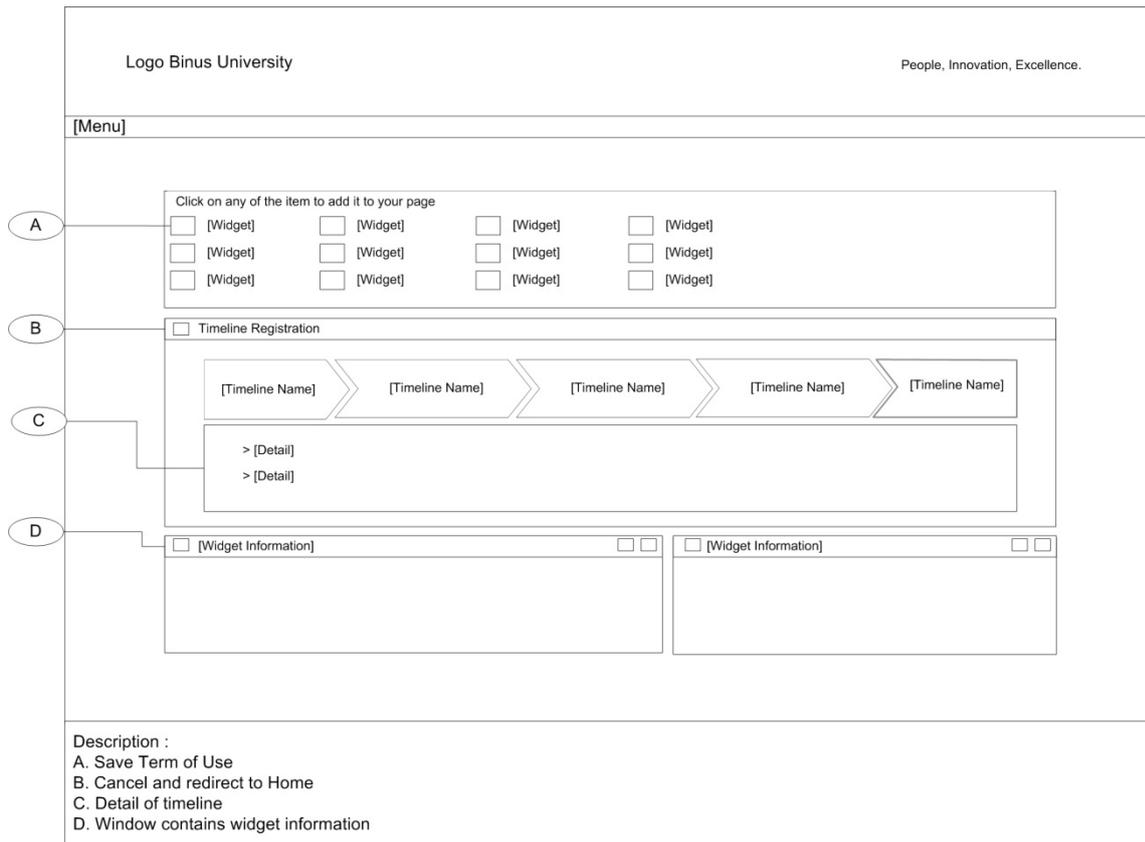
report, menampilkan *widget change password*, menampilkan *widget view profile*, menampilkan *widget entrance test schedule*, menampilkan *widget view test result*, menampilkan *widget FEP schedule*.



Gambar 3 Use case diagram admisi *online* setelah *login* (Sistem baru)

Rancangan Tampilan Halaman Setelah Login

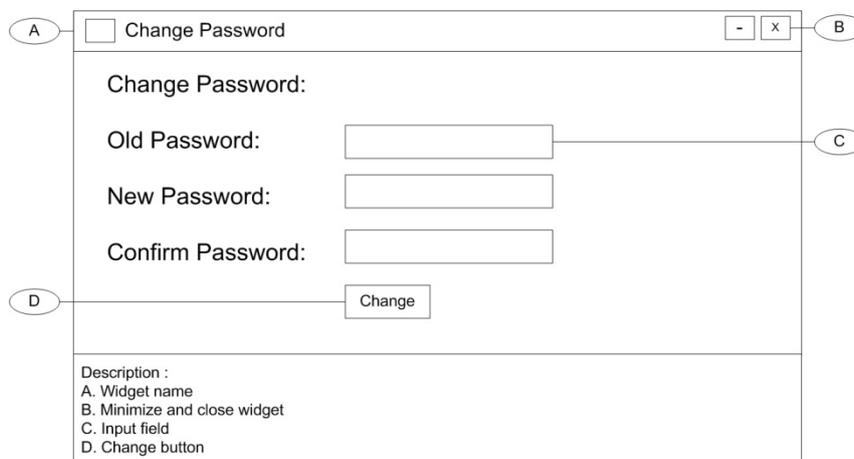
Rancangan tampilan halaman setelah *login* secara keseluruhan terdiri dari pemilihan *widget* dari daftar *widget* yang tersedia. Kemudian di bawahnya terdapat fitur *timeline registration* yang relevan dengan langkah yang harus ditempuh oleh pendaftar beserta perinciannya. Di bawah dari *timeline registration* terdapat susunan *widget* yang letaknya dapat dipersonalisasikan oleh pendaftar. Gambar rancangan tampilan halaman setelah *login* dapat dilihat pada Gambar 4.



Gambar 4 Rancangan tampilan halaman setelah login

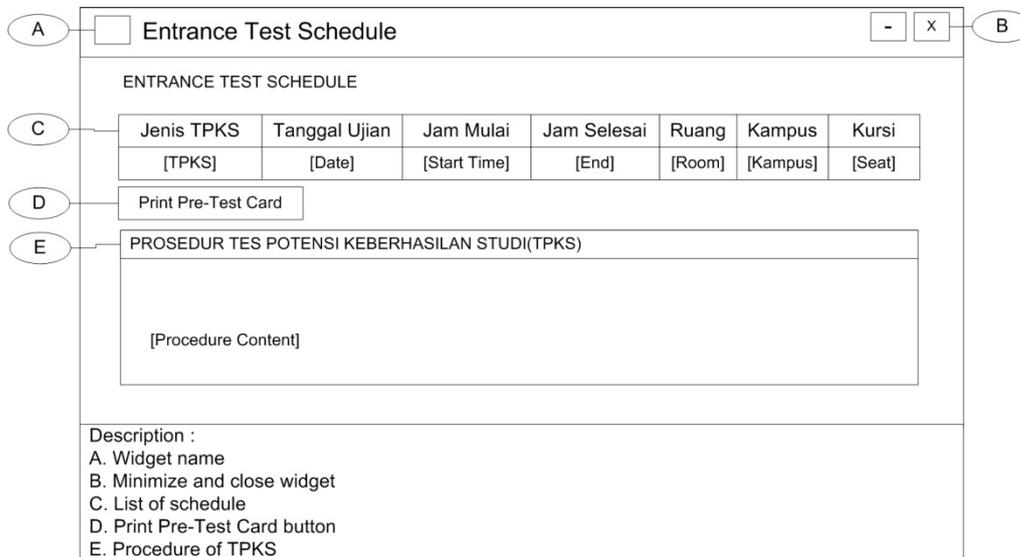
Setelah merancang tampilan setelah login secara keseluruhan, penulis membuat rancangan dari beberapa widget yang dibutuhkan, di antaranya: *Widget Change Password*, *Widget Entrance Test Schedule*, *Widget Payment Status* dan *Widget View Test Result*.

Widget Change Password berisi input untuk memasukkan password lama, password baru, dan konfirmasi password baru serta tombol change untuk memproses pengubahan password. Gambar rancangan widget Change Password dapat dilihat pada Gambar 5.



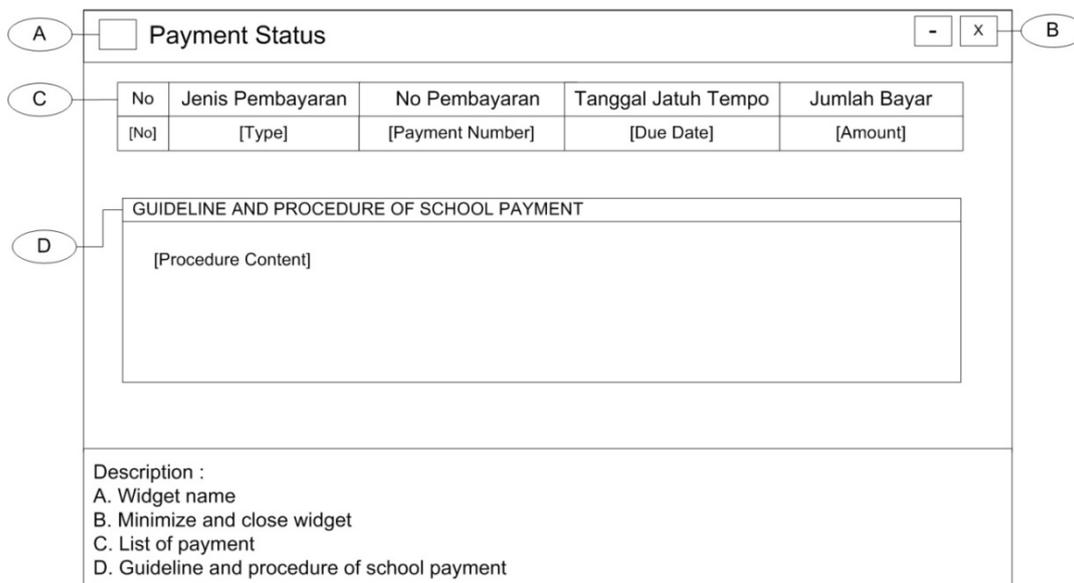
Gambar 5 Change Password Widget

Widget Entrance Test Schedule berisi informasi jadwal tes ujian masuk calon pendaftar. Informasi yang dimaksud berupa jenis TPKS, tanggal ujian, jam mulai, jam selesai, ruang, kampus, kursi, dan prosedur tes. Pendaftar dapat mencetak kartu tes masuk menggunakan tombol *print pre-test card*. Gambar rancangan *Widget Entrance Test Schedule* dapat dilihat pada Gambar 6.



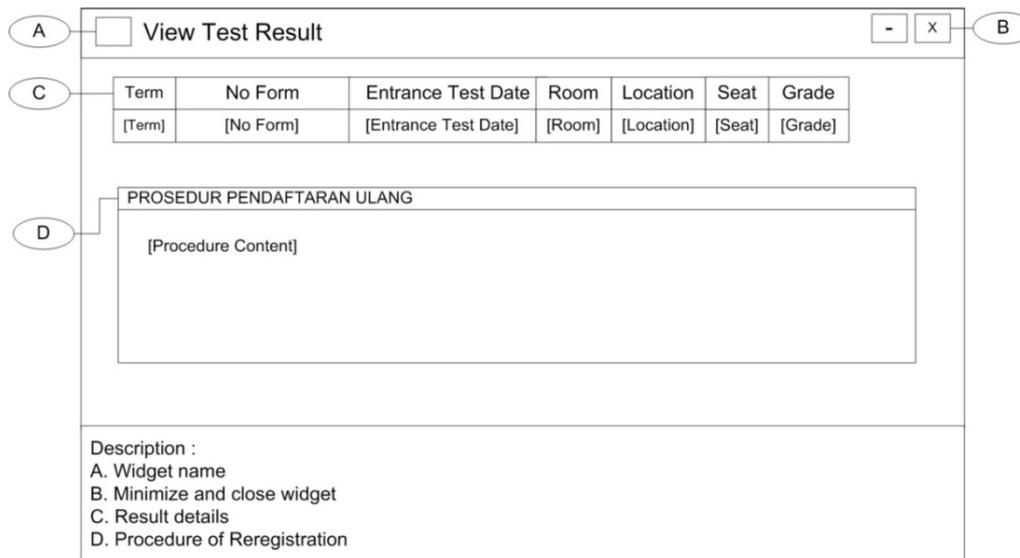
Gambar 6 *Entrance Test Schedule Widget*

Widget Payment Status berisi informasi pembayaran dari pendaftar. Terdiri dari jenis pembayaran, nomor pembayaran, tanggal jatuh tempo, jumlah bayar, dan petunjuk pembayaran. Gambar rancangan *Widget Payment Status* dapat dilihat pada Gambar 7.



Gambar 7 *Payment Status Widget*

Widget View Test Result berisi hasil test yang dilakukan pendaftar. Terdiri dari gelombang pendaftaran, nomor formulir, tanggal tes, nomor ruangan, lokasi, nomor kursi, dan hasil tes. Kemudian terdapat kotak informasi pada prosedur pendaftaran ulang. Gambar rancangan *Widget View Test Result* dapat dilihat pada Gambar 8.



Gambar 8 *View Test Result Widget*.

Implementasi

Halaman setelah *login* terdapat fitur *timeline registration* yang bersifat personalisasi, dimana fitur tersebut menampilkan informasi langkah-langkah yang harus ditempuh oleh pendaftar sampai menjadi mahasiswa Binus. Informasi yang disediakan lebih relevan seperti tanggal dalam *timeline registration* yang sesuai dengan pendaftar. Letak dari konten *widget* juga dapat dipersonalisasi sesuai dengan keinginan pendaftar. Gambar tampilan halaman setelah *login* dapat dilihat pada Gambar 9. Berikut adalah tampilan implementasi *Widget Change Password* (Gambar 10), *Widget Entrance Test Schedule* (Gambar 11), *Widget Payment Status* (Gambar 12), dan *Widget View Result Test* (Gambar 13).

PENUTUP

Berdasarkan hasil penelitian yang dilakukan pada admisi *online* Binus University, dapat disimpulkan bahwa aplikasi admisi *online* setelah *login* memberikan informasi yang benar-benar relevan/ sesuai dengan langkah-langkah pendaftaran yang sedang ditempuh oleh pendaftar. Selain itu, halaman yang ditampilkan lebih informatif dengan adanya personalisasi yang dilakukan oleh pendaftar.

DAFTAR PUSTAKA

Bonett, M. (2001). *Personalization of Web Services: Opportunities and Challenges*. Diakses 5 Januari 2012 dari <http://www.ariadne.ac.uk/issue28/personalization>.

Connolly, T. M., & Carolyn E. B. (2004). *Database Systems: A Practical Approach to Design, Implementation, and Management* (4th edition). Harlow: Addison Wesley.

Jalender, B. (2010). Drag and drop: influences on the design of reusable software components. *International Journal on Computer Science and Engineering (IJCSE)*, 2 (07), 2386 – 2393.

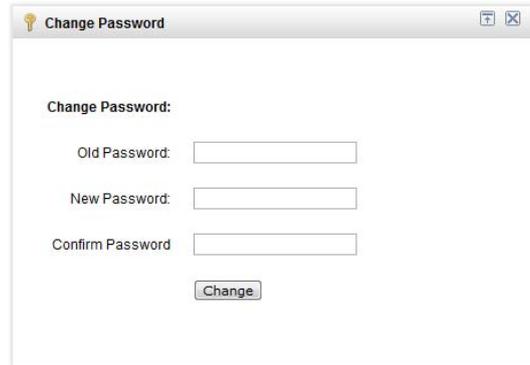
Pressman, R. S. (2006). *Software Engineering a Practitioner's Approach* (6th edition). New York: McGraw-Hill.

Zabir, O. A. (2007). *Building a Web 2.0 Portal with ASP.NET 3.5* (1st edition). Sebastopol: O'Reilly Media.

APPENDIX

The screenshot displays the user interface of the BINUS UNIVERSITY portal. At the top, the header includes the BINUS UNIVERSITY logo and the tagline "People. Innovation. Excellence." Below the header is a navigation menu with links for HOME, GUIDELINE, CONTACT US, FAQ, MY ACCOUNT, and LOGOUT. A central section titled "Click on any of the item to add it to your page" contains several interactive buttons: Parental Information Father, Change Password, View Profile, Information Service, FEP Schedule, Parental Information Mother, View Test Result, Entrance Test Schedule, BUEPT Report, and TPKS Report, Parental Information Guardian, and Payment Status. Below this is a "Timeline Registration" section showing a progress bar with five stages: Online Registration (07 Sep 2009), Entrance Test (29 Nov 2008), First Payment (15 Apr 2010), Reregistration (15 May 2012), and Briefing Schedule (23 Jul 2007). A message box below the timeline states: "Thank you for choosing Binus. You must complete Father Information, Mother Information, and Guardian Information with the available widget." At the bottom, there are two open windows. The left window, titled "Parental Information Father", shows the "DETAIL OF PARENTAL INFORMATION FATHER" form with a pink error message: "Please complete this detail below." The form includes fields for "Kedaaan ayah" (set to "Masih Hidup"), "Father Condition", and "Nama Ayah". The right window, titled "Information Service", contains an "Ask New Question" form with fields for "Subject", "Category" (set to "Biaya"), and "Question".

Gambar 9 Tampilan implementasi halaman setelah login



Gambar 10 Tampilan implementasi *Widget Change Password*



Gambar 11 Tampilan implementasi *Widget Entrance Test Schedule*



Gambar 12 Tampilan impementasi *Widget Payment Status*



Gambar 13 Tampilan implementasi *Widget View Test Result*