

PENGEMBANGAN SISTEM PENALARAN BERBASIS KASUS UNTUK MENGANTISIPASI MASALAH KEGAGALAN SISTEM INFORMASI

Iping Supriana Suwardi¹, Juwairiah^{1,2}

¹Program Studi Informatika, STEI, ITB

²Jurusan Teknik Informatika, FTI, UPN "Veteran" Yogyakarta

E-mail: iping@informatika.org, juwai_riah@yahoo.com

ABSTRAKSI

Berbagai upaya untuk mengatasi kegagalan sistem informasi pada saat operasional merupakan kajian dari banyak pihak. Makalah ini berisi kajian alternatif dalam rangka mengatasi masalah tersebut. Pendekatan yang digunakan disini adalah model Penalaran Berbasis Kasus (Case-Based Reasoning) dengan strategi utamanya menggunakan ukuran kemiripan (similarity). Secara garis besar, pengalaman-pengalaman kegagalan SI dikumpulkan dari berbagai sumber, kemudian akan dipelajari, diklasifikasi, dan ditransformasi sehingga menjadi elemen pengetahuan yang akan menjadi basis kerja dari mekanisme yang dikaji.

Kata kunci: Kegagalan SI, Penalaran Berbasis Kasus, ukuran kemiripan (similarity)

Pendahuluan

Sistem Informasi (SI) merupakan faktor yang penting bagi sebuah organisasi untuk dapat bertahan hidup dan bersaing dalam bisnis serta dapat meningkatkan efisiensi dan keefektifan bisnis sehari-hari dengan mengintegrasikan proses-proses dan struktur organisasi. Namun sering terjadi problem-problem dalam SI yang menyebabkan kegagalan sistem dan akan mempengaruhi organisasi yang terlibat dalam penciptaan, pemeliharaan, dan penggunaan, serta mempunyai dampak yang besar pada orang-orang yang terlibat di dalamnya.

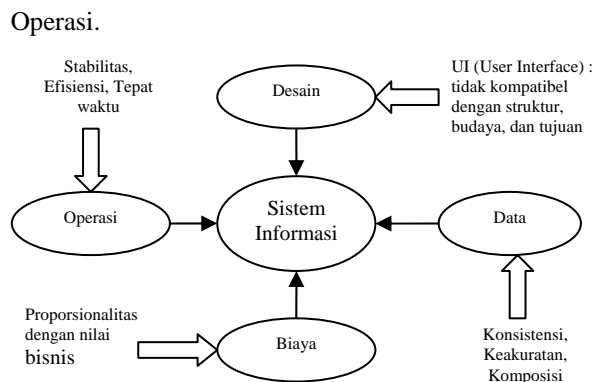
Sistem dianggap mengalami "kegagalan" (*failure*) jika tidak berjalan/ melakukan sesuai yang diharapkan atau tidak beroperasi pada suatu saat tertentu atau tidak dapat digunakan seperti yang dikehendaki [LAU02].

Dalam sistem informasi, terdapat tingkat kegagalan yang cukup tinggi dan sering terjadi kasus kegagalan yang berulang-ulang. Oleh karena itu maka dianggap perlu untuk mengambil pelajaran dari berbagai pengalaman kegagalan di masa lalu untuk mengantisipasi agar kegagalan yang sama tidak terjadi lagi dalam pembangunan sistem informasi yang akan dilakukan. Dengan membandingkan problem atau gejala kegagalan dalam pengalaman masa lalu, maka dapat diambil langkah antisipasi yang tepat. Pendekatan yang digunakan adalah model Penalaran Perbasis Kasus.

Mengidentifikasi Penyebab Problem-Problem Kegagalan Sistem

Karena penyebab kegagalan ini bervariasi, maka setiap kasus harus dipisah-pisahkan dan diselidiki untuk mengetahui apa yang salah di masa lalu. Setiap kasus kegagalan yang pernah terjadi dikumpulkan untuk mendapatkan informasi tentang problem sistem, penyebabnya, dan rekomendasi cara untuk mengatasi kegagalan tersebut.

Problem yang menyebabkan kegagalan sistem informasi dibagi menjadi banyak kategori. Dalam sistem Informasi dapat diklasifikasikan 4 bidang problem utama, yaitu: Desain, Data, Biaya, dan



Gambar 1. Bidang Problem Utama dalam SI

Hasil dari implementasi sistem informasi ditentukan oleh faktor-faktor berikut:

- Peran user dalam proses implementasi
- Tingkat dukungan manajemen untuk usaha implementasi
- Tingkat kompleksitas dan resiko dari proyek implementasi
- Kualitas dari manajemen proses implementasi

Kesalahan-kesalahan yang mungkin terjadi dalam tahapan pengembangan sistem yang dapat dianggap sebagai penyebab kegagalan jika proses implementasi dikelola secara buruk, yaitu: kesalahan dalam Analisis, Desain, Pemrograman, dan Konversi.

Karena SI yang dimaksudkan disini adalah yang berbasiskan software, maka dapat diklasifikasikan penyebab error software yang bisa menyebabkan kegagalan (*failure*) software, yaitu:

1. Kesalahan Definisi Requirement
2. Kegagalan Komunikasi antara Client-Developer
3. Penyimpangan Requirement software
4. Error Desain Logika
5. Error Pengkodean
6. Ketidakpatuhan instruksi pengkodean dengan dokumentasi
7. Kekurangan dalam proses Pengujian
8. Error Prosedur
9. Error Dokumentasi

Penalaran Berbasis Kasus (PBK)

Penalaran Berbasis Kasus atau PBK adalah salah satu pendekatan berbasis pengetahuan untuk mempelajari dan memecahkan problem berdasarkan pada pengalaman masa lalu. Pengalaman yang lalu disimpan dalam bentuk problem (“kasus”) yang dipecahkan dalam tempat yang disebut “Basis Kasus”. Basis Kasus adalah kumpulan dari kasus-kasus.

Siklus PBK digambarkan dengan 4 proses, yaitu: Menemukan kembali (*Retrieve*) kasus yang paling mirip, Menggunakan kembali (*Reuse*) informasi dan pengetahuan dalam kasus untuk memecahkan problem (proses ini disebut “transfer solusi”), Merevisi atau memperbaiki (*Revise*) solusi yang diusulkan, dan Menyimpan (*Retain*) pengalaman ini untuk memecahkan problem yang akan datang.

Ada 4 tempat (*Container*) pengetahuan untuk sistem PBK, yaitu: Kosa Kata (*Vocabulary*), Ukuran Kesamaan (*Similarity Measure*), Transformasi Solusi (*Solution Transformation*), dan Kasus-Kasus (*Cases*).

Ada 3 tipe berbeda dari sistem PBK berdasarkan tingkat keterlibatan user yang semakin meningkat, yaitu: Diagnosis, Pendukung Keputusan, dan Manajemen Pengetahuan [ALT01].

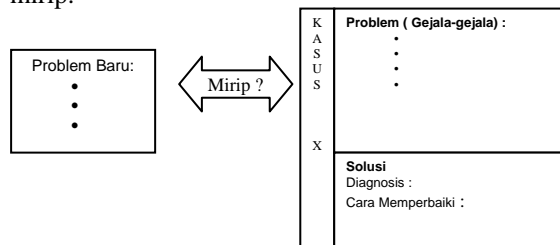
Diagnosis Berbasis Kasus untuk Kegagalan Sistem Informasi

Sebuah kasus menggambarkan situasi yang berkenaan dengan diagnosa dan memuat:

- deskripsi problem atau gejala-gejala (*symptoms*)
- deskripsi kegagalan dan penyebab
- deskripsi strategi memperbaiki

Dalam diagnosis berbasis kasus, kumpulan kasus-kasus akan disimpan dalam basis kasus, dan akan menemukan kasus yang mirip dengan problem yang ada saat ini serta menggunakan kembali cara untuk memperbaiki.

Cara kerjanya: membandingkan problem baru dengan setiap kasus, dan memilih kasus yang paling mirip.



Gambar 2. Cara Kerja Diagnosis Berbasis Kasus Similarity

Strategi utama dalam PBK adalah dengan menggunakan konsep kemiripan (*similarity*), karena problem yang ada saat ini tidak sama persis dengan problem yang ada dalam kasus-kasus lampau. Kasus dengan nilai *similarity* paling besar dianggap sebagai kasus yang “paling mirip”.

Ada beberapa rumus untuk perhitungan *similarity*. Dalam sistem yang akan dibangun, menggunakan perhitungan *similarity* lokal (*similarity* di antara setiap atribut kasus) simbolik bernilai tunggal, yaitu :

$$sim(a,b) = \begin{cases} 1 & \text{jika } a = b \\ 0 & \text{jika } a \neq b \end{cases}$$

kemudian menghitung *similarity* globalnya (*similarity* antara 2 kasus) menggunakan metode *Block-City*, yaitu :

$$sim(A,B) = \frac{1}{p} \sum_{i=1}^p sim_i(a_i,b_i)$$

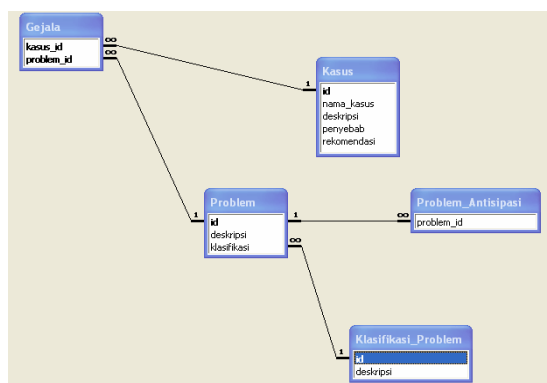
Ukuran *similarity* akan berupa nilai dalam range 0-1 atau 0-100%.

Rancangan Aplikasi PBK untuk Mengatasi Masalah Kegagalan Sistem Informasi

Sistem yang akan dibangun merupakan aplikasi untuk mendiagnosis penyebab kegagalan sistem informasi dan memberikan rekomendasi cara untuk mengatasinya. Sistem ini terdiri dari 2 aplikasi, yaitu aplikasi administrasi untuk admin dan aplikasi user untuk pengguna yang ingin mencari solusi.

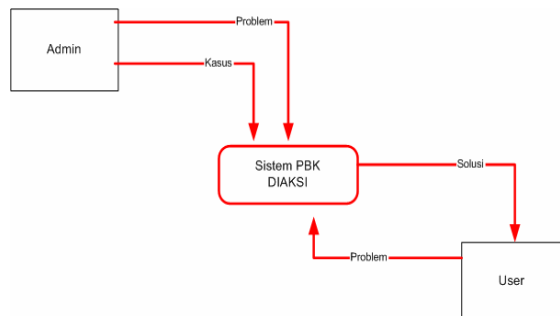
- Aplikasi administrasi digunakan untuk menangani pengelolaan problem dan pengelolaan kasus, yaitu : penambahan, perubahan, dan penghapusan. Aplikasi administrasi menggunakan 2 acuan (referensi), yaitu acuan problem dan acuan kasus.
- Aplikasi user untuk mencari solusi penyebab kegagalan sistem informasi dan cara mengatasinya. Pengguna memasukkan problem kegagalan sistem yang ingin diantisipasi, kemudian sistem akan melakukan pencarian beberapa kasus yang ada dalam basis kasus yang bersesuaian dengan problem yang dimasukkan tersebut. Sistem akan menghitung *similarity* masing-masing kasus, kemudian akan melakukan pengindeksan atau mengurutkan ranking berdasarkan ukuran *similarity*. Sistem akan memberikan rekomendasi kasus yang paling mirip yaitu kasus dengan ukuran *similarity* terbesar.

Diagram Relationship

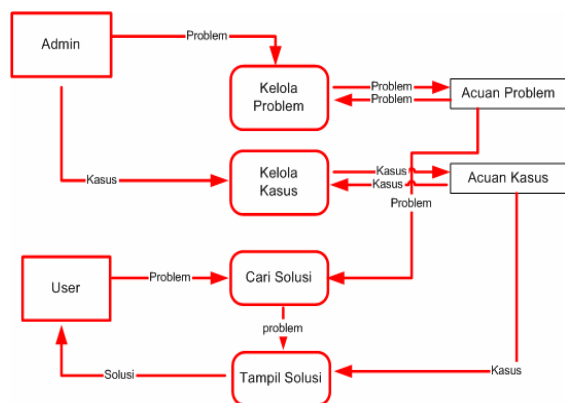


Gambar 3. Diagram Relationship

Diagram Aliran Data (DFD)



Gambar 4. DFD Level 0



Gambar 5. DFD Level 1

Kesimpulan dan Saran

PBK dapat dijadikan salah satu pendekatan untuk mengatasi masalah kegagalan sistem informasi, sehingga diharapkan agar tidak terjadi lagi kasus kegagalan yang sama seperti pada pengalaman masa lalu.

PBK tidak menjamin solusi terbaik atau solusi optimum karena PBK hanya memberikan solusi berdasarkan kasus-kasus yang disimpan. Untuk mendapatkan solusi yang tepat, PBK membutuhkan kasus yang banyak dan lengkap. Semakin banyak kasus yang tersedia, maka akan didapatkan solusi yang lebih tepat. Tetapi kekurangannya adalah karena PBK membutuhkan tempat penyimpanan yang besar untuk menyimpan kasus yang banyak.

Daftar Pustaka

- [1] Althoff, K.-D., *Case-Based Reasoning, Handbook of Software Engineering & Knowledge Engineering* (ed.S.K Chang) Vol 1, World Scientific, Singapore, 2001
- [2] Donaldson, John & Jenkins, John, *SYSTEMS FAILURES: An approach to understanding what can go wrong*, www.cs.mdx.ac.uk/research/SFCpapersAJMD_EuroMicro00.pdf, 2000
- [3] Galin, Daniel, “*Software Quality Assurance*”, Pearson Education Limited, England, 2004
- [4] Kolodner, Janet L., *An Introduction to Case-Based Reasoning*, Artificial Intelligence Review, USA, 1992
- [5] Kolodner, Janet L., *Case-Based Reasoning*, Morgan Kaufmann Publisher, Inc., USA, 1993
- [6] Laudon, Kenneth C & Laudon, Jane P, *Managing the Digital Firm Management Information System*, Prentice-Hall, Inc, New Jersey, 2002
- [7] Watson, Ian, *Applying Case-Based Reasoning : Techniques for Enterprise Systems*, Morgan Kaufmann Publisher, Inc., USA, 1997

