

APLIKASI *PRELIMINARY TEST COURSE* DAN *SMART STORAGE* DALAM MENUNJANG KEGIATAN PRAKTIKUM DI UNIVERSITAS GUNADARMA

Raditya Fajar¹⁾, Tri Handhika²⁾

¹Jurusan S1 Teknik Informatika, Fakultas Teknologi Industri, Universitas Gunadarma,

²Pusat Studi Komputasi Matematika, Universitas Gunadarma

Jl. Margonda Raya 100, Depok, 16424

Telp : 021-78881112

E-mail : radityafajar@student.gunadarma.ac.id¹⁾

E-mail : trihandika@staff.gunadarma.ac.id²⁾

Abstrak

Laboratorium Manajemen Menengah Universitas Gunadarma merupakan salah satu laboratorium yang memanfaatkan komputer sebagai media untuk mendukung kegiatan praktikum. Namun demikian, sistem penilaian yang diberlakukan selama ini belum mencerminkan performa mahasiswa secara obyektif. Selain itu, sistem penilaian secara manual pun menghambat kinerja asisten dalam pengolahan nilai praktikum tersebut. Dalam tulisan ini, penulis merancang aplikasi Preliminary Test Course dan Smart Storage berbasis desktop dengan metode Software Development Life Cycle (SDLC) model Waterfall menggunakan Java dan Microsoft SQL Server 2000. Aplikasi tersebut untuk meningkatkan akurasi penilaian serta mempercepat pengolahan nilai praktikum di Laboratorium Manajemen Menengah Universitas Gunadarma sehingga dapat berjalan dengan lebih efisien. Aplikasi Preliminary Test Course berfungsi untuk mendapatkan nilai tes pendahuluan bagi mahasiswa yang mengikuti praktikum di Laboratorium Manajemen Menengah Universitas Gunadarma. Sedangkan, aplikasi Smart Storage digunakan sebagai antar muka dalam mengolah nilai praktikum tersebut. Pada akhir semester, nilai praktikum yang telah disimpan dan diolah dapat diubah ke dalam bentuk dokumen Microsoft Excel.

Kata kunci: Aplikasi Preliminary Test Course, Aplikasi Smart Storage, Java, SQL Server 2000, SDLC model Waterfall

Abstract

The Middle Management Laboratory, Gunadarma University, is one of the laboratories which utilize the computer as a medium for supporting practicum activities. However, the scoring system was less objective for reflecting student's performance. In addition, the manual scoring system hamper the assistant's performance in the practicum score processing. In this paper, the author designed a Preliminary Test Course and Smart Storage applications based on desktop by Software Development Life Cycle (SDLC) method with Waterfall model using Java and SQL Server 2000. These are to improve scoring accuration and to accelerate scoring process in the Middle Management Laboratory, Gunadarma University, so it can be run more efficient. Preliminary Test Course application serves to obtain practicum scores for students who taking practicum in the Middle Management Laboratory, Gunadarma University. At the same time, Smart Storage application is used as the interface in the practicum score processes. At the end of the semester, the practicum scores that have been stored and processed can be exported to a Microsoft Excel document.

Keywords: Preliminary Test Course Application, Smart Storage Application, Java, SQL Server 2000, SDLC Waterfall

1. PENDAHULUAN

Dewasa ini perkembangan cukup pesat terjadi di dunia pendidikan. Hal ini disebabkan oleh mudahnya akses berbagai macam teknologi sehingga meningkatkan efektivitas proses belajar mengajar. Komputer sebagai salah satu produk teknologi dinilai tepat digunakan sebagai alat bantu pengajaran. Berbagai macam pendekatan instruksional yang dikemas dalam bentuk program pengajaran berbantuan komputer atau CAI (*Computer-assisted Instruction*), seperti: *drill and practice*, simulasi, tutorial dan permainan dapat diperoleh melalui komputer [2]. Laboratorium Manajemen Menengah merupakan salah satu laboratorium yang memanfaatkan komputer sebagai media untuk mendukung kegiatan praktikum di Universitas Gunadarma, khususnya pengembangan perangkat

lunak dalam sistem penilaian praktikum. Sebelumnya, penilaian hanya berasal dari kehadiran, keaktifan serta nilai laporan akhir mahasiswa dimana faktor-faktor tersebut kurang mencerminkan performa dari mahasiswa secara obyektif. Dengan demikian, penulis mendapat gagasan untuk menambahkan bobot penilaian baru, yakni tes pendahuluan bagi setiap mahasiswa sehingga pemahaman secara individu dapat lebih terukur. Selain itu, pengolahan nilai praktikum di Laboratorium Manajemen Menengah juga masih dijalankan secara manual, yaitu dengan menggunakan tabel yang ditulis di atas kertas dimana hal ini justru akan memperlambat dan mempersulit tugas asisten dalam mengolah nilai praktikum tersebut. Oleh sebab itu, kebutuhan sumber daya manusia yang cukup banyak tentu saja tidak terelakan agar pengerjaan menjadi lebih efisien. Untuk mengatasi permasalahan tersebut penulis mencoba membuat sebuah aplikasi *Preliminary Test Course* dan *Smart Storage* untuk mempermudah dan mempercepat pengolahan nilai praktikum di Laboratorium Manajemen Menengah Universitas Gunadarma.

2. LANDASAN TEORI

Berikut ini diberikan beberapa landasan teori yang diperlukan dalam merancang aplikasi *Preliminary Test Course* dan *Smart Storage* di Laboratorium Manajemen Menengah Universitas Gunadarma dengan menggunakan Java dan Microsoft SQL 2000.

2.1 Pengenalan Java

Java adalah bahasa pemrograman *multi-platform* dan *multi-device*. Pengguna dapat menjalankan program dengan Java di hampir semua komputer dan perangkat lain yang mendukung Java. Java merupakan bahasa berorientasi objek (OOP), yaitu suatu cara dalam pengorganisasian dan pengembangan perangkat lunak. Pada OOP, program komputer diibaratkan sebagai sekelompok objek yang saling berinteraksi. Deskripsi ringkas OOP adalah mengorganisasikan program sebagai suatu kumpulan komponen yang disebut objek. Objek-objek ini secara independen memiliki aturan-aturan berkomunikasi dengan objek lain [3].

2.2 Microsoft SQL Server

Microsoft SQL Server adalah sistem manajemen *database client/server*. Sebuah *client/server* dalam sistem manajemen database mencakup dua komponen, yaitu komponen *front-end* (klien) untuk menyajikan dan memanipulasi data, serta komponen *back-end* (*server database*) untuk menyimpan, menerima serta melindungi *database*. Penggunaan SQL Server 2000 dapat mendukung berbagai *database* untuk hampir semua ukuran [5].

2.3 Java Database Connectivity (JDBC)

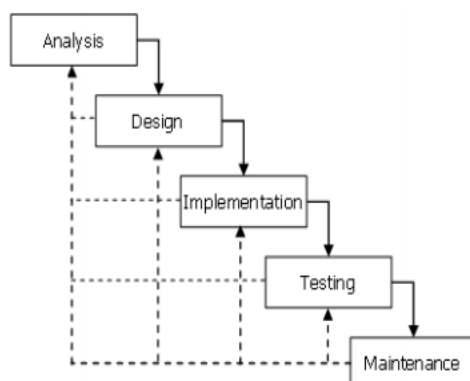
JDBC *Application Programming Interfaces* (API) adalah kelas Java API untuk mengakses segala data berupa tabel. JDBC juga memiliki *library* atau API yang dapat digunakan untuk kebutuhan *database* baik itu koneksi maupun pengolahan data. JDBC API memudahkan pengiriman perintah *Structured Query Language* (SQL) ke sistem *database* relasional dan mendukung bermacam-macam perintah SQL [4].

2.4 Netbeans IDE

Netbeans merupakan alat bantu pemrograman berbasis visual dan *event-driven* seperti halnya Borland Delphi dan Microsoft Visual Studio. Netbeans mencakup *compiler*, *builder* dan *debugger* internal. Ini memudahkan proses pasca perancangan program. Proses *deployment* atau tes dapat dilakukan pula dengan Netbeans [6].

3. METODE PENELITIAN

Metode penelitian yang digunakan dalam penulisan ini berupa tahapan prosedural sistematis yang dikenal dengan sebutan *Software Development Life Cycle* (SDLC) model Waterfall [1] sebagai berikut:



Gambar 1. SDLC model waterfall

4. HASIL DAN PEMBAHASAN

Aplikasi *Preliminary Test Course* dan *Smart Storage* dirancang untuk membantu memudahkan tugas asisten dalam bertugas di Laboratorium Manajemen Menengah Universitas Gunadarma. Kedua aplikasi tersebut berbasis aplikasi desktop dimana ketika diterapkan keduanya akan berjalan dalam komputer yang berbeda, yaitu pada sisi *client* dan *server*. Aplikasi *Preliminary Test Course* terdapat pada sisi *client* untuk mengerjakan soal tes pendahuluan, sedangkan aplikasi *Smart Storage* ditujukan untuk sisi *server* sebagai alat bantu penyimpanan nilai praktikum. Perancangan aplikasi tersebut terdiri dari beberapa fase sesuai dengan metode SDLC model Waterfall berikut ini:

4.1 Fase Analisis

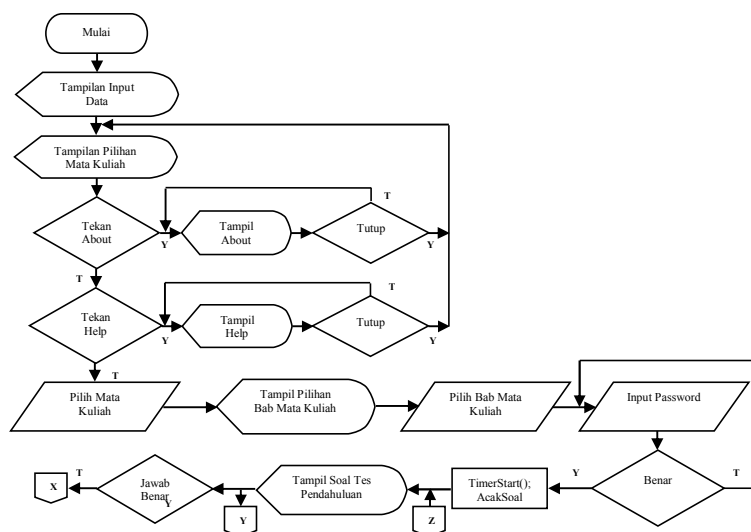
Pada fase ini penulis melakukan analisis terhadap masalah yang muncul di Laboratorium Manajemen Menengah Universitas Gunadarma, yakni proses perolehan nilai tes pendahuluan dan pengolahan nilai praktikum yang masih dilakukan secara manual kemudian akan digantikan dengan penggunaan komputer sebagai alat bantu. Pada sistem sebelumnya, tes pendahuluan diolah secara manual dimana pengerjaan dilakukan pada selembar kertas kemudian dikumpulkan dan dinilai untuk selanjutnya dimasukkan kedalam berkas induk. Proses ini tentu saja memerlukan waktu cukup lama jika peserta praktikum berjumlah besar sehingga muncul gagasan untuk merancang sebuah sistem baru yang akan menggantikan seluruh proses manual tersebut dengan menggunakan komputer sebagai alat bantu.

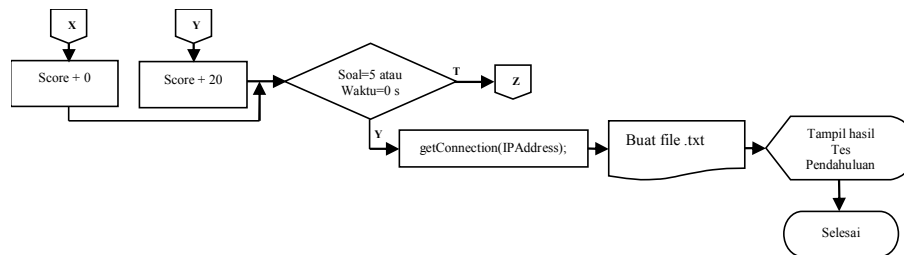
4.2 Fase Desain

Pada fase ini penulis merencanakan alur program, bentuk tampilan, cara kerja aplikasi dan kebutuhan *database* yang akan digunakan.

4.2.1 Aplikasi *Preliminary Test Course*

Pada aplikasi *Preliminary Test Course*, penulis menggunakan lima *form* sebagai antar muka aplikasi. Adapun alur kerja program dimulai dengan memasukkan data diri *user* ke dalam *form login* data mahasiswa, kemudian dilanjutkan dengan *form* pilih matakuliah yaitu pemilihan mata kuliah serta topik yang akan dikerjakan. Demi alasan keamanan, *user* diwajibkan untuk memasukkan kata kunci terlebih dahulu sebelum mengerjakan soal pada setiap bab yang diujikan. Selanjutnya, mahasiswa akan mulai mengerjakan soal tes pendahuluan dimana untuk tujuan simulasi hanya diberikan lima soal yang diperoleh secara acak dengan batas waktu pengerjaan maksimal 2 menit dengan nilai untuk setiap soal benar adalah 20. Perubahan setiap kali selesai menjawab ditampilkan dalam *form* hasil tes pendahuluan sehingga *user* dapat mengetahui nilai pada saat itu. Jika pengerjaan soal telah selesai atau waktu telah tercapai walaupun seluruh soal belum selesai dikerjakan maka program akan otomatis tertutup dan mengirimkan nilai yang didapatkan ke *database server* secara simultan. Pada tahap akhir, program akan menampilkan kembali data mahasiswa beserta nilai yang diperoleh. Penjelasan program tersebut dapat dipahami dengan menggunakan diagram alir pada Gambar 2 berikut ini:

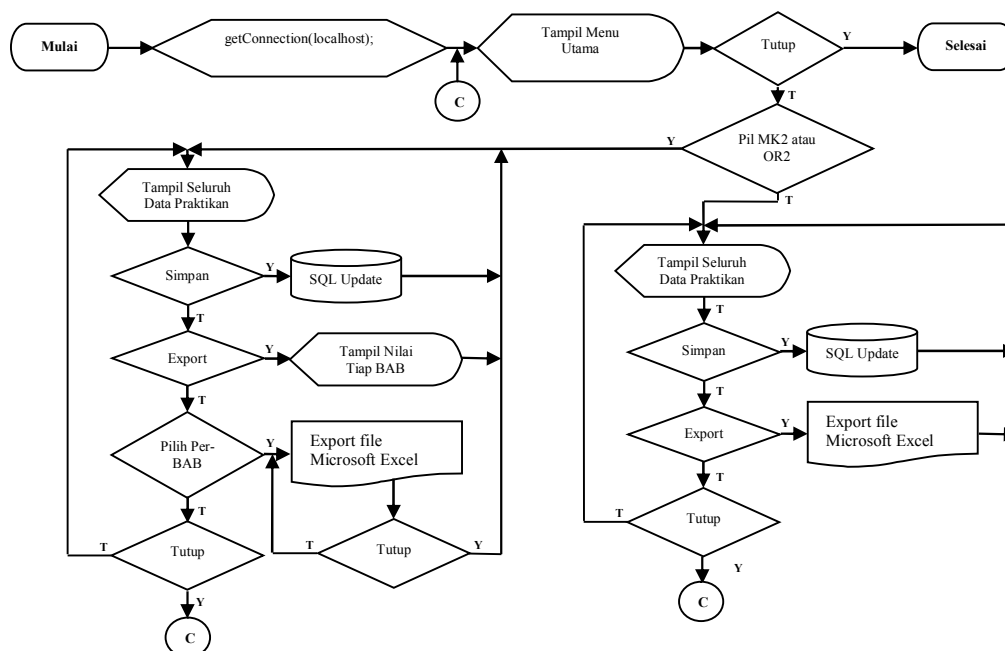




Gambar 2. Diagram Alir Preliminary Test Course

4.2.2 Aplikasi Smart Storage

Aplikasi *Smart Storage* dirancang khusus untuk digunakan oleh asisten dalam mengolah nilai-nilai praktikum di Laboratorium Manajemen Menengah Universitas Gunadarma. Aplikasi ini digunakan oleh para asisten untuk melihat, menambahkan serta mengubah data nilai. Proses penyimpanan nilai yang masih dalam bentuk berkas digantikan dengan menggunakan komputer untuk menyimpan nilai. Aplikasi ini juga dapat membuat laporan hasil akhir praktikum dalam bentuk dokumen Microsoft Excel. Adapun diagram alir dari permasalahan tersebut diberikan pada Gambar 3 berikut ini:



Gambar 3. Diagram Alir Smart Storage

4.3 Fase Implementasi

Pada fase ini penulis mulai membangun *database* menggunakan SQL Server 2000. Serta mulai membuat aplikasi dengan menggunakan pemrograman Java dan Netbeans IDE 7.3 sebagai alat bantu pemrograman. Alur program yang digunakan sesuai dengan rancangan yang telah dibuat sebelumnya pada fase desain. Soal yang terdapat pada tes pendahuluan juga dimasukkan dalam fase implementasi dimana soal-soal tersebut sebelumnya telah disediakan oleh tim Penelitian dan Pengembangan (Litbang). Pada akhir fase ini dibuatlah file *executable* program *Preliminary Test Course* agar dapat didistribusikan pada komputer *client*.

4.4 Fase Testing

Uji coba dilakukan pertama kali pada satu unit *laptop* yang telah terpasang Java dan SQL Server 2000, pada fase *Testing* ini penulis menggunakan alamat *Internet Protocol* (IP) *localhost* sehingga sisi *client* dan *server* masih dalam satu unit *laptop* yang digunakan. Setelah berhasil menjalankan program menggunakan *localhost*, uji coba berikutnya dilakukan pada 43 unit komputer *client* dan satu unit komputer *server* yang telah terpasang Java. Adapun komputer *server* telah terpasang pula SQL Server 2000. Proses instalasi pertama kali dilakukan pada komputer *server*, yaitu memasang *database server* yang sebelumnya telah dibuat dan menjalankan *service* SQL Server agar *database* dapat diakses melalui komputer lain. Instalasi berikutnya dikerjakan pada komputer *client* dimana prosesnya dilakukan dengan melakukan *copy file executable* aplikasi *Preliminary Test Course*. Seluruh komputer *client* dan *server* terhubung dalam satu jaringan LAN, sehingga setiap mahasiswa telah selesai mengerjakan tes pendahuluan maka otomatis nilai tersebut akan masuk ke dalam *database server*. Pada sisi *server*, pengolahan nilai mahasiswa cukup hanya dengan mengoperasikan aplikasi *Smart Storage*. Pada fase *Testing* ini

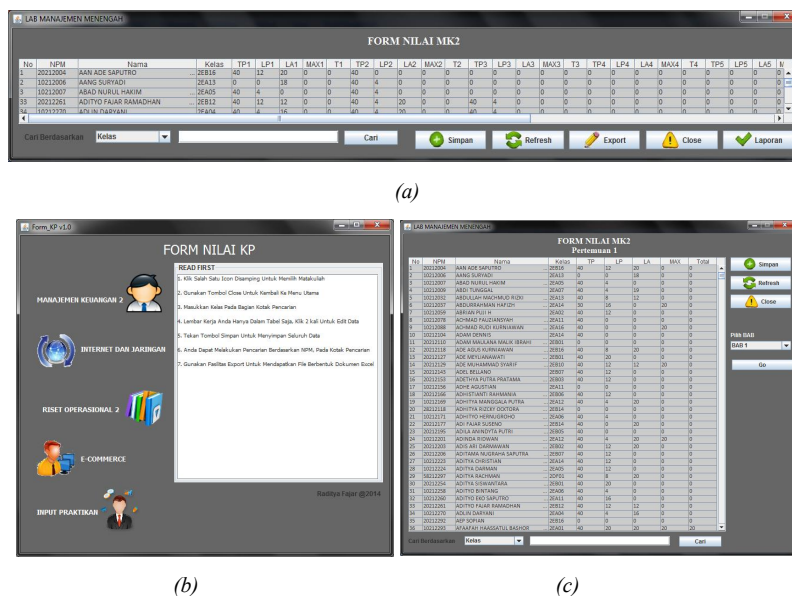
dilakukan validasi apakah aplikasi yang telah dirancang sesuai dengan kebutuhan sistem yang ingin dibangun, apakah mempercepat dan mempermudah tugas asisten, termasuk di dalamnya adalah tahap *debugging* program. Pada proses validasi diperoleh bahwa mahasiswa mengerjakan tes pendahuluan hanya dengan menjawab soal yang terdapat pada aplikasi *Preliminary Test Course* sehingga memudahkan tugas asisten yang sebelumnya harus mengumpulkan dan menilai satu per satu tes pendahuluan tersebut. Selain itu, para asisten tidak perlu lagi mencatat nilai mahasiswa ke dalam berkas induk secara manual, melainkan cukup dengan mengoperasikan aplikasi *Smart Storage* yang dapat mengolah seluruh nilai mahasiswa dengan cepat dan tepat.

Beberapa *output* dari aplikasi *Preliminary Test Course* dapat dilihat pada Gambar 4 berikut ini dimana Gambar 4.a) menampilkan pilihan mata kuliah, Gambar 4.b) menampilkan soal yang diujikan berbentuk pilihan ganda, dan Gambar 4.c) menampilkan nilai praktikum mahasiswa bersangkutan.



Gambar 4. Output Preliminary Test Course

Gambar 5 menampilkan beberapa *output* dari aplikasi *Smart Storage* dimana masing-masing Gambar 5.a), 5.b) dan 5.c) secara berurutan menampilkan data yang disimpan ke dalam bentuk tabel, menu utama program serta data berdasarkan masing-masing bab.



Gambar 5. Output Program Penyimpanan Nilai Praktikum

4.5 Fase Maintenance

Setelah sistem baru diterapkan, tahap selanjutnya adalah melakukan pemeliharaan sistem. Di antaranya meningkatkan kualitas dan performa aplikasi dengan pemeriksaan kembali *bug* program, jika ditemukan adanya *bug* pada program maka dilakukan perbaikan atas *bug* program yang muncul. Pemeliharaan lainnya yaitu

memenuhi kebutuhan baru dari pemakai aplikasi seperti antar muka yang tersedia apakah perlu dilakukan perubahan agar lebih memudahkan *user*. Selain itu dilakukan perawatan terhadap jaringan komputer di Lab. Manajemen Menengah. Pemeriksaan dan perawatan berkala terhadap *database server* dilakukan juga pada tahap ini seperti melakukan *backup database*, mengecek apakah terdapat kesalahan *query* yang berpotensi merusak program.

5. KESIMPULAN DAN SARAN

Berikut ini merupakan kesimpulan dari penelitian yang telah dilakukan. Selain itu, diberikan pula saran dalam mengembangkan aplikasi yang telah dibuat sehingga pemanfaatan Java dan Microsoft SQL Server 2000 dalam menunjang kegiatan praktikum di Universitas Gunadarma, khususnya Laboratorium Manajemen Menengah, tidak terhenti sampai tahap penulisan ini saja.

5.1 Kesimpulan

Berdasarkan hasil yang diperoleh, aplikasi *Preliminary Test Course* dan *Smart Storage* dapat meningkatkan efisiensi dalam kegiatan praktikum di Laboratorium Manajemen Menengah Universitas Gunadarma. Hal ini tercermin dari waktu pengerjaan tes pendahuluan oleh mahasiswa serta proses pengolahan nilai praktikum oleh asisten yang sebelumnya berlangsung lama karena masih dikerjakan secara manual dengan sumber daya manusia yang terbatas dapat dipercepat dengan perancangan aplikasi tersebut.

5.2 Saran

Dalam pembuatan aplikasi *Preliminary Test Course* dan *Smart Storage* Laboratorium Manajemen Menengah Universitas Gunadarma, penulis tidak memperhatikan sisi desain yang digunakan agar tampilan menjadi lebih menarik. Selain itu, tidak tersedia pula *shortcut* pada *keyboard* yang memudahkan pengguna dalam menjalankan aplikasi tersebut. Dengan demikian, tidak tertutup kemungkinan untuk mengembangkan aplikasi ini lebih lanjut, baik terkait *user friendly* maupun dikombinasikan dengan aplikasi lain yang bermanfaat dalam menunjang kegiatan praktikum di Universitas Gunadarma.

6. DAFTAR RUJUKAN

- [1] Bassil, Youssef., 2012. A Simulation Model for the Waterfall Software Development Life Cycle. *International Journal of Engineering & Technology (iJET)*, 2(5). pp. 742-749.
- [2] Jaya, Hendra., 2012. Pengembangan Laboratorium Virtual Untuk Kegiatan Praktikum dan Memfasilitasi Pendidikan Karakter Di SMK. *Jurnal Pendidikan Vokasi*, 2(1), pp. 81-90.
- [3] Naughton, P. dan Schildt, H., 1999. *Java 2: The Complete Reference*. 4th. USA: MCGraw-Hill.
- [4] Reese, G., 2000. *Database Programming With JDBC and JAVA*. 2nd ed. USA: O'Reilly.
- [5] Rozanne, W. et al., 2001. *Microsoft SQL Server 2000: Database Design*. 1st ed. USA: Element K.
- [6] Syakur, A. dan Yusufa, A.S., 2013. Aplikasi Pengaduan Masyarakat Untuk Internal Pegawai Pusat Pelaporan dan Analisis Transaksi Keuangan (PPATK) Berbasis Web. *UG Jurnal*, 7(8), pp. 06-10.