

Jurnal

Teknologi Pembelajaran

Teori dan Penelitian

TAHUN 4, NOMOR 2, OKTOBER 1996

ISSN 0854-7599

- Alexander Romiszowski* Instructional System Design and Development for A Networked Society
- Justus H. Lewis*
Dapeng Tien Neural Network Based Learning Organizations and Their Relevance to Distance Education
- Gunadi Harry Sulisty* Measurement of the Trait Structure of Communicative Speaking Abilities
- Y. G. Harto Pramono* Pengembangan Prosedur Perangkat Lunak Pembelajaran Berbantuan Komputer
- Umar Tirtaraharja* Link and Match sebagai Teknologi Pemecahan Masalah Relevansi Pendidikan
- Waras Bin Khamdi* Pergeseran Paradigma Penelitian Kelas: Dari Proses-produk ke Analisis Interpretatif
- I Nyoman Sudana Degeng* Pengaruh Strategi Penataan Isi Modul, Gaya Kognitif, dan Strategi Belajar Mahasiswa terhadap Perolehan Belajar
- Amat Mukhadis* Pengaruh Pengorganisasian Isi Prosedural, Locus of Control, dan Bakat Berpikir Mekanik terhadap Hasil Belajar dan Transfer Belajar di Sekolah Teknologi Menengah



Diterbitkan oleh
Ikatan Profesi Teknologi Pendidikan Indonesia (IPTPI) Jakarta
Bekerja Sama dengan Program Studi Teknologi Pembelajaran
Program Pascasarjana IKIP MALANG

Jurnal Teknologi Pembelajaran Teori dan Penelitian

Penyunting

I Nyoman Sudana Degeng

Penyunting Pelaksana

Amat Nyoto

Waras Bin Khamdi

Wasis Djoko Dwiyo

Pelaksana Tata Usaha

Sugondo

Alamat Kantor

Program Studi Teknologi Pembelajaran, Program Pascasarjana IKIP MALANG, Jalan Surabaya 6 Malang 65145, Gedung F1, Telepon (0314)551334; atau 551312, 551213, 551314, 551315, pesawat 317; Fax (0341)551334.

Penyunting Ahli

Abdul Gafur	IKIP YOGYAKARTA
Arief S. Sadiman	PUSTEKKOMDIKBUD
Iskandar Wiryokusumo	IKIP SURABAYA
Lusiana	PAM KEPERAWATAN DEPKES
M. Dimiyati	IKIP MALANG
Suhardjono	UNIVERSITAS BRAWIJAYA
T. Raka Joni	IKIP MALANG
Umar Tirtarahardja	IKIP UJUNG PANDANG
Veronica L. Diptoadi	UNIVERSITAS WIDYA MANDALA
Wayan Ardhana	IKIP MALANG
Yusufhadi Miarso	IKIP JAKARTA

Jurnal Teknologi Pembelajaran: Teori dan Penelitian (ISSN 0854-7599) diterbitkan oleh Ikatan Profesi Teknologi Pendidikan Indonesia (IPTPI) bekerja sama dengan Program Studi Teknologi Pembelajaran, Program Pascasarjana IKIP MALANG; berisi tulisan tentang gagasan konseptual, kajian dan atau aplikasi teori; dan hasil penelitian dan atau pengembangan teknologi pembelajaran; serta resensi buku, resensi perangkat lunak pembelajaran, dan obituari tokoh ilmuwan teknologi pembelajaran yang sesuai dengan misi dan tujuan jurnal ini. Terbit dua kali setahun pada bulan April dan Oktober. Harga langganan 2 nomor setahun Rp15.000,00 (tambah ongkos kirim Rp 1.500,00). Uang langganan dapat dikirim lewat wesel pos ke alamat kantor Tata Usaha, atau melalui Bank Umum Nasional, Jalan Kawi-Malang, rekening nomor 617.265811.6 atas nama Dr. Lusiana. Jika para penyumbang tulisan menghendaki *reprint* dapat menghubungi pelaksana tata usaha.

Penyunting menerima sumbangan tulisan yang belum pernah dipublikasikan dalam media cetak lain. Naskah diketik dengan spasi renggang pada kertas kuarto, panjang 10-20 halaman sebanyak dua eksemplar (lebih lanjut, baca Petunjuk bagi Penulis pada sampul belakang). Naskah yang masuk dievaluasi oleh Penyunting Ahli dan atau Peninjau Ahli. Penyunting Pelaksana dapat melakukan perubahan pada tulisan yang layak dimuat untuk keseragaman format, tanpa mengubah maksud dan isinya.

Jurnal Teknologi Pembelajaran

Teori dan Penelitian

TAHUN 4, NOMOR 2, OKTOBER 1996

ISSN 0854-7599

TEORI

- Alexander Romiszowski.* Instructional System Design and Development for A Networked Society 89
- Justus H. Lewis & Dapeng Tien.* Neural Network Based Learning Organizations and Their Relevance to Distance Education 108
- Gunadi Harry Sulisty.* Measurement of the Trait Structure of Communicative Speaking Abilities 112
- Y. G. Harto Pramono.* Pengembangan Prosedur Perangkat Lunak Pembelajaran Berbantuan Komputer 123
- Umar Tirtaraharja.* Link and Match sebagai Teknologi Pemecahan Masalah Relevansi Pendidikan 132
- Waras Bin Khamdi.* Pergeseran Paradigma Penelitian Kelas: Dari Proses-produk ke Analisis Interpretatif 146
- Marzuki.* Belajar ala Scandura: Teori, Domain, Analisis, dan Prinsip Pembelajaran 154

PENELITIAN

- I Nyoman Sudana Degeng.* Pengaruh Strategi Penataan Isi Modul, Gaya Kognitif, dan Strategi Belajar Mahasiswa terhadap Perolehan Belajar 164
- Amat Mukhadis.* Pengaruh Pengorganisasian Isi Prosedural, Locus of Control, dan Bakat Berpikir Mekanik terhadap Hasil dan Transfer Belajar di Sekolah Teknologi Menengah 172
- Sri Anitah.* Penerapan Teori Elaborasi untuk Meningkatkan Perolehan Belajar Teori-Musik Dasar Mahasiswa Program D-II PGSD 185

Prosedur Pengembangan Perangkat Lunak Pembelajaran Berbantuan Komputer

Y.G. Harto Pramono

Universitas Katolik Widya Mandala Surabaya

Abstract

It is apparent that good Computer Assisted Instruction (CAI) help learners learn. However, learners do not always get benefits as much as CAI can actually offer. It is due to the fact that producing good and effective CAI software needs a careful planning and development. Otherwise, the CAI software developed is ineffective, and even useless. In developing a good and effective CAI software, there are procedures which should be done. The procedures begin with conducting preliminary planning, and are then followed by preparing the content for CAI software, designing the lesson, creating the lesson, writing documentation, and validating the lesson. Following those steps will improve the quality of CAI software.

Kata kunci: *software* PBK, prosedur pengembangan

Pembelajaran Berbantuan Komputer (PBK) atau *Computer Assisted Instruction (CAI)* kini telah diterapkan secara luas. Guru dan siswa mulai menaruh minatnya terhadap PBK, apalagi saat ini PBK didukung dengan sarana yang semakin terjangkau, yaitu harga komputer yang semakin murah sehingga hampir setiap rumah memiliki komputer. Faktor pendukung lainnya yang sangat berarti untuk pengembangan *software* PBK yaitu bertambahnya jumlah pemrogram komputer yang berkualitas sehingga mampu memproduksi *software* PBK dengan memanfaatkan kecanggihan komputer mutakhir.

Disamping faktor-faktor pendukung tersebut, pengembangan *software* PBK dipandang layak dilakukan karena pembelajaran melalui komputer dapat memberikan keuntungan-keuntungan

kepada siswa, antara lain sebagai berikut: (1) berdasarkan hasil berbagai riset mengenai keefektifan PBK, bila dirancang dengan baik PBK merupakan media pembelajaran yang sangat efektif, dapat memudahkan belajar, dan dapat meningkatkan kualitas pembelajaran; (2) pada saat dipergunakan dalam pembelajaran, komputer dapat meningkatkan motivasi belajar siswa (Higgins dalam Poedjosoedarmo, 1988); (3) Komputer, berbeda dari media lain, dapat digunakan sebagai penyampai balikan langsung (*immediate feedback*) secara sangat efektif kepada siswa. Balikan merupakan unsur penting dalam pembelajaran karena dengan balikan siswa dapat mengetahui unjuk kerjanya, apakah sudah betul atau belum. Perlunya balikan dalam pembelajaran ditekankan oleh Cates (1988:115)

"Without feedback, a learner is left to perform with no sense of direction or measure of correctness"; (4) pembelajaran berbantuan komputer sangat mendukung pembelajaran individual, di mana sistem pembelajaran tersebut dianjurkan dalam pendidikan modern (Chapelle & Jamieson, 1986). Dengan komputer siswa dapat bekerja sendiri dan sesuai dengan tingkat kemampuan serta kecepatan belajar siswa sendiri, atau menurut Kweldju (1995) komputer memiliki ciri *self-access procedure*; (5) pembelajaran berbantuan komputer melatih siswa untuk terampil memilih bagian-bagian isi pembelajaran yang dikehendaki; (6) pembelajaran berbantuan komputer memungkinkan siswa untuk lebih mengenal dan terbiasa dengan komputer yang menjadi semakin penting dalam masyarakat modern sekarang ini dan yang dapat memberikan bantuan besar dalam pelatihan di masa mendatang, serta untuk menunjang prospek karier (Dhaif, 1989); (7) komputer yang dilengkapi dengan fasilitas warna, lagu, gambar, dan grafik yang disertai animasi mampu menyajikan realisme yang menarik sehingga mendorong siswa untuk mengadakan latihan-latihan kerja; dan masih banyak lagi keuntungan-keuntungan yang dijanjikan oleh pembelajaran berbantuan komputer.

Namun demikian, pengalaman empirik menunjukkan bahwa tidak semua *software* PBK yang beredar di lapangan memiliki kualitas yang baik dan mampu memberikan keuntungan-keuntungan sebagaimana yang dijanjikan di atas. Banyak *software* PBK yang tidak efektif, dan terkesan seolah-olah dikembangkan secara serampangan—asal jadi.

Agar keuntungan-keuntungan sebagaimana diuraikan di atas dapat diwujudkan oleh *software* PBK, maka *software* PBK harus dikembangkan sebaik mungkin melalui proses pengembangan yang hati-hati, dan berdasarkan pada prinsip-prinsip pembelajaran.

Dalam tulisan ini akan disajikan prosedur pengembangan *software* pembelajaran berbantuan komputer (*software* PBK) yang dianggap dapat

meningkatkan kualitas *software* PBK. Tulisan ini diharapkan dapat membantu pengembangan *software* PBK dalam meningkatkan kualitas *software* PBK yang dihasilkan, membantu siapa saja yang ingin mencoba untuk mengembangkan *software* PBK, atau memberikan informasi kepada siapa saja yang ingin mengetahui pengembangan *software* PBK.

TEORI YANG MELANDASI PBK

Model PBK yang menonjol sekarang ini merupakan bentuk baru dari pembelajaran terprogram (*Programmed Instruction*) yang dilandasi oleh Hukum Akibat (*Law of Effect*). Hukum akibat juga sebagai landasan psikologi behavioristik. Asumsi utama yang diyakini oleh hukum ini sangatlah sederhana, yaitu: Tingkah laku yang diikuti dengan rasa senang besar kemungkinannya untuk dilakukan atau diulang lagi daripada tingkah laku yang tidak diikuti dengan rasa senang. Berdasarkan hukum akibat ini muncullah teori S-R (yang meliputi *stimulus*, *response*, dan *reinforcement*). Dalam pembelajaran yang dilandasi dengan teori S-R ini, siswa diberi pertanyaan sebagai stimulus, ia kemudian menjawab pertanyaan tersebut (atau memberikan respon). Balikan kemudian diterima atas respon yang diberikan. Teorinya adalah siswa akan memperoleh hadiah, dan pembelajaran diperkuat dengan balikan yang hanya menyatakan bahwa jawaban atau respon siswa benar.

Dengan dilandasi teori ini, maka karakteristik utama dari pembelajaran terprogram dan PBK adalah: (1) *small steps*; (2) *active responding*; dan (3) *immediate feedback* (Burke, 1982).

MENGEMBANGKAN SOFTWARE PBK

Untuk dapat menghasilkan *software* PBK yang berkualitas perlu dilakukan perencanaan dan pengembangan yang hati-hati. Prosedur pengembangan yang disarankan di bawah ini membantu pengembang dalam proses pengembangan *software* PBK langkah demi

langkah sehingga dapat dihasilkan software PBK yang berkualitas.

Perencanaan Awal

Dalam pengembangan program pembelajaran yang efektif, tidak terkecuali program pembelajaran melalui komputer, selalu diawali dengan pengidentifikasian tujuan, kebutuhan belajar, atau dalam beberapa hal, masalah-masalah yang muncul dalam pembelajaran. Sebagaimana dikemukakan oleh Hord (1984) jika kita tidak dapat menjelaskan alasan mengapa kita melakukan apa yang kita lakukan ini, berarti kita memiliki kesempatan besar untuk memboroskan waktu kita dengan percuma. Jadi, apapun tujuan itu, tujuan adalah unsur paling penting dalam mendesain pembelajaran karena semua unsur tergantung pada tujuan.

Langkah kedua dalam perencanaan awal yang perlu dilakukan adalah analisis karakteristik siswa. Karakteristik siswa yang akan menggunakan dan belajar dari materi yang dikembangkan tidak dapat diabaikan dalam rumusan tujuan. Oleh karena itu, perlu diketahui beberapa karakteristik siswa yang relevan dan kondisi dimana program yang dikembangkan akan digunakan. Sekurang-kurangnya perlu diketahui tingkatan siswa, apakah program akan digunakan di kelas, bersama dengan materi lain, atau digunakan untuk belajar mandiri.

Kemudian, perlu dipertimbangkan mengenai strategi pembelajaran. Khususnya, dalam hal ini, perlu dipilih jenis media apa yang paling cocok untuk pembelajaran yang dikembangkan. Apakah pembelajaran melalui komputer dianggap paling sesuai dengan kebutuhan? Setelah sampai pada keputusan bahwa pembelajaran melalui komputer adalah yang paling sesuai dengan kebutuhan, maka langkah berikut ini baru dapat dilakukan yaitu merencanakan dan menyusun software pembelajaran berbantuan komputer.

Perencanaan dan penyusunan software PBK mungkin dapat dilakukan sendiri tanpa bantuan

pihak lain. Jika demikian, pengembang software PBK harus memiliki tiga keterampilan berikut ini: (1) menguasai bidang studi; (2) menguasai prosedur pengembangan media; dan (3) menguasai keterampilan teknis yang diperlukan dalam pemrograman komputer serta menguasai bahasa komputer. Namun, jika pengembang tidak menguasai beberapa di antara ketiga keterampilan tersebut, maka ia dapat mencari bantuan pihak lain, atau, dengan meminjam istilah Kemp dan Dayton (1985), dapat melakukan *team approach*. Dalam *team approach*, tiga orang atau tiga kelompok dengan keterampilannya masing-masing dapat membentuk suatu tim produksi. Keterampilan mereka saling melengkapi satu sama lain.

Menyiapkan Materi Untuk Software PBK

Pada langkah ini yang perlu dipikirkan pengembang adalah bagaimana menyusun materi untuk software PBK. Ada tiga petunjuk yang dapat dipertimbangkan dalam menyusun materi software PBK, yaitu sebagai berikut.

Pemilihan Materi yang Sesuai

Dalam memilih materi yang sesuai untuk PBK, dapat dipertimbangkan rambu-rambu sebagai berikut: (a) materi harus relevan dengan tujuan; (b) materi harus cocok untuk pembelajaran melalui komputer; dalam hal ini materi harus dapat disajikan melalui simbol-simbol yang ada pada komputer; (c) materi yang dipilih hendaknya materi yang dibutuhkan banyak orang; (d) materi untuk PBK hendaknya materi yang tidak sering berubah karena materi PBK sebaiknya dapat berguna untuk selamanya; dan (e) dengan pertimbangan bahwa materi PBK akan digunakan bersama dengan materi lain yang telah ada, maka pengembang hendaknya sudah mengenali dengan baik materi yang sudah ada. Dengan demikian, materi yang dikembangkan dapat dibuat lebih bermanfaat.

Tentukan Lingkup Pembelajaran

Adalah bijaksana jika pengembang menentukan banyaknya materi sedemikian rupa sehingga dapat dipelajari dalam tempo yang wajar. Pembelajaran yang terlalu panjang dapat melelahkan dan membosankan.

Mendesain Software PBK

Setelah langkah-langkah awal dilakukan, sekarang tibalah saatnya untuk memulai mendesain software PBK. Berikut ini dipaparkan hal-hal yang perlu dilakukan dalam mendesain software PBK.

Menentukan Desain Software PBK

Di sini perlu dipilih desain software yang sesuai untuk digunakan dalam mengembangkan software PBK. Sebelum desain ditentukan, terlebih dahulu perlu dilakukan analisis tugas (*task analysis*) karena analisis tugas dapat memberikan banyak informasi yang dapat dijadikan dasar untuk memilih desain software yang sesuai. Ada tiga jenis desain yang dapat dipilih menurut Burke (1982), yaitu: *functional design*, *physical design*, dan *logical design*. *Functional design* (juga disebut tipe software PBK) berkaitan dengan fungsi pembelajaran yang dapat diberikan oleh software PBK—misalnya, apakah software PBK memperkenalkan materi baru; apakah ini merupakan media utama yang digunakan untuk menyampaikan materi; apakah software PBK ini berperan untuk melengkapi atau menguatkan tindak belajar yang telah berlangsung melalui media lain? Berkaitan dengan fungsi pembelajaran, terdapat lima jenis desain fungsional yang dapat dipertimbangkan, yaitu: *tutorial design*, *drill-and-practice design*, *problem-solving design*, *simulation design*, dan *game design* (Burke, 1992; Kemp dan Dayton, 1985). Di antara lima jenis desain tersebut pilihlah salah satu atau beberapa jenis desain untuk digunakan dalam pengembangan software PBK.

Physical design (desain fisik) suatu pembelajaran berkaitan dengan alur yang harus diikuti

siswa melalui pembelajaran. Menurut Burke (1982) dimensi inilah yang mungkin sangat menarik sebab desain ini dapat mencerminkan karakteristik dan kecanggihan teknologi komputer. Ada tiga desain fisik yang dapat dipilih, yaitu *linear design* atau *sequence structure*, *branching design* atau *choice structure*, dan *repetition design* (Hord, 1984; Kemp dan Dayton, 1985). Pilihlah salah satu atau beberapa desain tersebut untuk diterapkan dalam software PBK.

Logical design suatu pembelajaran berkaitan dengan strategi yang menstruktur cara berpikir pengembang dan memberikan pengalaman kepada siswa untuk berpikir secara logis dimana hal ini dialaminya melalui materi yang dipelajari. *Logical design* yang lazim digunakan meliputi *deduction*, *induction*, *analogy*, *EGRUL* (EG = examples diikuti RUL = rules), *RULEG* (Rules diikuti dengan Examples), dan sebagainya. Pilihlah salah satu atau beberapa desain tersebut untuk diterapkan pada software PBK.

Mengembangkan 'Flowchart'

Langkah berikutnya adalah menyiapkan representasi visual dari alur program. Teknik yang digunakan disebut *flowcharting*. *Flowcharting* sangat berguna untuk mengkomunikasikan ide pengembang kepada ahli pemrograman komputer/partner. *Flowchart* digunakan untuk menggambarkan bagian-bagian utama dari software PBK dan untuk menyusun bagian-bagian tersebut dalam urutan yang akan diikuti oleh software PBK. *Flowchart* yang lebih rinci kemudian dibuat berdasarkan *flowchart* utama ini.

Menyusun Materi Software PBK

Langkah berikutnya yaitu menyusun materi software PBK. Langkah ini dilakukan setelah perencanaan awal, penentuan materi, pemilihan desain software, dan pengembangan *flowchart* software selesai dibuat. Dalam menyusun materi ada beberapa langkah yang perlu dilakukan, yaitu:

Mengembangkan 'Screen Map'

Setelah *flowchart* selesai dikembangkan,

mulailah dengan menyusun naskah materi pada setiap *frame*. Teknik penyusunan naskah materi ini hampir sama dengan teknik mendesain *storyboard* untuk presentasi slide. Teknik ini disebut dengan istilah *screen mapping*. Pada *screen map* penyajian materi tampak sama persis seperti apa yang akan tampak pada layar monitor. Sebagaimana *frame* pada *storyboard*, *screen* pada *screen map* menampilkan apa yang akan dilihat oleh pemakai pada layar monitor, dan menunjukkan sequen secara logis mengenai *screen* yang akan tampak pada program. *Screen map* juga sangat bermanfaat untuk mengkomunikasikan ide pengembang kepada programmer atau partner kerja.

Dalam menyiapkan *screen map*, pengembang cukup menuliskan, mengetik, atau menggambar informasi yang akan muncul pada layar monitor untuk setiap *screen* mulai dari awal sampai akhir program. Teknik yang disarankan Burke (1982), Kemp dan Dayton (1985) sangat membantu dalam

penyusunan naskah pada setiap *screen* yaitu teknik *screen coding form* sebagaimana tampak pada Gambar 1. Dengan format seperti ini pengembang software PBK dapat menyusun *lay out frame* sama persis seperti apa yang akan tampak pada layar monitor.

Menulis 'Criterion Frame'

Criterion frame adalah pertanyaan yang didesain untuk mengukur kemajuan pemahaman siswa, apakah ia telah mencapai kriteria pencapaian belajar, disamping untuk mengetahui apakah ia sudah siap untuk pindah ke bagian selanjutnya. Tidak ada penyajian atau penjelasan informasi pada *criterion frame*. *Criterion frame* juga merupakan alat utama yang digunakan pengembang software PBK untuk menentukan keefektifan sequen pembelajaran.

Bukanlah tak lazim dalam penyusunan materi pembelajaran dimulai dari menuliskan semua *cri-*

SCREEN CODING FORM

Course: MAP READING
 Lesson: 2
 Screen Number: 137

YOU HAVE JUST COMPLETED LESSON TWO
 OF THE MAP READING CAOURSE

DO YOU WANT TO:

A. REVIEW THIE LESSON
 B. GO TO THE NEXT LESSON
 C. STOP

Branching Instruction: IF A - GO TO SCREEN 65
 IF B - GO TO SCREEN 138

Note: LAYOUT SHOULD BE CONSISTENT WITH THAT IN SCREEN 62

Gambar 1. Screen Coding Form (Kemp & Dayton, 1985:251)

terion frame terlebih dahulu. Penulis kemudian hanya menuliskan *teaching frame* secukupnya di depan *criterion frame*, untuk membantu siswa menjawab setiap *criterion frame* dengan benar (Burke, 1982). Contoh mengenai *criterion frame* dapat diperiksa pada Gambar 2.

Menulis 'Teaching Frame'

Teaching frame berisi informasi yang akan dipelajari siswa. Sering sekali *criterion frame* memuat informasi yang cukup sehingga siswa dapat menjawab pertanyaan tanpa melihat *frame-frame* yang lain. Hampir semua *teaching frame*

tem. Computer language (bahasa komputer) adalah seperangkat perintah yang dapat digunakan untuk menulis program, disamping berisi rumus-rumus yang mengatur bagaimana perintah-perintah tersebut harus digunakan untuk menulis program. Bahasa komputer yang lazim digunakan untuk tujuan-tujuan umum yaitu BASIC, PASCAL, dan FORTRAN.

Authoring language telah dikembangkan untuk meningkatkan efisiensi proses pemrograman. Program bahasa penulisan seperti ini memang didesain secara khusus untuk memproduksi software PBK. PILOT adalah salah satu contoh

The bag is red.

pencil	interesting
window	green
book	white
teacher	open
pen	smart
chalk	

Use the words to invent sentences like the model above.

Gambar 2. Criterion Frame

ditulis secara langsung untuk menunjang *criterion frame* berikutnya. Sebagaimana saran Burke (1982), *teaching frame* dapat berisi uraian materi saja, atau berisi uraian dan pertanyaan sebagaimana tampak pada Gambar 3 dan 4.

Menulis Program

Sampai pada langkah ini *microcomputer* masih belum disinggung-singgung. Sekaranglah saatnya untuk mulai memasukkan materi ke dalam komputer dengan menggunakan *computer language*, *authoring language*, atau *authoring sys-*

authoring language. Program bahasa seperti ini sangat lebih mudah untuk dipelajari daripada kebanyakan bahasa komputer umum, seperti BASIC. Namun, *authoring language* masih menuntut pengembang untuk memiliki pengetahuan pemrograman yang cukup banyak. Untuk mengatasi masalah ini, sejumlah *authoring system* telah dikembangkan sehingga pengembang tidak dituntut untuk memiliki pengalaman pemrograman yang banyak atau bahkan tidak dituntut untuk memiliki

THE SMALLER UNIT OF TIME
USUALLY COMES BEFORE THE LARGER.

Study the sentence below:

- The meeting is at three o'clock
(smaller)

on Monday, June 18th.
(larger) (larger)

Gambar 3. Teaching Frame (tanpa pertanyaan)

THE SMALLER UNIT OF TIME
USUALLY COMES BEFORE THE LARGER.

Rearrange the sentence below:

- I left last week, at five in the morning.

Gambar 4. Teaching Frame

kemampuan pemrograman sama sekali, sebelum ia mulai mengembangkan software PBK. Kenyataannya *authoring system* membantu lebih banyak daripada *authoring language*. Sebagaimana pengalaman penulis sendiri, dalam menyusun software PBK penulis banyak terbantu oleh *authoring system*, yaitu tanpa harus memiliki

bekal pemrograman komputer penulis telah mampu memproduksi software PBK dengan mudah dan berhasil baik.

Untuk memasukkan materi ke dalam komputer, *flowchart (detailed flowchart)* yang telah dikembangkan perlu diperiksa lagi, dan *screen map* perlu disusun dalam urutan yang benar sesuai

dengan flowchart tersebut. Kemudian informasi diketik dalam program dengan urutan sesuai dengan *screen map*.

Mengembangkan Strategi Perekaman

Banyak penulis software PBK lupa untuk merekam jawaban siswa melalui *record keeping*. Teknik perekaman hendaknya disertakan dalam software PBK karena diperlukan untuk tujuan validasi (Beebe, 1983). Untuk memvalidasi materi, penulis perlu mengetahui secara jelas frame mana yang tidak terjawab oleh siswa, dan manakah jawaban-jawaban yang salah. Teknik ini juga sangat membantu untuk mengetahui berapa lama waktu yang dibutuhkan oleh siswa untuk menjawab setiap pertanyaan, baik jawaban benar maupun salah. Jawaban salah yang dibuat dengan cepat dapat diartikan bahwa siswa yang bersangkutan sangat bingung. Sedangkan jawaban benar yang dibuat dalam waktu terlalu lama dapat diartikan bahwa uraian atau sajian informasi sebelumnya perlu diperbaiki. Perekaman seperti ini akan dibutuhkan dalam proses validasi terhadap produk jadi.

Menyusun Dokumentasi

Dokumentasi merupakan unsur terakhir pada software PBK yang sudah jadi. Dokumentasi memberikan deskripsi materi yang menyertai program dan menjelaskan tujuan program tersebut. Dengan deskripsi ini siswa dan guru dapat mengetahui bagaimana cara menjalankan program software PBK. Dokumentasi merupakan seperangkat petunjuk yang mendeskripsikan apa, bagaimana, mengapa, dan apapun juga yang perlu untuk diketahui oleh pemakai agar program dapat berjalan. Kemp dan Dayton (1985) menyarankan bahwa dokumentasi harus memuat: (1) deskripsi tentang spesifikasi komputer yang dibutuhkan oleh program sehingga software dapat dioperasikan; (2) daftar tujuan software PBK; dan (3) petunjuk tentang cara pengoperasian software PBK.

Untuk dapat menyusun dokumentasi yang

lebih mendetail, silahkan periksa petunjuk dari Burke (1982) berikut ini. Menurut Burke, dokumentasi yang baik adalah dokumentasi yang: (1) memuat informasi-informasi penting secara lengkap, misalnya, menginformasikan mengenai model komputer, ukuran *memory*, modifikasi *hardware* secara khusus, disk, dan sebagainya; (2) mudah untuk digunakan walaupun oleh pemakai yang tidak berpengalaman; (3) ditulis dalam bahasa yang sederhana, jelas, dan lengkap; (4) menginformasikan waktu rata-rata yang dibutuhkan siswa dalam menyelesaikan software ini; (5) menunjukkan sejauh mana bantuan guru diperlukan dalam pembelajaran ini; dan (6) menyertakan informasi mengenai proses pengembangan—misalnya, rasional, tujuan, analisis tugas, dan data validasi.

Sedangkan teknik penyertaan dokumentasi pada software PBK dapat dilakukan dengan dua cara, yaitu berupa materi cetakan, atau berupa file yang terpadu pada software itu sendiri.

Memvalidasi Software PBK

Software PBK belum dapat disebut sebagai software PBK sebelum divalidasi (Burke, 1982, Dick dan Carey, 1985). Memvalidasi program adalah membuktikan validitasnya secara empirik dengan melakukan evaluasi lapangan (*field-testing*) terhadap software PBK hasil pengembangan. Jadi, software PBK harus diujicobakan di lapangan dengan sampel siswa yang representatif, yaitu siswa yang dapat mewakili siswa-siswa yang akan menggunakan software tersebut. Hanya setelah divalidasi melalui serangkaian ujicoba, software hasil pengembangan ini baru dapat disebut sebagai software PBK. Proses validasi dilakukan dengan prosedur berikut: (a) mengembangkan strategi pengidentifikasian frame; (b) melaksanakan pretes dan pascates; dan (c) melaksanakan evaluasi lapangan secara bertahap (sebagai referensi lebih lanjut silakan baca Dick dan Carey, 1985).

KESIMPULAN

Memproduksi software PBK diperlukan perencanaan dan pengembangan secara hati-hati. Jika tidak, software PBK yang dikembangkan akan tidak efektif, tidak bermanfaat, dan bahkan semata-mata hanya merupakan pemborosan waktu, tenaga, dan dana dengan percuma. Dengan mengikuti prosedur sebagaimana dipaparkan dalam tulisan ini pengembang diharapkan dapat meningkatkan kualitas software PBK sehingga dapat memberikan manfaat yang berarti bagi pembelajaran.

DAFTAR PUSTAKA

- Beebe, H.B. 1983. How to Write Your Own Instruction Using a Computer Authoring System. *Instructional Innovator*, 28(6), 35 and 38.
- Burke, R. L. 1982. *Computer Assisted Instruction*. Englewood Cliffs, New Jersey: Prentice Hall, Inc.
- Cates, J.S. 1988. Delay Feedback and Cognitive Task Level in Practice Exercises. *A paper presented at the 1988 annual convention of the association for educational communications and technologies*, New Orleans, L.A., USA.
- Chapelle, C. dan Jamieson, J. 1986. Computer-assisted Language Learning as a Predictor of Success in Acquiring English as a Second Language. *TESOL Quarterly*, XX (1): 27-43.
- Dhaif, H.A. 1989. Can Computers Teach Language? *Forum*, XXVII (3): 12-15.
- Dick, W. dan Carey, L. 1985. *The Systematic Design of Instruction*. (2nd ed.) London: Scott, Foresman and Company.
- Hord, E.V. 1984. Guidelines for Designing Computer-assisted Instruction. *Instructional Innovator*, 29 (1): 19-23.
- Kemp, J.E. and Dayton, D.K. 1985. *Planning and Producing Instructional Media*. Cambridge: Harper & Row, Publishers, New York.
- Kweldju, S. 1995. Komputer sebagai Media Belajar Kosakata Bahasa Inggris bagi Anak-anak Usia Pra-sekolah Dasar. *Jurnal Teknologi Pembelajaran: Teori dan Penelitian*, 3(1-2), 37.