

SISTEM PAKAR PENGENALAN TIPE KEPERIBADIAN MAHASISWA TERHADAP KEMATANGAN PILIHAN PROFESI

Lia Mazia¹⁾, Fadhillah Ahmad²⁾

^{1,2)} Sistem Informasi, STMIK Nusa Mandiri Jakarta
Jl. Ciledug Raya No.108, Cipulir, Jakarta Selatan, Indonesia
lia_mazia@yahoo.com¹⁾, fadhillah.ahmad12@yahoo.co.id²⁾

Abstract

Expert Expert Systems in Introduction of the Personality Type is an expert system designed as a tool to diagnose personality type with a dynamic knowledge base. This knowledge is obtained from various sources including research by experts in the field as well as books related to personality. Expert system method used is forward chaining. This expert system will display a selection of personality profile option that can be selected by the user, in which each option will bring the user's personality profile to get the final result. At the end, the system will display the results of the introduction of personality type and other details. The system is developed using Microsoft Visual Basic 6.0. Conclusions obtained in this study is that the expert system is designed to help make decisions that will be undertaken profession.

Key Words : Expert System, Personality, Forward Chaining.

I. PENDAHULUAN

Seorang ahli psikologi dan kepribadian, Littauer (2008a:5) menjelaskan bahwa “keadaan kita, IQ kita, kebangsaan kita, ekonomi kita, lingkungan kita, dan pengaruh orang tua kita bisa mencetak kepribadian kita”.

Berkaitan dengan ilmu pengetahuan, khususnya di bidang komputer, psikologi dan kepribadian bisa didokumentasikan dan diolah berdasarkan pengetahuan para ahli, yang lebih dikenal dengan kecerdasan buatan (*Artificial Intellegent*). Salah satu bagian pokok dari kecerdasan buatan adalah sistem pakar (*Expert System*). Menurut Turban (2005:2), “sistem pakar adalah suatu sistem yang menggunakan pengetahuan manusia yang terekam dalam komputer untuk memecahkan persoalan yang biasanya memerlukan keahlian manusia”.

Selama ini banyak lulusan perguruan tinggi yang merasa tidak sesuai dengan bidang profesinya karena tuntutan keluarga, pasangan, atau lingkungan. Sedangkan besar sekali pengaruh ketepatan pilihan profesi terhadap prestasi di berbagai bidang. Sebagai akibat tidak sesuai tipe kepribadian dengan pilihan profesinya, pada akhirnya akan menumbuhkan pribadi yang kurang percaya diri, dan berakibat kurangnya potensi diri untuk berprestasi.

Oleh karena itu sistem pakar untuk mengenali tipe kepribadian mahasiswa ini dibuat dengan harapan agar pemakai dapat lebih mengenali dirinya sendiri, bakat dan kemampuan apa yang dimilikinya, sehingga ia bisa menentukan

jalur mana yang akan dia tempuh untuk mengembangkan potensi yang dimilikinya.

II. LANDASAN TEORI

1. Konsep Dasar Sistem Pakar

Sistem menurut Fatta (2008:3) didalam bukunya diartikan sebagai “sekumpulan objek-objek yang saling berelasi dan berinteraksi serta hubungan antar objek bisa dilihat sebagai satu kesatuan yang dirancang untuk mencapai satu tujuan”. Sedangkan menurut Sutabri (2003:3) “sistem adalah suatu kumpulan atau himpunan dari unsur, komponen atau variabel-variabel yang terorganisasi, saling berinteraksi, saling tergantung satu sama lain dan terpadu”.

a. Keahlian

Suatu pengetahuan khusus yang diperoleh dari latihan, belajar, dan pengetahuan. Pengetahuan dapat berupa teori, fakta, aturan, strategi global untuk memecahkan masalah.

b. Ahli (*Expert*)

Melibatkan kegiatan mengenali, dan memformulasikan permasalahan, memecahkan masalah secara cepat dan tepat, menerangkan pemecahannya, belajar dari pengalaman, merestrukturisasi pengetahuan, memecahkan aturan.

c. Mentransfer keahlian (*Trasfering expertise*)

Proses transfer keahlian dari seorang pakar kedalam komputer agar pengetahuan tersebut dapat digunakan oleh orang lain yang bukan pakar.

d. Menyimpulkan aturan (*Inferencing rule*)

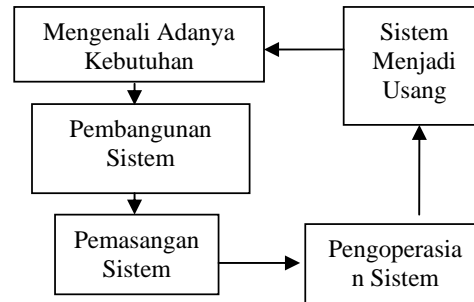
Penyimpulan ini dilakukan komputer yang meliputi prosedur tentang penyelesaian masalah.

- e. Peraturan (*Rule*)
Diperlukan karena mayoritas dari sistem pakar bersifat *rule-based system*, yang berarti pengetahuan disimpan dalam bentuk peraturan.
- f. Kemampuan menjelaskan (*Explanation capability*)
Karakteristik dari sistem pakar yang memiliki kemampuan menjelaskan atau memberi saran mengapa tindakan tertentu dianjurkan atau tidak dianjurkan.

Daur Hidup Sistem

- a. Mengenalinya kebutuhan
Sebelum segala sesuatunya terjadi, timbul suatu kebutuhan atau problem yang harus dapat dikenali sebagaimana adanya.
- b. Pembangunan sistem
Suatu proses atau seperangkat prosedur yang harus diikuti untuk menganalisa kebutuhan yang timbul dan membangun suatu sistem untuk dapat memenuhi kebutuhan tersebut.
- c. Pemasangan sistem
Peralihan dari tahap pembangunan menuju tahap operasional terjadi pemasangan sistem yang sebenarnya, yang merupakan langkah akhir dari suatu pembangunan sistem.
- d. Pengoperasian sistem
Program-program komputer dan prosedur-prosedur pengoperasian yang membentuk suatu sistem informasi semuanya bersifat statis, sedangkan organisasi-organisasi yang ditunjang dengan sistem informasi tadi akan selalu mengalami perubahan-perubahan dan itu dapat disebabkan oleh pertumbuhan kegiatan bisnis, perubahan aturan dan kebijaksanaan ataupun kemajuan teknologi.
- e. Sistem menjadi usang
Kadang-kadang perubahan yang terjadi begitu drastis, sehingga tidak dapat diatasi hanya dengan melakukan perbaikan-perbaikan pada sistem yang berjalan. Tibalah saatnya dimana secara ekonomis dan teknis sistem yang ada sudah tidak layak lagi untuk dioperasikan dan sistem yang baru perlu dibangun untuk menggantikannya.

Untuk dapat menggambarkan daur hidup sistem dapat dilihat pada gambar dibawah ini:

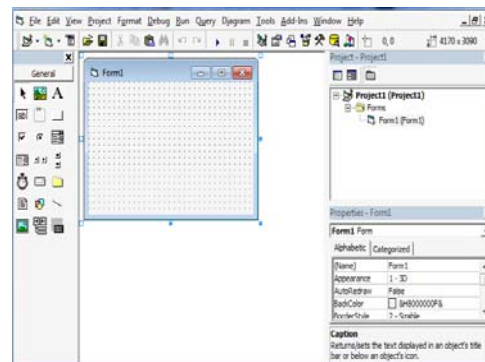


Gambar 1. Daur Hidup Sistem

2. Konsep Dasar Program

Program adalah suatu rangkaian instruksi-instruksi dalam bahasa komputer yang disusun secara logis dan sistematis. Menurut Sutedjo (2002:3) "Program adalah kata, ekspresi, pernyataan, atau kombinasinya yang disusun dan dirangkai menjadi satu kesatuan prosedur yang berupa urutan langkah untuk menyelesaikan masalah yang diimplementasikan dengan menggunakan bahasa pemrograman sehingga dapat dieksekusi oleh komputer". Salah satu tahapan dari pengembangan suatu program adalah menterjemahkan atau memberi kode rancangan terinci yang telah dibuat menjadi suatu program komputer yang siap pakai.

Dalam membuat perancangan program, penulis menggunakan program *Microsoft Visual Basic 6.0*. *Visual Basic* juga merupakan salah satu pemrograman OOP (*Objek Oriented Programming*) yaitu pemrograman yang berorientasi objek menggunakan suatu identitas yang disebut *property*. Dalam pengembangan aplikasi *Visual Basic* menggunakan pendekatan *visual* untuk merancang *user interface* dalam bentuk *form*, sedangkan untuk bahasa pemrogramannya menggunakan bahasa *basic* yang mudah dipelajari.



Gambar 2. Lingkungan Interface Visual Basic 6.0

a. Konsep Dasar Kepribadian

“Kepribadian merupakan pola khas seseorang dalam berpikir, merasakan dan berperilaku yang relatif stabil dan dapat diperkirakan” menurut Dorland (2002:2030). “Kepribadian merupakan jumlah total kecenderungan bawaan atau *herediter* dengan berbagai pengaruh dari lingkungan serta pendidikan, yang membentuk kondisi kejiwaan seseorang dan mempengaruhi sikapnya terhadap kehidupan” menurut Weller (2005:411). Kepribadian yang diolah menjadi sistem dan program oleh penulis pada karya tulis ini membagi tipe kepribadian menjadi beberapa pengelompokan tipe kepribadian. Diantaranya adalah sanguinis, melankolis, koleris, dan phlegmatis.

Littauer (2008b:11) menulis dalam bukunya bahwa kita akan bersenang-senang dengan orang Sanguinis yang Populer, yang mengeluarkan antusiasme. Kita akan serius dengan orang Melankolis yang Sempurna, yang berusaha mengejar kesempurnaan dalam segala hal. Kita akan maju kedepan bersama orang Koleris yang Kuat, yang dilahirkan dengan bakat pemimpin. Dan kita akan rileks dengan orang Phlegmatis yang Damai, yang dengan bahagia menerima kehidupan. Tidak peduli siapa diri kita, kita punya sesuatu untuk dipelajari dari setiap jenis ini.

Dalam bukunya pula, Littauer (2008d:250) juga mengatakan bahwa diantara 4 tipe kepribadian diatas, manusia juga dapat mempunyai kemungkinan campuran diantara ke empatnya. Tipe kepribadian campuran tersebut antara lain:

1. Campuran alami yaitu antara kepribadian sanguinis dengan koleris serta campuran antara kepribadian melankolis dan phlegmatis.
2. Campuran pelengkap yaitu antara kepribadian koleris dan melankolis serta campuran kepribadian sanguinis dan phlegmatis.
3. Campuran yang berlawanan yaitu antara kepribadian sanguinis dan melankolis serta antara kepribadian koleris dan phlegmatis.

III. METODE PENELITIAN

Riset dilakukan untuk mendapatkan data atau informasi yang akurat mengenai keadaan sebenarnya, untuk dianalisa agar dapat mengetahui fakta dan data guna membangun

sebuah sistem pakar yang baik dan benar.

1. Analisa Penelitian

a. Perencanaan

Pada tahap ini dilakukan perencanaan kapan akan mulai melakukan penelitian, berapa lama waktu yang diperlukan dan masalah mana yang terlebih dahulu akan diteliti.

b. Analisis

Di mulai dengan menyatakan suatu masalah, membuat model situasi dari dunia nyata, dan menggambarkan sifat yang penting. Hasil analisa digambarkan dengan memodelkannya kedalam bentuk diagram. Pada tahap ini pula ditentukan kebutuhan yang diperlukan dalam pembuatan sistem pakar pengenalan tipe kepribadian.

c. Desain

Merupakan tahap lanjutan setelah analisis, dimana tujuan sistem diorganisasikan kedalam sub-sistem berdasarkan struktur analisis dan arsitektur yang dibutuhkan. Pada tahap ini dilakukan perancangan sistem dalam kesatuan basis data lalu merancang bentuk antar muka dan dialog layar dengan user. Desain pada sistemnya dinyatakan dalam diagram-diagram UML (*Unified Modelling Language*), dan untuk desain databasenya menggunakan ERD (*Entity Relationship Diagram*). Untuk desain masukan (*input*) disediakan tampilan untuk memasukkan identitas pengguna, dan pilihan-pilihan pernyataan yang sesuai menurut pengguna. Sedangkan untuk keluaran (*output*) disediakan menu cetak hasil pengenalan tipe kepribadian.

d. Implementasi

Pada tahap ini, sistem yang dibuat diimplementasikan ke dalam perangkat lunak. Ujicoba dilakukan dengan menekankan pada proses pengolahan data input, dan pencetakan hasil pengenalan tipe kepribadian.

2. Metode Pengumpulan Data

Metode pengumpulan data yang digunakan oleh penulis dalam menyusun jurnal ini adalah dengan beberapa cara, diantaranya:

a. Observasi (*Observation*)

Penulis melihat objek penelitian kelapangan dengan mengamati secara langsung dan menyebar kuesioner kepada 20 orang mahasiswa.

b. Wawancara (*Interview*)

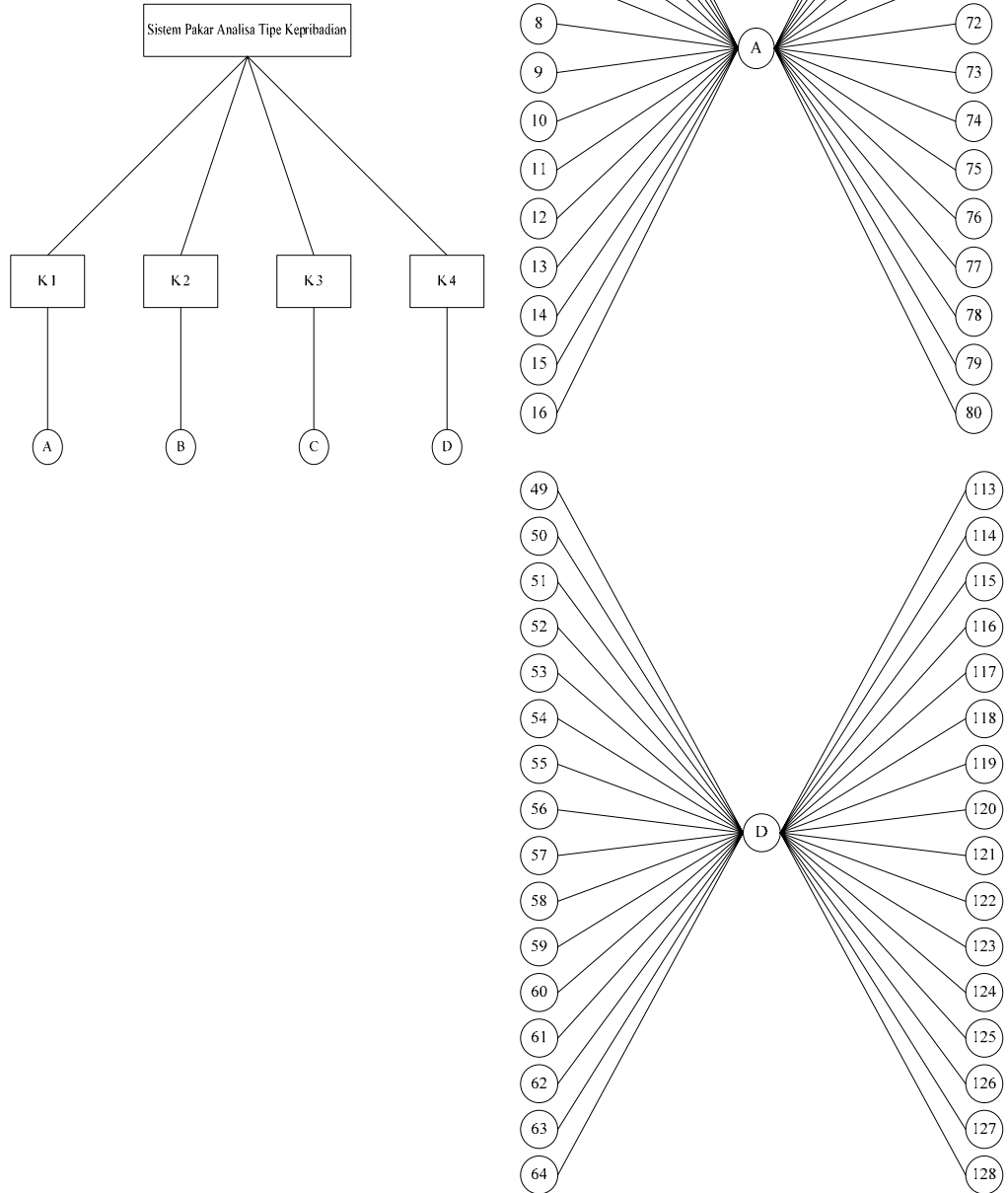
Dilakukan dengan mewawancarai

langsung beberapa orang mahasiswa tingkat akhir dan beberapa pakar yang bergerak di bidang psikologi untuk mendapatkan fakta-fakta yang dapat mendukung penulisan jurnal ini.

- c. Studi Pustaka (*Literature*)
Pengumpulan data yang dilakukan dengan cara mempelajari buku-buku, catatan yang diperoleh penulis.

IV. HASIL DAN PEMBAHASAN

Pohon Keputusan Pakar



Gambar 3. Pohon keputusan Pakar

Pohon keputusan pakar melahirkan 128 profil pengguna yang masing-masing berbeda, untuk membuat hasil pengenalan tipe kepribadian menjadi lebih akurat.

Entity Relationship Diagram (ERD) menurut Ladjamudin (2005:142) “adalah suatu model jaringan yang menggunakan susunan data yang disimpan dalam sistem secara abstrak”. Digunakan untuk menggambarkan hubungan antara *data store* yang ada didalam diagram hubungan data, antara lain:

a. Entitas

Digambarkan dengan kotak persegi panjang dan digunakan untuk menunjukkan sekumpulan orang, tempat, objek atau konsep dan sebagainya yang menunjukkan dimana data dicatat atau disimpan.

b. Hubungan atau Relasi

Digambarkan dengan kotak berbentuk *diamond* atau belah ketupat dengan garis yang menghubungkan ke entitas yang terkait. Maka *relationship* diberi nama dengan kata kerja. Hubungan atau relasi menunjukkan abstraksi dari sekumpulan hubungan yang mengaitkan antara entitas yang berbeda.

c. Atribut

Digambarkan dengan bentuk elips. Atribut menunjukkan karakteristik dari tiap entitas atau sesuatu yang menjelaskan entitas atau hubungan. Sehingga atribut dikatakan elemen dari entitas dan relasi. Dari setiap atribut entitas terdapat satu atribut yang dijadikan sebagai kunci (*key*). Beberapa jenis kunci tersebut antara lain: *Primary key*, *Candidate key*, *Composite key*, *Secondary key*, *Alternate key* dan *Foreign key*.

d. Tingkat Hubungan (*Cardinality*)

Entity Relation Diagram (ERD) juga menunjukkan tingkat hubungan yang terjadi, dilihat dari segi kejadian atau banyak tidaknya hubungan antara entitas tersebut, diantaranya ada tiga kemungkinan hubungan yang ada, yaitu:

1) Satu ke satu (*one to one*)

Tingkat hubungan dinyatakan *one to one* (1:1) jika suatu kejadian pada entitas pertama hanya mempunyai satu hubungan dengan satu kejadian maupun satu hubungan dengan satu kejadian pada entitas kedua

2) Satu ke banyak (*one to many*)

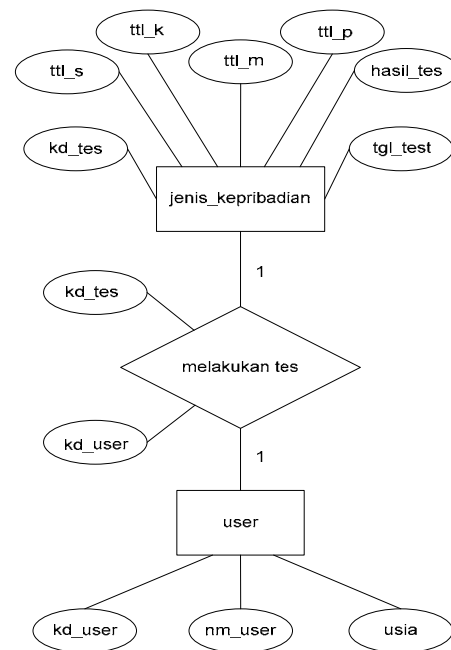
Tingkat hubungan *one to many* (1:M) adalah sama dengan *many to one* (M:1), tergantung dari arah mana hubungan-hubungan tersebut dilihat. Untuk satu kejadian pada entitas yang pertama

dapat mempunyai banyak hubungan dengan kejadian yang kedua, sebaliknya banyak kejadian pada entitas yang kedua hanya bisa mempunyai satu hubungan dengan satu kejadian pada entitas yang pertama.

3) Banyak ke banyak (*many to many*)

Tingkat hubungan banyak ke banyak terjadi jika tiap kejadian pada sebuah entitas akan mempunyai hubungan dengan kejadian pada entitas lainnya, baik dilihat dari entitas yang pertama maupun dilihat dari entitas yang kedua.

Beberapa *modelling tool* yang dipakai adalah bagian dari dasar *Unified Modelling Language*. Menurut Munawar (2005a:25) memberikan penjelasan bahwa “*UML (Unified Modeling Language)* adalah sistem notasi yang sudah dibakukan didunia pengembangan sistem, hasil kerja bersama dari Grady Booch, James Rumbaugh dan Ivar Jacobson”.



Gambar 4. *Entity Relationship Diagram* Sistem Pakar Pengenalan Tipe Kepribadian

UML merupakan salah satu alat bantu yang handal di dunia pengembangan sistem yang berorientasi objek. Dengan menggunakan *UML* kita dapat membuat model untuk semua jenis aplikasi piranti lunak, dimana aplikasi tersebut dapat berjalan pada piranti keras, sistem operasi dan jaringan apapun, serta ditulis dalam bahasa pemrograman apapun. Dengan

UML akan bisa menceritakan apa yang seharusnya dilakukan oleh sebuah *system* bukan bagaimana yang seharusnya dilakukan oleh sebuah sistem.

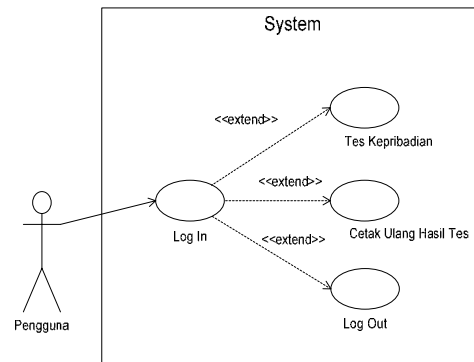
Di dalam UML terdapat 9 Diagram, diantaranya:

- Use Case Diagram*
- Class Diagram*
- Statechart Diagram*
- Activity Diagram*
- Sequence Diagram*
- Collaboration Diagram*
- Component Diagram*
- Deployment Diagram*
- Package Diagram*

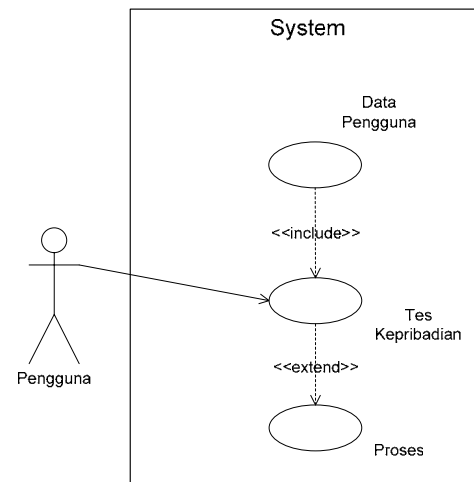
Langkah-langkah penggunaan UML secara umum dapat dibagi menjadi beberapa langkah sebagai berikut:

- Langkah pertama yaitu membuat daftar *business process* dari level tertinggi untuk mendefinisikan aktivitas dan proses yang mungkin muncul.
- Selanjutnya *Use Case* untuk tiap *business process* dipetakan untuk mendefinisikan dengan tepat fungsionalitas yang harus disediakan oleh sistem. Dan *Use Case diagram* diperhalus dan dilengkapi dengan *requirement*, *constraints* dan catatan-catatan lain.
- Fungsi *Deployment Diagram* secara kasar untuk mendefinisikan arsitektur fisik sistem.
- Pendefinisian *requirement* lain (*non-fungsional*, *security* dan sebagainya) yang juga harus disediakan oleh sistem.
- Berdasarkan *Use Case diagram*, mulailah membuat *activity diagram*.
- Diperlukan adanya definisi objek-objek *level* atas (*package* atau *domain*) kemudian pembuatan *sequence* atau *Collaboration Diagram* untuk tiap alir pekerjaan. Jika sebuah *Use Case* memiliki kemungkinan alir normal dan *error*, perlu dibuat satu diagram untuk masing-masing alir.
- Selanjutnya diperlukan adanya rancangan *user Interface* model yang menyediakan antarmuka bagi pengguna untuk menjalankan skenario *Use Case*.
- Berdasarkan model-model yang sudah ada, dapat dibuat *Class Diagram*. Setiap *package* atau *domain* dipecah menjadi *hirarki class* lengkap dengan atribut dan metodenya. Akan lebih baik jika untuk setiap *class* dibuat *unit test* untuk menguji *fungsionalitas class* dan interaksi dengan *class* lain.

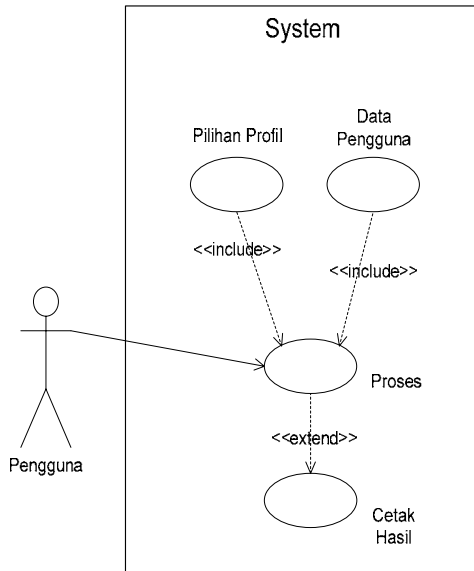
- Setelah *Class Diagram* dibuat, kita dapat melihat kemungkinan pengelompokan *class* menjadi komponen komponen. Karena itu perlu dibuatnya *Component Diagram* pada tahap ini. Juga, diperlukan adanya definisi tes integrasi untuk setiap komponen meyakinkan ia berinteraksi dengan baik.
- Perhalus *Deployment Diagram* yang sudah dibuat. Detailkan kemampuan dan *requirement* piranti lunak, sistem operasi, jaringan, dan sebagainya. Petakan komponen ke dalam *node*.
- Setelah tahap-tahap diatas baru dapat dimulai membangun sistem.
- Apabila tahap-tahap diatas telah terpenuhi maka diperlukan adanya uji modul dan uji integrasi serta perbaikan model beserta *code*-nya. Model harus selalu sesuai dengan *code* yang aktual.



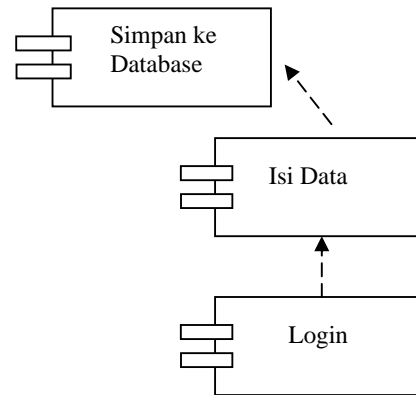
Gambar 5. Use Case Menu Utama



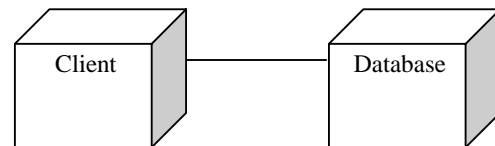
Gambar 6. Use Case Tes Kepribadian



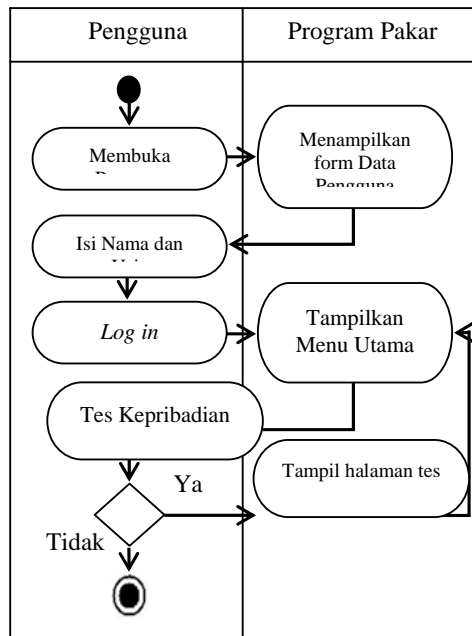
Gambar 7. Use Case Proses



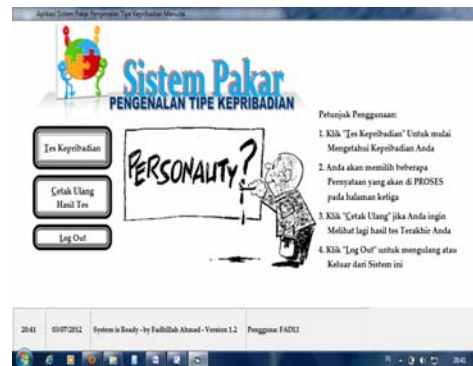
Gambar 9. Component Diagram



Gambar 10. Deployment Diagram



Gambar 8. Activity Diagram Tes Kepribadian



Gambar 11. Tampilan Menu Utama pada Aplikasi



Gambar 12. Tampilan Form Tes Kepribadian



Gambar 13. Tampilan Hasil Pengenalan Kepribadian

Tabel 1. Hasil olah data Kuesioner

Pernyataan	Jumlah*					Jumlah Responden
	S	S	N	T	ST	
Banyak orang ingin mengetahui tipe kepribadiannya	9	10	1	0	0	20
Beberapa orang menyadari profesinya tidak sesuai dengan kepribadiannya	5	9	5	1	0	20
Penggunaan Sistem Pakar Pengenalan kepribadian karena adanya dorongan mengetahui tipe kepribadian diri	3	9	8	0	0	20
Sistem Pakar Pengenalan Kepribadian bisa membantu dalam pemilihan profesi	6	13	1	0	0	20
Sistem Pakar Pengenalan Kepribadian dapat mengenali	10	8	1	1	0	20

diri saya dengan benar						
Penggunaan Sistem Pakar Pengenalan kepribadian hemat biaya	6	9	5	0	0	20
Sistem pakar pengenalan kepribadian untuk membantu pilihan profesi, jarang ditemui	1	9	8	2	0	20
Sistem pakar pengenalan kepribadian hasilnya akurat	6	6	8	0	0	20
Proses Sistem pakar hanya membutuhkan waktu yang singkat.	9	8	3	0	0	20

V. KESIMPULAN

Berdasarkan pembahasan sebelumnya, maka dapat diambil kesimpulan sebagai berikut:

1. Rancangan sistem pakar ini diharapkan mampu menjadi alternatif pemecahan masalah dalam mengetahui tipe kepribadian seseorang.
2. Sistem pakar untuk mengenali tipe kepribadian mahasiswa ini dibuat dengan harapan agar pemakai dapat lebih mengenali dirinya sendiri, bakat dan kemampuan apa yang dimilikinya, sehingga ia bisa menentukan jalur mana yang akan dia tempuh untuk mengembangkan potensi yang dimilikinya.
3. Banyak orang yang ingin mendapatkan saran, khususnya mahasiswa, tentang pilihan profesi yang tepat sesuai dengan kepribadiannya.
4. Terdapat hubungan antara kesesuaian tipe kepribadian dengan pilihan profesinya, yang pada akhirnya berkaitan dengan rasa percaya diri, dan potensi diri untuk berprestasi.
5. Dengan memaksimalkan penggunaan sistem pakar ini, waktu yang dibutuhkan seseorang untuk mendapatkan hasil pengenalan tipe kepribadiannya menjadi

lebih singkat.

Saran-saran

1. Sosialisasi tentang adanya aplikasi sistem pakar pengenalan tipe kepribadian ini hendaknya dilakukan dengan terus menerus.
2. Perlu diadakannya pengembangan aturan-aturan dalam sistem ini dikemudian hari karena sifat ilmu pengetahuan yang dinamis untuk memaksimalkan hasilnya.
3. Perlu dipertimbangkan mengenai adanya personal khusus yang memelihara dan merawat sistem ini agar penanganan terhadap masalah ataupun gangguan yang mungkin timbul dapat ditangani.
4. Tenaga pendidik, orang tua, dan pihak lain yang terlibat dalam membantu siswa atau mahasiswa dalam mempersiapkan dan melakukan pilihan terhadap kariernya, diharapkan dapat mengarahkan siswa atau mahasiswa memilih profesi berdasarkan tipe kepribadiannya yang dapat didapat dari hasil pengenalan tipe kepribadian dengan sistem pakar ini.
5. Adanya pengarahan yang tepat untuk setiap siswa atau mahasiswa, agar mereka dapat lebih mengenali dirinya sendiri, bakat dan kemampuan apa yang dimilikinya, sehingga ia bisa menentukan jalur mana yang akan dia tempuh untuk mengembangkan potensi yang dimilikinya

Daftar Pustaka

- Lttauer, Florence.(2008). *Personality Plus*. Jakarta: Binarupa Aksara.
- Turban, Efraim, Jay Aronson dan Ting Peng Liang. (2005). *Decision Support Systems and Intelligent Systems*, Yogyakarta: Andi.
- Fatta, Hanif. (2008). Analisis & Perancangan Sistem Informasi. Yogyakarta: Andi,.
- Sutabri, Tata.(2003). Analisa Sistem Informasi. Jakarta : Akademi Manajemen Informatika dan Komputer.
- Sutedjo, Budi. (2002). Algoritma dan Struktur Pemrograman. Yogyakarta: Andi Offset.
- Dorland, WAN. (2002). Kamus Kedokteran Dorland, Edisi 29, ECG.
- Weller, B. F. (2005). Kamus Saku Perawat, Edisi 22, EGC.
- Ladjamudin, Al-Bahra Bin. (2005). Analisis dan Desain Sistem Informasi. Yogyakarta: Graha Ilmu.
- Munawar. (2005). Pemodelan Visual dengan UML. Yogyakarta: Penerbit Graha Ilmu.
- Dalyono, M. (2002). Psikologi Pendidikan. Jakarta: Penerbit Rineka Cipta.
- Haryanto, Imam. (2009). Membuat Database Dengan Microsoft Office Access. Bandung: Penerbit Informatika.
- Pascal, Steven Andy. (2007). Tip & Trik Microsoft Office 2007. Jakarta: PT. Elex Media Komputindo.
- Rachmahana, Ratna Syifa'a. (2003). Kepercayaan Diri dan Kemasakan Kerja pada Mahasiswa Tingkat Akhir Universitas Islam Indonesia. Fenomena: Vol 1 Nomor 1: 40-49, 1 Maret 2003. <http://data.dppm.uui.ac.id/uploads/010104KepercayaanDiriDanKemasakanKerja.pdf>.
- Rey, Lourdes, Natalio Extremera dan Mario Pena. (2011). *Perceived Emotional Intelligence, Self-Esteem and Life Satisfaction in Adolescents*. Psychosocial Intervention, Vol 20: 227-234, 2 Agustus 2011. <http://redalyc.uaemex.mx/redalyc/pdf/1798/179819285010.pdf>.
- Sucipto. (2008). Hubungan Antara Kesesuaian Tipe Kepribadian dan Model Lingkungan dengan Kematangan Arah Pilihan Karier. http://eprints.umk.ac.id/143/1/HUBUNGAN_ANTARA_KESESUAIAN_TIPES_KARIBADIAN.pdf.