

## PEMANFAATAN STUDENT MODEL PADA SISTEM E-LEARNING

Ana Hadiana

Sekolah Tinggi Manajemen Informatika dan Komputer LIKMI

Jl. Ir. H. Juanda 96 Bandung 40132

E-mail: anahadiana@yahoo.com

---

### ABSTRAK

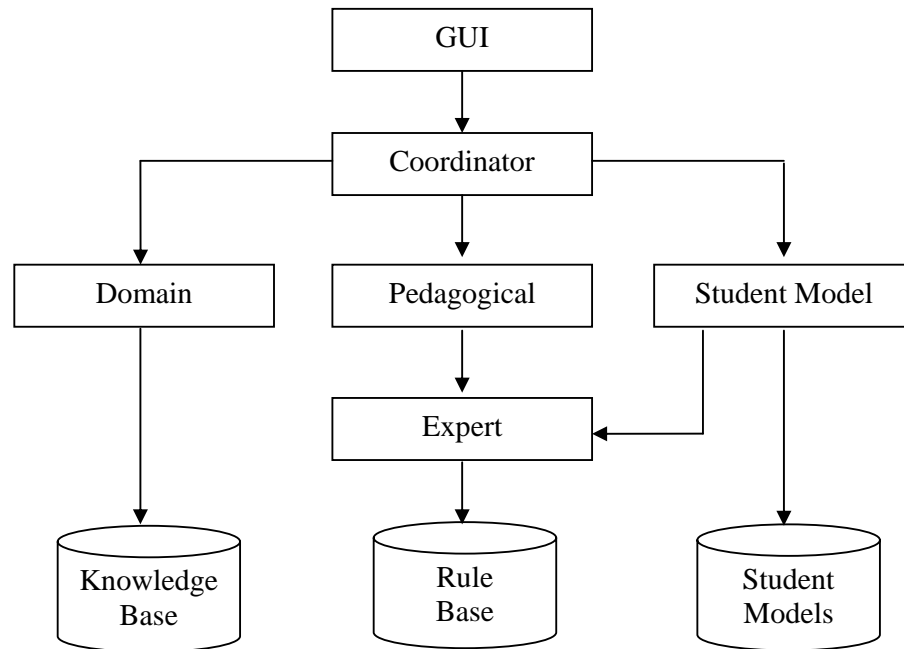
Makalah ini membahas secara global tentang bagaimana pemanfaatan *Student Model* dalam suatu sistem pembelajaran berbasis komputer atau internet atau *e-Learning*. *Student Model* pada dasarnya merupakan penyimpanan proses pembelajaran pelaku *e-Learning* yang dibuat dengan menggunakan sebuah model *template* yang pada awalnya diisi oleh beberapa atribut dari masing-masing pelaku.

**Kata-kata kunci:** *Student Model*, *e-Learning*, internet

---

### 1. PENDAHULUAN

*E-Learning* merupakan salah satu bentuk aplikasi sistem informasi yang dikhususkan untuk mendukung proses pembelajaran melalui jaringan internet. Dalam setiap sesi pembelajaran pada *e-Learning*, sistem secara permanen akan mengakses kemampuan yang dimiliki oleh setiap pelaku sistem dan mencocokkannya dengan *Student Model* secara bersamaan[1]. Kemudian, sistem akan mencocokkan jenis materi apa saja yang bisa dan layak diberikan kepada setiap pelaku untuk dipelajari berdasarkan pada parameter dari masing-masing *Student Model* -nya.



**Gambar 1. Arsitektur Sistem ITS**

Pada gambar 1 dijelaskan secara garis besar tentang arsitektur dasar dari sistem *e-Learning* yang menggunakan *Student Model* yang pada intinya tersusun dari beberapa modul utama[2], yaitu sebagai berikut:

- a. *Pedagogical modul* menyediakan basis infrastruktur pengetahuan yang diperlukan untuk mempresentasikan materi ajar berdasarkan pada keadaan atau status *Student Model*.
- b. *Expert module* menggunakan JESS (*Java Expert System Shell*) sebagai *a rule-based interface engine*-nya. *Pedagogical module* menggunakan *Expert module* untuk membuat keputusan di dalam *curriculum sequencing* dan mengevaluasi *student model*.
- c. *Domain modul* mengandung pengetahuan tentang *design pattern* dan materi pembelajaran aktual.
- d. *Coordinator modul* yang mengontrol fungsionalitas dari keseluruhan sistem.
- e. *Student Model modul*, menyimpan secara detail perkembangan dari hasil pembelajaran yang berlangsung.
- f. *GUI modul*, menyediakan *interface* yang diperlukan dalam proses pembelajaran dengan sistem ini.

Adapun ke depan dalam pengembangan sistem pembelajaran ini, komponen *software* yang bisa digunakan adalah pada dasarnya cukup menggunakan komponen GUI berbasis HTML untuk pembuatan antar muka *client*-nya. Sedangkan untuk sisi *server* sebagai *JSP container*-nya dengan menggunakan *software Web Server Tomcat 4.0* [4].

## 2. STUDENT MODEL

*Student model* pada prinsipnya merupakan suatu struktur data yang berfungsi untuk menyimpan secara rinci data-data yang berkaitan erat dengan *student's current problem-solving state*, data tentang kemajuan pembelajaran *long-term knowledge*, dan data tentang *essential for adapting the material to the student's characteristics (attributes)*. Oleh sebab itu, di dalam sistem *e-Learning* ini sangat penting dipertimbangkan tiga kategori tentang karakteristik *student*[3].

- a. *Personal Data* – mencatat karakteristik setiap *student* (nama, identitas, *e-mail*, ...)
- b. *Performance Data* – *student's cognitive* dan *individual characteristic*, dan juga data lain seperti karakteristik umum dalam waktu yang panjang (*general long-term characteristics*)
- c. *Overlay Data* – level sekarang dari *mastery of design patterns* dan *attribute* yang terkait dengan elemen-elemen di dalam model *domain*.

Sebenarnya untuk kepentingan sistem *e-Learning*, ada beberapa teknik yang bisa digunakan untuk membuat suatu *Student Model*, dimana yang paling sering dan populer digunakan adalah teknik *overlay*, *stereotype model*, dan model kombinasi *overlay-stereotype*. Di dalam sistem ini digunakan *Student Model* dengan model kombinasi, karena memiliki beberapa kelebihan dari pada hanya menggunakan *overlay* atau *stereotype* saja.

Pada sistem ini, saat pertama kali pelaku melakukan proses pembelajaran, terlebih dahulu harus melakukan proses registrasi ke dalam sistem, kemudian sistem akan membuat sebuah *student model* baru dan melakukan inisialisasi *Student Model* tersebut dengan menset *default value* dari *stereotype*. Inilah dimana sistem memilih *stereotype*, berdasarkan pada interaksi pertama kali antara pelaku dengan sistem. Sistem secara bertahap akan memproses beberapa karakteristik lain dari *Student*

*Model* berdasarkan pada estimasi pengetahuan *student*, seperti tingkat pemahaman (*degree mastery*), level pengalaman, *learning style*, *detail level*, dan yang lainnya.

*Attribute value* didalam *Student Model* dihitung dengan cara meng-*apply* sebuah *dedicated group* dari *rules* dan *simple functions* dari *Pedagogical Module*. Nilainya diubah-ubah pada seluruh sesi pembelajaran.

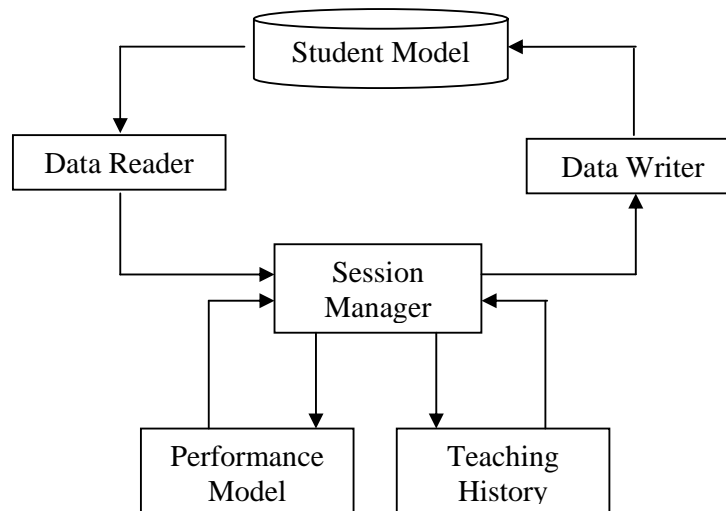
Pada akhir setiap sesi, sistem akan menyimpan *Student Model* sebagai dokumen XML. Sehingga pada waktu berikutnya pada saat *student login* ke sistem lagi, maka data dari dokumen XML yang tersimpan akan digunakan kembali untuk melakukan proses inisialisasi *Student Model*.

Beberapa komponen di dalam *Pedagogical Module* menentukan bentuk dari *Student Model*. Ketentuan tersebut berkaitan erat dengan pemilihan pada materi pembelajaran yang tepat bagi setiap pelaku, pemilihan *learning method* yang tepat, *curriculum sequencing*, dan lain-lain. Selama proses seperti ini, *Pedagogical Module* secara kontinyu terus berkomunikasi dengan *Student Model* dalam rangka mendapatkan informasi yang relevan.

### 3. KOMPONEN PENDUKUNG STUDENT MODEL

*Student Model* pada prinsipnya mempunyai lima komponen utama[5] yaitu seperti pada gambar 2 sebagai berikut:

- a. *Performance model*, menyimpan data yang terkait dengan *assessment* dari *skill* yang dimiliki *student*, dan data yang terkait dengan pengetahuan *student*, *learning style* (gaya belajar) yang sebelum-sebelum nya.
- b. *Teaching history model*, menyimpan *track* data materi pelajaran yang disampaikan kepada *student* dalam suatu sesi dan *teaching* unit dari *student* yang telah dikuasai.
- c. *Data Reader*, membaca data dari *student model* dalam bentuk dokumen XML, ini berdasarkan pada SAX Parser, dan dijalankan pada saat memulai suatu sesi.
- d. *Data Writer*, menyimpan data *student model* dalam bentuk dokumen XML, pada setiap mengakhiri sesi.
- e. *Session Manager*, berfungsi untuk mengkoordinir seluruh komponen.

Gambar 2: Komponen Pendukung *Student Model*

#### 4. KESIMPULAN

*Student Model* dalam sistem *e-Learning* secara fungsi sepenuhnya tersusun dari komponen sistem yang lainnya. Fitur yang membedakan antara model yang diusulkan dengan model lainnya adalah cara untuk menyimpan seluruh data yang mempunyai relevansi tentang pelaku *e-Learning* – dokumen XML yang tersendiri, dimana dibuat pada saat *student* pertama kali registrasi ke sistem.

Pendekatan terhadap *Student Model* di sistem ini yaitu dengan menggunakan kombinasi dari teknik *stereotype* dan *overlay*. Kelebihan utamanya dibanding dengan sistem yang serupa adalah pendekatan yang universal dalam pembuatan model. Cara seperti ini bisa diaplikasikan ke dalam sistem *e-Learning* dalam berbagai *domain* tanpa perlu mengubah sama sekali atau hanya dengan sedikit perubahan saja, hal ini sangat tergantung pada *requirement* dari *pedagogical module*.

Pada penelitian berikutnya, *Student Model* perlu lebih dimodifikasi dan dikembangkan agar bisa diterapkan untuk mendukung konsep pembelajaran berbasis *collaboration learning* di antara *student-student* yang memiliki kemiripan model pembelajarannya.

## 5. DAFTAR PUSTAKA

- [1]. Bunt, C. Conati, “*Probilistic Student Modeling to Improve Exploratory Behavior*”, Jour. Of User Mod. And User-Adapted Inter. 13(3), 2003, pp.269-309.
- [2]. E. Gamma, et. Al., “*Design Pattern – Elements of Reusable Object Oriented Software*”, Addison-Wesley Publishing Company, USA 1995
- [3]. O. Conlan, “*Novel Components for supporting Adaptivity in Education Systems – Model based Integration Approach*,” In. Proc. Of the 8<sup>th</sup> ACM Int. Conf. on Mult, CA, USA, 2000, pp. 519-520.
- [4]. Parsian Mahmoud, “*JDBC Recipes – A Problem Solution Approuch*”, APress Publishing, 2005
- [5]. Z. Yuijian, M. W. Evens, “*A Practical Student Model in an Intelligent Tutoring System*,” In. Proc. Of the 111<sup>th</sup> IEEE Int. Conf. on Tools with AI, USA, 1999, pp. 13-18