

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/292448816>

Pembelajaran Pembuatan Halaman Web Dinamis Tingkat Dasar di SMK

Article · July 2014

CITATIONS

0

READS

1,270

3 authors, including:



[Admaja Dwi Herlambang](#)

Brawijaya University

11 PUBLICATIONS 0 CITATIONS

SEE PROFILE

Some of the authors of this publication are also working on these related projects:



Perbandingan System Functionality, System Interactivity, dan Usability pada Instant Messaging (IM) sebagai Media Pembelajaran Sinkron [View project](#)



Learner Autonomy and Control in Online Learning [View project](#)

Pembelajaran Pembuatan Halaman *Web* Dinamis Tingkat Dasar di SMK

Admaja Dwi Herlambang, Haris Anwar Syafrudie, Eddy Sutadji

Pendidikan Kejuruan-Universitas Negeri Malang
Jl. Semarang 5 Malang. E-mail: admherlambang@gmail.com

Abstract: Basic dynamic *web* page development was a fundamental knowledge at Software Engineering Vocational High School for professional *web* development. Research aimed for describe about teaching implementation on dynamic *web* page concept introductory, technical environment preparation, simple dynamic *web* page development, dynamic *web* page function embedding, and dynamic *web* page testing. Research result showed: (1) dynamic *web* page concept introductory dominated by moderate category indicators, (2) technical environment preparation dominated by moderate category indicators, (3) simple dynamic *web* page development dominated by moderate category indicators, (4) dynamic *web* page function embedding dominated by moderate category indicators, and (5) dynamic *web* page testing dominated by moderate category indicators.

Key Words: vocational high school, software engineering, teaching, dynamic *web*

Abstrak: Pembuatan halaman *web* dinamis tingkat dasar merupakan pengetahuan dasar di Sekolah Menengah Kejuruan Kompetensi Keahlian Rekayasa Perangkat Lunak (SMK RPL) untuk pengembangan *web* profesional. Tujuan penelitian adalah untuk mendeskripsikan pelaksanaan pengenalan konsep halaman *web* dinamis, penyiapan lingkungan teknis, pembuatan halaman *web* dinamis sederhana, penambahan fungsi halaman *web* dinamis, dan pengujian halaman *web* dinamis. Hasil penelitian menunjukkan: (1) pengenalan konsep halaman *web* dinamis didominasi oleh indikator berkategori sedang, (2) penyiapan lingkungan teknis didominasi oleh indikator berkategori sedang, (3) pembuatan halaman *web* dinamis sederhana didominasi oleh indikator berkategori sedang, (4) penambahan fungsi halaman *web* dinamis didominasi oleh indikator berkategori sedang, dan (5) pengujian halaman *web* dinamis didominasi oleh indikator berkategori sedang.

Kata kunci: SMK, rekayasa perangkat lunak, pembelajaran, *web* dinamis

Kemajuan teknologi internet dan perkembangan aplikasi berbasis *web* adalah alasan utama mengapa pekerjaan pengembang *web* (*web developer*) menjadi sangat populer (Wang, 2006:211). Pengembang *web* juga dinilai mempunyai perkembangan karir ke depan yang menggembirakan dan banyak dibutuhkan oleh personal maupun korporasi (Wetfeet, 2013). *Cable News Network* mengungkapkan bahwa gaji rata-rata untuk pengembang *web* dalam skala internasional adalah \$609.00 (CNN, 2013), sedangkan untuk skala Indonesia berkisar antara Rp3.000.000,00 hingga Rp7.000.000,00 (Kelly, 2012).

SMK Kompetensi Keahlian Rekayasa Perangkat Lunak (SMK RPL) merupakan salah satu jenjang pendidikan penghasil pengembang *web* di Indonesia.

Pengembang *web* yang mampu bersaing di pasar kerja industri rekayasa perangkat lunak dapat dihasilkan apabila didukung oleh pelaksanaan pembelajaran yang mampu memfasilitasi siswa untuk belajar sedekat mungkin dengan lingkungan kerja di industri rekayasa perangkat lunak. Hal tersebut relevan dengan penelitian Sodikin (2007:65) dan Judowati (2010:59) yang menyatakan bahwa pembelajaran di kelas bertugas membekali siswa dengan berbagai keterampilan yang berguna di tempat kerja. Pelaksanaan pembelajaran di kelas memiliki peran penting untuk memfasilitasi siswa SMK RPL dalam mengembangkan keterampilan kerja sesuai dengan kaidah-kaidah pengembangan halaman *web* (*web engineering*). Pelaksanaan pembelajaran pembuatan halaman *web* di SMK RPL merupakan gejala yang perlu diungkap

karena pelaksanaan pembelajaran pembuatan halaman *web* berperan untuk menumbuhkembangkan keterampilan siswa sebagai pengembang *web*. Penelitian Walker & Browman (1999:1), Greer (2002:19), Taylor (2006:2), Wang (2006:211), dan Fabro, Almeida, & Sluzarski (2012:29) menyimpulkan bahwa pelaksanaan pembelajaran pembuatan halaman *web* secara garis besar dapat diukur melalui tujuan pembelajaran, materi pembelajaran, kegiatan pembelajaran, sarana dan prasarana pembelajaran, sumber belajar, atau penilaian pembelajaran. Hasil observasi pra penelitian menyatakan bahwa materi pembuatan halaman *web* dinamis tingkat dasar merupakan pengetahuan dasar untuk mengembangkan halaman *web* dinamis yang banyak dibutuhkan oleh korporasi. Materi pembelajaran pembuatan halaman *web* dinamis tingkat dasar ada lima, yaitu pengenalan konsep halaman *web* dinamis, penyiapan lingkungan teknis, pembuatan halaman *web* dinamis sederhana, penambahan fungsi *web* dinamis, dan pengujian halaman *web* dinamis. Materi tersebut relevan dengan deskripsi Peraturan Menteri Pendidikan Nasional Republik Indonesia Nomor 22 Tahun 2006 tentang Standar Isi SMK/MAK dan Peraturan Menteri Pendidikan Nasional Republik Indonesia Nomor 23 Tahun 2006 tentang Standar Kompetensi Lulusan SMK/MAK.

Konsep halaman *web* dinamis memiliki fokus pada pengontrolan isi *web* secara dinamis (Lecky & Thompson, 2008:21). Eksistensi halaman *web* dinamis berpusat kepada pelayanan keunikan kebutuhan dari setiap pengguna, sehingga isi yang dihasilkan setiap halaman *web* dinamis belum tentu sama untuk setiap pengguna. Eksistensi halaman *web* dinamis juga memudahkan pengelolaan sumber daya berupa isi dan kode. Halaman *web* dinamis memiliki isi yang tersimpan di dalam layanan *web* dan dihasilkan secara dinamis berdasarkan permintaan dari pengguna. Proses pembentukan isi halaman *web* dinamis terjadi pada sisi layanan dan membutuhkan kode yang khusus dijalankan pada aplikasi layanan (*server*) (Christodoulou & Papatheodorou, 2005:41). Pengelolaan kode halaman *web* dinamis mudah dilakukan karena sebagian besar sumber daya telah tersimpan di dalam basis data (Singh & Ranjan, 2010:182). Isi yang berhasil dieksekusi oleh kode akan dikirim ke pengguna dalam bentuk respon. Proses permintaan dan respon memiliki bermacam pola yang dapat dijelaskan melalui arsitektur *web* yang digunakan.

Lingkungan teknis merupakan bidang kerja yang di digunakan siswa untuk mengembangkan halaman *web* dinamis. Aplikasi utama yang dibutuhkan di dalam pembuatan halaman *web* dinamis, yaitu perangkat

layanan lokal, aplikasi klien, dan bahasa pemrograman lapisan layanan (*server side script*) (Doyle & Lopez, 2005:5). Perangkat layanan lokal merupakan aplikasi untuk menyimpan dan mensimulasikan halaman *web* pada komputer yang tidak terhubung dengan internet (Powers, 2010:11). Aplikasi klien adalah aplikasi untuk mengakses halaman *web* yang telah di simpan di perangkat layanan (Chen & Heath, 2005:78). Bahasa pemrograman layanan adalah bahasa pemrograman yang dipergunakan untuk membangun halaman *web* dari sisi *server* (Souza, Falbo, & Guizzard, 2010:310). Contoh aplikasi layanan lokal adalah XAMPP, LAMPP, atau WampServer. Contoh aplikasi aplikasi klien adalah Mozilla Firefox, Internet Explorer, atau Google Chrome. Contoh bahasa pemrograman layanan adalah JSP, CGI, atau PHP (Hadjerrouit, 2005a:117; Hauser, Olsen, & Fadel 2010:51). Simpulan yang dapat dirumuskan adalah bahwa ada tiga batasan materi yang bisa disajikan dalam penyiapan lingkungan teknis. Pertama, materi tentang menyiapkan layanan lokal yang berisi tentang instalasi Apache Web Server, aplikasi PHP (PHP Engine), instalasi layanan basis data MySQL, instalasi Wampserver pada sistem operasi Windows, instalasi LAMPP pada sistem operasi Unix, dan instalasi MAMPP pada sistem operasi Macintosh. Kedua, materi tentang pengujian *web* browser yang berisi materi tentang verifikasi layanan lokal melalui *browser*, verifikasi aplikasi PHP melalui berkas kode layanan, dan instalasi alat pengujian halaman *web* di sisi klien. Ketiga, materi tentang bahasa pemrograman layanan yang berisi materi tentang jenis pengkodean layanan, gaya penulisan kode PHP, dan penggunaan variabel, operator, dan ekspresi pada PHP.

Halaman *web* dinamis sederhana merupakan halaman *web* yang telah memiliki operasi-operasi dasar yang bersifat dinamis, yaitu berupa operasi logika dan operasi masukan sederhana. Pembuatan *web* dinamis sederhana membutuhkan pengetahuan tentang logika kontrol dan logika perulangan (Valade, 2005:9). Logika kontrol dan logika perulangan merupakan teknik untuk mengatur aliran proses komputasi dalam pembuatan halaman *web* dinamis sederhana (Nixon, 2009:61). Pemrosesan formulir juga merupakan metode masukan dasar yang harus dikuasai untuk melakukan pemrosesan informasi yang dimasukkan oleh pengguna melalui halaman *web* dinamis (Powers, 2010:103). Formulir dipergunakan untuk mengolah masukan yang diberikan oleh pengguna dengan memanfaatkan metode kirim (*post*) dan metode balasan (*get*) (Gilmore, 2008:349). Simpulan yang dapat dirumuskan adalah bahwa ada tiga batasan materi

pembelajaran yang harus disajikan agar siswa mampu membuat halaman *web* dinamis sederhana, yaitu materi tentang formulir, logika kontrol, dan logika perulangan. Formulir terdiri dari materi tentang penggunaan masukan-keluaran dan metode *post-get*. Logika kontrol terdiri dari materi tentang logika jika dan logika tukar. Logika perulangan terdiri dari materi tentang logika ketika dan logika untuk.

Fungsi (*function*) merupakan blok kode yang berisi instruksi tertentu dan dapat dieksekusi berkali-kali untuk melakukan operasi khusus. Ada dua jenis fungsi, yaitu fungsi sistem dan sistem buatan. Fungsi sistem merupakan fungsi yang sudah terintegrasi dengan PHP Engine. Fungsi buatan merupakan fungsi yang dibuat sendiri oleh penulis kode. Kegunaan utama fungsi adalah untuk mengelola sekumpulan larik (*array*) atau variabel yang berisi larik (Tatroe, McIntyre, & Lerdorf, 2013:63). Pembelajaran tentang fungsi memerlukan materi tentang larik. Larik merupakan sekumpulan variabel yang terindeks dan digabungkan menjadi satu kesatuan sehingga mudah untuk direferensikan melalui antar baris kode, fungsi, maupun halaman *web*. Larik juga dapat diartikan sebagai sebuah variabel yang memiliki banyak nilai. Batasan pengetahuan untuk mempelajari larik adalah pengetahuan tentang jenis-jenis larik, bagaimana cara membuat larik, bagaimana cara mengelola isi larik, dan dimensi pada larik (Converse, Park, & Morgan, 2004:160). Batasan konsep untuk mempelajari larik, yaitu adalah anatomi larik, pembuatan larik, pengaksesan larik, logika perulangan larik, larik multidimensi, dan manipulasi larik (Doyle, 2010:101). Batasan materi tentang larik mencakup tentang bagaimana cara menyusun larik, memodifikasi elemen larik, pemrosesan perulangan larik, fungsi dalam larik. Jenis larik ada dua, yaitu larik numerik dan larik asosiatif (Tarr, 2012:72). Fungsi larik hanya terdapat pada larik jenis larik asosiatif, sehingga dapat disimpulkan bahwa fungsi larik merupakan fungsi sistem. Simpulan mengenai batasan materi yang harus dipelajari pada bahasan larik adalah tentang penggunaan *array*, yang berisi materi tentang anatomi larik, jenis-jenis larik, pembuatan larik, pengelolaan larik, larik multidimensi, dan fungsi larik.

Materi-materi tentang fungsi akan lebih mudah dipelajari apabila materi tentang larik sudah dikuasai terlebih dahulu. Batasan materi tentang fungsi meliputi pengetahuan tentang apa itu fungsi, bagaimana cara memanggil fungsi, bagaimana cara mendefinisikan fungsi, bagaimana cara mengelola nilai balikan dari fungsi, dan eksistensi variabel di dalam fungsi (Meloni, 2012:119). Batasan materi tentang fungsi

bisa tentang pembuatan fungsi, penggunaan variabel di dalam fungsi, mereferensikan nilai kepada fungsi, mengembalikan nilai dari fungsi, dan penggunaan fungsi terdefinisi (Valade, Ballad, & Ballad, 2008:178). Batasan materi tentang fungsi bisa tentang bagaimana mendeklarasikan fungsi, pernyataan balikan, pernyataan umum dan khusus, argumen, rekursi, dan pemanggilan fungsi secara dinamis (Atkinson, 2001:81). Batasan materi tentang fungsi juga meliputi materi tentang pendefinisian dan pemanggilan fungsi, penggunaan argumen dan nilai balikan, pendefinisian variabel global dan variabel khusus, dan memasukkan fungsi yang telah didefinisikan (Vaswani, 2005:107). Simpulan mengenai batasan materi yang harus dipelajari pada bahasan fungsi adalah tentang struktur fungsi, pendeklarasian fungsi, pengelolaan nilai balik, pereferensian, fungsi rekursif, dan fungsi dinamis.

Pengujian halaman *web* dinamis adalah aktivitas untuk memeriksa kelayakan halaman *web* dinamis yang telah dikembangkan dengan menggunakan alat ukur yang dikembangkan siswa dan berbagai perangkat lunak. Pengetahuan yang harus dikuasai dalam melakukan pengujian *web*, yaitu mengenai batasan kualitas *web*, mengenai tipe-tipe kesalahan pada *web*, tahap-tahap di dalam strategi pengujian *web*, dan aspek-aspek di dalam perencanaan pengujian (Pressman & Lowe, 2009:360). Batasan tambahan mengenai pengetahuan yang perlu diketahui di dalam pengujian *web*, yaitu mengenai karakteristik kualitas *web*, mengenai motivasi diadakannya pengujian, mengenai level pengujian dan mengenai spesifikasi pengujian *web* (Steindl, Ramler, & Altamann, 2006:133). Batasan materi mengenai pengujian halaman *web*, yaitu spesifikasi kualitas *web*, strategi pengujian *web*, evaluasi usability *web*, dan alat pengujian *web* (Casteleyn, Daniel, Dolog, & Matera, 2009:256). Simpulan yang dapat dirumuskan adalah bahwa ada dua batasan materi pokok yang disajikan, yaitu kontrol kualitas halaman *web* dinamis dan manajemen pengujian halaman *web* dinamis. Materi kontrol kualitas halaman *web* dinamis terdiri dari batasan kualitas halaman *web* dan tipe-tipe kesalahan di dalam *web*. Materi manajemen pengujian halaman *web* dinamis terdiri dari strategi pengujian *web*, perencanaan pengujian *web*, dan alat pengujian *web*.

Identifikasi permasalahan utama yang disoroti di dalam penelitian adalah bagaimanakah pelaksanaan pembelajaran halaman *web* dinamis tingkat dasar di SMK RPL. Permasalahan yang teridentifikasi dibagi menjadi lima, yaitu bagaimana pelaksanaan pengenalan konsep halaman *web* dinamis; bagaimana pelaksanaan penyiapan lingkungan teknis; bagaimana

pelaksanaan pembuatan halaman *web* dinamis sederhana; bagaimana pelaksanaan penambahan fungsi halaman *web* dinamis; dan bagaimana pelaksanaan pengujian halaman *web* dinamis.

METODE

Penelitian menggunakan pendekatan kuantitatif dengan metode deskriptif jenis *expost facto*. Populasi penelitian adalah semua siswa di tiga SMK RPL Bojonegoro yang telah dikenai pelaksanaan pembelajaran *web* dinamis tingkat dasar, yaitu siswa Kelas XII Semester Genap. Jumlah populasi (N) siswa adalah 183 siswa yang tersebar di SMKN Purwosari sejumlah 60 siswa, SMKN Ngasem sejumlah 32 siswa, dan SMKN 4 Bojonegoro sejumlah 91 siswa. Ukuran sampel (n) ditentukan menggunakan formula dari Isaac dan Michael dengan taraf kepercayaan 95% dan proporsi (p) 0.5, yaitu sebesar 125 siswa. Sampel dipilih secara acak proporsional. SMKN Purwosari dipilih sebanyak 41 siswa, SMKN Ngasem dipilih sebanyak 22 siswa, dan SMKN 4 Bojonegoro dipilih sebanyak 62 siswa.

Alat ukur yang digunakan adalah angket berskala dikotomi yang telah lolos uji validitas dan reliabilitas. Angket untuk menggali data tentang pengenalan konsep halaman *web* dinamis memiliki koefisien reliabilitas (r_h) = 0.93 dengan standar *error* (s_e) = 0.35. Angket untuk menggali data tentang penyiapan lingkungan teknis memiliki r_h = 0.87 dengan s_e = 0.40. Angket untuk menggali data tentang pembuatan halaman *web* dinamis sederhana memiliki r_h = 0.88 dengan s_e = 0.39. Angket untuk menggali data tentang penambahan fungsi halaman *web* dinamis memiliki r_h = 0.87 dengan s_e = 0.41. Angket untuk menggali data tentang pengujian halaman *web* dinamis memiliki r_h = 0.87 dengan s_e = 0.43.

Deskripsi data disajikan dalam bentuk tabel. Teknik analisis deskriptif yang digunakan adalah

formula persentase. Persentase setiap indikator dihitung dengan cara menjumlahkan perolehan skor tiap butir penyusun indikator yang diperoleh dari lapangan, membandingkannya dengan skor ideal, dan dikalikan dengan 100%. Skor ideal untuk indikator berbutir satu adalah 125 dan berbutir dua adalah 250. Perolehan persentase setiap indikator dikonsultasikan dengan norma pada Tabel 1. Norma disusun berdasarkan teori model distribusi normal yang terdiri dari enam satuan simpang baku (SD). Enam satuan SD dibagi menjadi lima kelompok kategori, yaitu sangat kuat, kuat, sedang, lemah, dan sangat lemah. Norma setiap kategori didapatkan dari perhitungan persentase ideal minimum (p_{\min}), persentase ideal maksimum (p_{\max}), jangkauan (J), rerata persentase ideal (M_p), dan simpang baku ideal (SD_p). Skor p_{\min} sebesar 0, p_{\max} sebesar 100, J sebesar $100-0=100$, rerata persentase ideal (M_p) sebesar $100/2=50$, dan SD_p sebesar $100/6=16.67$. Kemudian jumlah indikator yang termasuk dalam kategori sangat kuat, kuat, sedang, lemah, atau sangat lemah dihitung untuk ditentukan berapa porsi indikator yang direkomendasikan untuk dipertahankan dan ditingkatkan.

HASIL

Pengenalan Konsep Halaman *Web* Dinamis

Deskripsi pelaksanaan pengenalan konsep halaman *web* dinamis di lapangan dapat dianalisis melalui perolehan persentase dan kategori setiap indikator. Hasil analisis pada Tabel 2 menunjukkan bahwa ada empat indikator yang berkategori kuat (30.77%), lima indikator berkategori sedang (38.46%), dan empat indikator berkategori lemah (30.77%). Kategori kuat bermakna bahwa pelaksanaan pada indikator yang berkategori kuat sebaiknya dipertahankan. Kategori sedang atau lemah bermakna bahwa pelaksanaan pada indikator yang berkategori sedang atau lemah

Tabel 1. Kategori Persentase

Formula	Persentase (%)	Kategori	Rekomendasi Pelaksanaan
$M_p + 1.5 SD_p < x \leq 100.00$	$75.01 < x \leq 100.00$	Sangat kuat	Dipertahankan
$M_p + 0.5 SD_p < x \leq M_p + 1.5 SD_p$	$58.34 < x \leq 75.01$	Kuat	Dipertahankan
$M_p - 0.5 SD_p < x \leq M_p + 0.5 SD_p$	$41.66 < x \leq 58.34$	Sedang	Ditingkatkan
$M_p - 1.5 SD_p < x \leq M_p - 0.5 SD_p$	$24.99 < x \leq 41.66$	Lemah	Ditingkatkan
$0.00 < x \leq M_p - 1.5 SD_p$	$0.00 < x \leq 24.99$	Sangat lemah	Ditingkatkan

sebaiknya ditingkatkan. Simpulan yang dapat dirumuskan adalah ada empat (30.77%) pelaksanaan indikator yang direkomendasikan untuk dipertahankan dan ada sembilan (69.23%) pelaksanaan indikator yang direkomendasikan untuk ditingkatkan.

Penyiapan Lingkungan Teknis

Tabel 3 menunjukkan deskripsi pelaksanaan penyiapan lingkungan teknis di lapangan yang dianalisis melalui perolehan persentase dan kategori setiap indikator. Analisis pada Tabel 3 menunjukkan bahwa ada satu indikator yang berkategori sangat kuat (8.33%), empat indikator berkategori kuat (33.33%), lima indikator berkategori sedang (41.67%), dan dua indikator berkategori lemah (16.67%). Kategori sangat kuat atau kuat bermakna bahwa pelaksanaan pada indikator yang berkategori sangat kuat atau kuat sebaiknya dipertahankan. Kategori sedang atau lemah bermakna bahwa pelaksanaan pada indikator yang berkategori sedang atau lemah sebaiknya ditingkatkan. Simpulan yang dapat dirumuskan adalah ada lima (41.67%) pelaksanaan indikator yang direkomendasikan untuk dipertahankan dan ada tujuh (58.33%) pe-

laksanaan indikator yang direkomendasikan untuk ditingkatkan.

Pembuatan Halaman Web Dinamis Sederhana

Deskripsi pelaksanaan pembuatan halaman *web* dinamis sederhana di lapangan dapat dianalisis melalui perolehan persentase dan kategori setiap indikator. Analisis pada Tabel 4 menunjukkan bahwa ada satu indikator yang berkategori sangat kuat (8.33%), empat indikator berkategori kuat (33.33%), lima indikator berkategori sedang (41.67%), dan dua indikator berkategori lemah (16.67%). Kategori sangat kuat atau kuat bermakna bahwa pelaksanaan pada indikator yang berkategori sangat kuat atau kuat sebaiknya dipertahankan. Kategori sedang atau lemah bermakna bahwa pelaksanaan pada indikator yang berkategori sedang atau lemah sebaiknya ditingkatkan. Simpulan yang dapat dirumuskan adalah ada lima (41.67%) pelaksanaan indikator yang direkomendasikan untuk dipertahankan dan ada tujuh (58.33%) pelaksanaan indikator yang direkomendasikan untuk ditingkatkan.

Tabel 2. Pengenalan Konsep Halaman Web Dinamis

No	Indikator	Persentase (%)	Kategori
1	Mengembangkan prototipe konsep dan kebutuhan	38.80	Lemah
2	Menganalisis komponen fragmentasi	36.80	Lemah
3	Menganalisis pengelolaan konten	39.20	Lemah
4	Membedakan bahasa pemrograman	65.20	Kuat
5	Menganalisis arsitektur halaman <i>web</i> dinamis	50.80	Sedang
6	Materi konsep halaman <i>web</i> dinamis	60.80	Kuat
7	Prasarana pengenalan konsep <i>web</i> dinamis	51.60	Sedang
8	Sarana pengenalan konsep <i>web</i> dinamis	52.80	Sedang
9	Sumber belajar pengenalan konsep <i>web</i> dinamis	46.40	Sedang
10	Prakonsepsi konsep <i>web</i> dinamis	62.80	Kuat
11	Demonstrasi prototipe konsep dan kebutuhan	64.80	Kuat
12	Konsolidasi konsep <i>web</i> dinamis	43.80	Sedang
13	Penilaian konsep halaman <i>web</i> dinamis	39.20	Lemah

Tabel 3. Penyiapan Lingkungan Teknis

No.	Indikator	Persentase (%)	Kategori
1	Mengembangkan prototipe horisontal	28.40	Lemah
2	Menginstal aplikasi layanan	47.20	Sedang
3	Mengatur <i>web browser</i>	51.60	Sedang
4	Mengatur bahasa pemrograman layanan	76.00	Sangat Kuat
5	Materi penyiapan lingkungan teknis	68.00	Kuat
6	Prasarana penyiapan lingkungan teknis	63.20	Kuat
7	Sarana penyiapan lingkungan teknis	60.00	Kuat
8	Sumber belajar penyiapan lingkungan teknis	69.20	Kuat
9	Prakonsepsi penyiapan lingkungan teknis	56.40	Sedang
10	Demonstrasi prototipe horisontal	47.20	Sedang
11	Konsolidasi penyiapan lingkungan teknis	61.20	Sedang
12	Penilaian penyiapan lingkungan teknis	26.80	Lemah

Tabel 4. Pembuatan Halaman Web Dinamis Sederhana

No.	Indikator	Persentase (%)	Kategori
1	Mengembangkan prototipe vertikal	25.20	Lemah
2	Menganalisis fungsionalitas halaman <i>web</i>	39.00	Lemah
3	Menganalisis keluaran halaman <i>web</i>	49.60	Sedang
4	Mengevaluasi <i>web</i> dinamis sederhana	44.80	Sedang
5	Materi halaman <i>web</i> sederhana	70.40	Kuat
6	Prasarana pembuatan <i>web</i> sederhana	66.40	Kuat
7	Sarana pembuatan <i>web</i> sederhana	68.00	Kuat
8	Sumber belajar pembuatan <i>web</i> sederhana	69.60	Kuat
9	Prakonsepsi pembuatan <i>web</i> sederhana	47.60	Sedang
10	Demonstrasi prototipe vertikal	43.20	Sedang
11	Konsolidasi pembuatan halaman <i>web</i>	44.00	Sedang
12	Penilaian halaman <i>web</i> dinamis sederhana	76.60	Sangat Kuat

Tabel 5. Penambahan Fungsi Halaman Web Dinamis

No.	Indikator	Persentase (%)	Kategori
1	Mengembangkan prototipe fungsional	26.30	Lemah
2	Menganalisis kompatibilitas fungsi	29.00	Lemah
3	Menganalisis keluaran fungsi	47.60	Sedang
4	Mengevaluasi fungsi	47.80	Sedang
5	Materi penambahan fungsi	69.40	Kuat
6	Prasarana penambahan fungsi	65.40	Kuat
7	Sarana penambahan fungsi	67.00	Kuat
8	Sumber belajar penambahan fungsi	63.60	Kuat
9	Prakonsepsi penambahan fungsi	44.60	Sedang
10	Demonstrasi prototipe fungsional	44.20	Sedang
11	Konsolidasi penambahan fungsi	44.00	Sedang
12	Penilaian penambahan fungsi	84.00	Sangat Kuat

Penambahan Fungsi Halaman Web Dinamis

Deskripsi pelaksanaan penambahan fungsi halaman *web* dinamis di lapangan dapat dianalisis melalui perolehan persentase dan kategori setiap indikator. Hasil analisis menunjukkan bahwa ada satu indikator yang berkategori sangat kuat (8.33%), empat indikator berkategori kuat (33.33%), lima indikator berkategori sedang (41.67%), dan dua indikator berkategori lemah (16.67%). Kategori sangat kuat atau kuat bermakna bahwa pelaksanaan pada indikator yang berkategori sangat kuat atau kuat sebaiknya dipertahankan. Kategori sedang atau lemah bermakna bahwa pelaksanaan pada indikator yang berkategori sedang atau lemah sebaiknya ditingkatkan. Simpulan yang dapat dirumuskan adalah ada lima (41.67%) pelaksanaan indikator yang direkomendasikan untuk dipertahankan dan ada tujuh (58.33%) pelaksanaan indikator yang direkomendasikan untuk ditingkatkan. Analisis tentang penambahan fungsi halaman *web* dinamis disajikan pada Tabel 5.

Pengujian Halaman Web Dinamis

Deskripsi pelaksanaan pengujian halaman *web* dinamis di lapangan dapat dianalisis melalui perolehan

persentase dan kategori setiap indikator. Analisis pada Tabel 6 menunjukkan bahwa ada tiga indikator yang berkategori kuat (25.00%), lima indikator berkategori sedang (41.67%), dan empat indikator berkategori lemah (33.33%). Kategori kuat bermakna bahwa pelaksanaan pada indikator yang berkategori kuat sebaiknya dipertahankan. Kategori sedang atau lemah bermakna bahwa pelaksanaan pada indikator yang berkategori sedang atau lemah sebaiknya ditingkatkan. Simpulan yang dapat dirumuskan adalah ada tiga (25.00%) pelaksanaan indikator yang direkomendasikan untuk dipertahankan dan ada sembilan (75.00%) pelaksanaan indikator yang direkomendasikan untuk ditingkatkan.

PEMBAHASAN

Pengenalan Konsep Halaman Web Dinamis

Pelaksanaan indikator yang harus dipertahankan adalah membedakan bahasa pemrograman, materi konsep halaman *web* dinamis, prakonsepsi konsep *web* dinamis, dan demonstrasi prototipe konsep dan kebutuhan. Pelaksanaan indikator yang harus ditingkatkan adalah mengembangkan prototipe konsep dan kebutuhan, menganalisis komponen fragmentasi,

Tabel 6. Pengujian Halaman Web Dinamis

No.	Indikator	Persentase (%)	Kategori
1	Mengembangkan alat ukur halaman <i>web</i>	25.40	Lemah
2	Menganalisis kontrol kualitas <i>web</i> dinamis	26.60	Lemah
3	Memajemen pengujian <i>web</i> dinamis	45.40	Sedang
4	Menguji <i>web</i> dinamis	47.00	Sedang
5	Materi pengujian halaman <i>web</i> dinamis	59.20	Kuat
6	Prasarana pengujian <i>web</i> dinamis	63.00	Kuat
7	Sarana pengujian <i>web</i> dinamis	66.60	Kuat
8	Sumber belajar pengujian <i>web</i> dinamis	27.00	Lemah
9	Prakonsepsi pengujian <i>web</i> dinamis	42.60	Sedang
10	Demonstrasi alat ukur halaman <i>web</i>	43.60	Sedang
11	Konsolidasi pengujian <i>web</i> dinamis	50.40	Sedang
12	Penilaian pengujian halaman <i>web</i> dinamis	25.20	Lemah

menganalisis pengelolaan konten, menganalisis arsitektur halaman *web* dinamis, prasarana pengenalan konsep halaman *web* dinamis, sarana pengenalan konsep halaman *web* dinamis, sumber belajar pengenalan konsep halaman *web* dinamis, konsolidasi konsep *web* dinamis, dan penilaian konsep *web* dinamis. Maknanya adalah pelaksanaan pengenalan konsep halaman *web* dinamis di lapangan secara garis besar dianggap memiliki konten pengetahuan yang bersifat teoretis saja dan tidak perlu ditindak lanjuti dengan skema penilaian tertentu untuk menggali sejauh mana kemampuan siswa menguasai konsep halaman *web* dinamis. Perlu beberapa teori konfirmasi untuk menjelaskan lemahnya relevansi temuan dengan disiplin ilmu rekayasa *web* (*web engineering*).

Teori yang telah dikaji dan dijadikan sebagai landasan konfirmasi temuan di lapangan menyatakan bahwa indikator primer di dalam pelaksanaan pengenalan konsep halaman *web* dinamis adalah pengembangan prototipe konsep dan kebutuhan. Identifikasi konsep dan arsitektur halaman *web* dinamis secara substansial tercantum pada pelaksanaan pengenalan konsep halaman *web* dinamis yang diekspresikan dalam bentuk pengembangan prototipe. Pengembangan prototipe merupakan bagian dari model pengembangan aplikasi *web* air terjun (*waterfall*) yang direkomendasikan untuk dikuasai oleh pengembang *web* pada tingkat (*level*) dasar (Chen & Heath, 2005:83).

Jenis prototipe yang relevan dengan karakter pelaksanaan pengenalan konsep halaman *web* dinamis adalah prototipe konsep dan kebutuhan. Pengembangan prototipe konsep dan kebutuhan merupakan tipe prototipe awal yang berisi gambaran umum *web* dinamis. Prototipe konsep memberikan kerangka konseptual mengenai strategi penyajian tampilan depan dan tampilan belakang *web* dinamis, sehingga pengguna awam mampu menterjemahkan prototipe tersebut meskipun tidak memiliki pengetahuan teknis tentang

halaman *web* dinamis secara komprehensif. Prototipe konsep dapat berbentuk deskripsi dalam bentuk teks atau diagram alir. Kegiatan pengembangan prototipe dapat digunakan sebagai bukti bahwa siswa mampu mengekspresikan konsep halaman *web* dinamis ke dalam bentuk prototipe, sehingga diharapkan siswa terbiasa mengkomunikasikan bahasa-bahasa teknis kepada konsumen awam melalui prototipe. Suh (2005:93) menyatakan “*developers have technical knowledge and skills that can bring new ideas and features to an application of which the users might never think*”.

Argumentasi yang telah dirumuskan relevan dengan penelitian Downey & Stein (2006:22), yaitu pembelajaran pengenalan pemrograman dalam pengembangan perangkat lunak harus diawali dengan pengenalan konsep desain perangkat lunak. Desain perangkat lunak berisi materi tentang bagaimana proses-proses yang terjadi di dalam perangkat lunak. Pengenalan desain perangkat lunak dalam pembelajaran pemrograman komputer akan memberikan gambaran kepada siswa mengenai bentuk perangkat lunak yang akan dikembangkan dan siswa dapat membuat desain antar muka perangkat lunak. Relevansi dengan pengenalan konsep halaman dinamis adalah siswa dikenalkan dengan desain halaman *web* dinamis melalui pengelolaan isi, bahasa pemrograman, arsitektur, dan fragmentasi komponen pada halaman *web* dinamis. Desain antar muka di dalam pengembangan halaman *web* dinamis diekspresikan ke dalam bentuk prototipe kebutuhan dan prototipe konsep.

Sekolah perlu meninjau ulang pelaksanaan pengenalan konsep halaman *web* dinamis berdasarkan beberapa indikator lain yang telah dikaji, sehingga pelaksanaannya relevan dengan tujuan pengenalan konsep halaman *web* dinamis. Tujuan pengenalan konsep halaman *web* dinamis harus relevan dengan

tujuan Standar Kompetensi Kerja Nasional Indonesia sektor Teknologi Informasi dan Komunikasi (SKKNI TIK) subsektor penulis kode program komputer dengan kode dokumen TIK.PR02.004.01. Penulis kode komputer atau siswa diharapkan memiliki kemampuan untuk merancang program yang akan dikembangkan dalam bentuk diagram dan deskripsi. Spesifikasi diagram dan deskripsi dapat diekspresikan oleh siswa dalam tagihan tugas berbentuk prototipe konsep dan prototipe kebutuhan. Implikasinya adalah perlu ada persiapan berbagai keperluan yang relevan dengan tujuan pengenalan konsep halaman *web* dinamis, seperti sumber belajar, infrastruktur, aktivitas pembelajaran, atau teknik penilaian yang hendak digunakan.

Guru dapat memilih berbagai sumber belajar untuk membantu siswa dalam mencapai tujuan belajar pengenalan konsep halaman *web* dinamis. Sumber belajar dapat berupa buku teks cetak maupun berbasis elektronik yang bersumber dari internet. Peraturan Menteri Pendidikan Nasional Republik Indonesia Nomor 2 Tahun 2008 tentang buku menyebutkan pada pasal 6 (a) bahwa buku teks merupakan acuan wajib untuk guru dan siswa. Pasal 6 (b) menyebutkan bahwa jumlah buku teks yang wajib dijadikan acuan oleh guru dan siswa adalah dua buah. Peraturan Menteri Pendidikan dan Kebudayaan Republik Indonesia Nomor 65 Tahun 2013 tentang Standar Proses menyebutkan bahwa buku teks pelajaran merupakan salah satu bentuk sumber belajar yang bisa berbentuk media cetak maupun media elektronik. Buku teks maupun halaman internet diharapkan dapat memberi siswa kesempatan untuk mengelaborasi konsep halaman *web* dinamis. Internet dapat dijadikan sumber pendukung pada pembelajaran rumpun ilmu komputer karena telah didukung oleh perkembangan teknologi *web-site* (Hazzan, Lapidot, & Ragonis, 2011:138). Akses ke buku teks dan internet sangat direkomendasikan di dalam pembelajaran rumpun teknologi ataupun pemrograman, karena akan membantu siswa menjawab rasa ingin tahu, memperoleh sosok materi yang dipelajari, dan memperkaya referensi siswa untuk memecahkan masalah secara algoritmik (Malmi & Korhonen, 2008:219; Simmons & Hawkins, 2009:64; National Research Council, 2010:38). Variasi sumber belajar akan membantu guru untuk memilih aktivitas pembelajaran yang sesuai dengan tujuan pembelajaran.

Peraturan Menteri Pendidikan dan Kebudayaan Republik Indonesia nomor 65 tahun 2013 tentang standar proses pendidikan dasar dan menengah yang menyatakan bahwa aktivitas pembelajaran harus disitua-

sikan dalam berbagai kondisi pemecahan masalah. Apabila dihubungkan dengan tujuan pengenalan konsep halaman *web* dinamis, maka aktivitas pembelajaran yang relevan adalah pembelajaran yang berorientasi pada analisis berbagai permasalahan mengenai halaman *web* dinamis dan diekspresikan dalam bentuk prototipe. Implikasinya adalah guru bertugas memilih sistem penilaian yang dapat menafsirkan pengetahuan teoretis dan praktis siswa dalam menguasai konsep halaman *web* dinamis dan pengembangan prototipe. Indikator yang dapat dijadikan sebagai indikator penilaian, yaitu pemrosesan klien, pemrosesan layanan, aliran permintaan data, lapisan klien, lapisan layanan, fragmen dasar, fragmen delta, kode pemrograman klien, dan kode pemrograman layanan.

Sekolah dapat memanfaatkan Pedoman Penilaian Ujian Praktik Kejuruan Kompetensi Keahlian Rekayasa Perangkat Lunak yang dikeluarkan oleh Direktorat Pembinaan Sekolah Menengah Kejuruan dan Peraturan Menteri Pendidikan Nasional Nomor 40 Tahun 2008 tentang standar sarana dan prasarana SMK/MAK sebagai pedoman kebutuhan sarana dan prasarana yang dapat dipersiapkan oleh sekolah untuk mendukung pelaksanaan pengenalan konsep halaman *web* dinamis. Ketersediaan sumber belajar, infrastruktur yang relevan, dan aktivitas pembelajaran yang bervariasi diharapkan dapat mendukung pencapaian tujuan pengenalan konsep halaman *web* dinamis, sehingga siswa terfasilitasi untuk menganalisis isi *web* dinamis, bahasa pemrograman halaman *web* dinamis, arsitektur halaman *web* dinamis, dan fragmentasi halaman *web* dinamis.

Proses analisis isi *web* dinamis bertujuan agar siswa memiliki konsep yang kuat tentang tata kelola isi *web* yang bersifat dinamis. Siswa diharapkan dapat memahami bahwa sifat dinamis berarti isi *web* yang ditampilkan di klien adalah respon dari aplikasi klien dan aplikasi layanan secara algoritmik atas permintaan dari pengguna. Siswa diharapkan dapat memilih dan memilah berbagai bahasa pemrograman *web* dengan mudah apabila telah memahami konsep tata kelola isi *web* dinamis.

Proses analisis bahasa pemrograman *web* dinamis memiliki sasaran agar siswa mengenali ketidak-samaan berbagai bahasa pemrograman yang dapat digunakan untuk menyusun isi *web* dinamis berdasarkan rumpun teknologi yang digunakan, baik sisi klien dan sisi layanan. Siswa diharapkan memahami bahwa rumpun teknologi sisi klien memiliki beberapa pilihan bahasa pemrograman yang bisa digunakan, semisal JavaScript. Kelebihan JavaScript daripada kode klien yang lain, yaitu kompatibel dengan berbagai aplikasi

klien, sistem operasi, dan memiliki penterjemah khusus sehingga bisa dieksekusi di berbagai perangkat bersistem operasi (Mueller, 2013:13). Siswa juga diharapkan memahami bahwa rumpun teknologi sisi layanan memiliki beberapa pilihan bahasa pemrograman yang bisa digunakan, semisal *Hypertext Pre-processor* (PHP). PHP dipilih oleh banyak *web developer* karena gratis, mudah dipelajari, terintegrasi dengan bahasa markah, kompatibel dengan berbagai sistem operasi, stabil, dan bisa dieksekusi dengan cepat (Converse, Park, & Morgan, 2004:22).

Proses analisis arsitektur halaman *web* dinamis memiliki sasaran agar siswa menguasai konsep siklus permintaan dan respon yang terjadi antara aplikasi layanan dan aplikasi klien. Siswa diharapkan mengetahui bahwa aplikasi layanan bertugas mengolah permintaan dari pengguna berupa masukan yang dikirim melalui aplikasi klien. Permintaan diolah secara algoritmik oleh aplikasi layanan. Aplikasi layanan mengeksekusi berkas yang terlibat dalam penyusunan isi untuk mengelompokkan berkas ke dalam fragmen halaman *web* (Ruping, 2009:11). Siswa diharapkan dapat mengenali bagian-bagian halaman *web* dinamis yang terfragmentasi dengan mudah setelah memahami arsitektur halaman *web* dinamis. Proses menganalisis halaman *web* dinamis dari aspek fragmentasi, tata kelola isi, bahasa pemrograman, dan arsitektur *web* dinamis akan memperkuat konsepsi siswa dalam mengembangkan prototipe halaman *web* dinamis dan turut membangun konsepsi siswa tentang sasaran pengujian halaman *web* dinamis.

Penyiapan Lingkungan Teknis

Pelaksanaan indikator yang harus dipertahankan adalah mengatur bahasa pemrograman layanan, materi penyiapan lingkungan teknis, prasarana penyiapan lingkungan teknis, sarana penyiapan lingkungan teknis, dan sumber belajar penyiapan lingkungan teknis. Pelaksanaan indikator yang harus ditingkatkan adalah mengembangkan prototipe horisontal, menginstal aplikasi layanan, mengatur *web* browser, prakonsepsi penyiapan lingkungan teknis, demonstrasi prototipe horisontal, konsolidasi penyiapan lingkungan teknis, dan penilaian penyiapan lingkungan teknis. Maknanya, temuan di lapangan menunjukkan bahwa konten pengetahuan pelaksanaan penyiapan lingkungan teknis di lapangan dianggap sebagai pengetahuan prosedural yang sederhana dan tidak memerlukan skema penilaian khusus untuk menggali sejauh mana kemampuan siswa dalam menyiapkan, mengetes, dan

melakukan *troubleshooting* kegagalan teknis berbagai perangkat.

Teori yang telah dikaji dan dijadikan sebagai landasan konfirmasi temuan di lapangan menyatakan bahwa indikator primer di dalam pelaksanaan pengenalan konsep halaman *web* dinamis adalah pengembangan prototipe horisontal. Spesifikasi alat ukur, kebutuhan perangkat keras, dan kebutuhan perangkat lunak komputer secara substansial tercantum pada konstruk penyiapan lingkungan teknis yang dapat diekspresikan dalam bentuk pengembangan prototipe horisontal.

Prototipe horisontal merupakan prototipe lanjutan dari prototipe kebutuhan. Prototipe horisontal memiliki penjelasan antar muka halaman *web* dinamis lebih jelas dan fokus pada model muka seluruh sistem yang hendak dikembangkan (*external system*). Prototipe horisontal bisa berisi proses logika tetapi tidak terlalu detail. Penjelasan mengenai sistem eksternal pada prototipe horisontal akan meningkatkan kualitas komunikasi antara pengembang *web* dan konsumen karena lebih konkret daripada prototipe kebutuhan. Sifat konkret memiliki arti bahwa prototipe horisontal mampu mendeskripsikan bagaimana tampilan sistem eksternal halaman *web* dinamis apabila dijalankan di lingkungan sesungguhnya, sehingga dibutuhkan penyiapan lingkungan teknis untuk mendemonstrasikan tampilan eksternal halaman *web* dinamis.

Kegiatan pembelajaran penyiapan lingkungan teknis disituasikan agar siswa menyiapkan sendiri lingkungan pembelajaran. Lingkungan pembelajaran pembuatan halaman *web* dinamis yang paling utama adalah aplikasi layanan *web*. Aplikasi layanan *web* memungkinkan halaman *web* yang dikembangkan oleh siswa dapat diakses melalui internet. Aplikasi layanan *web* dapat digantikan dengan simulator seperti Wampserver atau XAMPP, sehingga kendala teknis yang berhubungan dengan kualitas hubungan internet dapat diperkecil. Simulator dapat merepresentasikan kinerja aplikasi layanan *web* di internet.

Penggunaan simulator di dalam pembelajaran rumpun teknologi sangat dianjurkan dan relevan dengan penelitian yang dilakukan oleh Gilbuena & Kirsch (2012:1). Penelitian menyatakan bahwa penggunaan simulator dapat menekan gangguan teknis di lingkungan asli, menekan biaya, dan meningkatkan motivasi siswa karena gangguan teknis sangat kecil. Siswa tetap dapat merasakan bekerja di lingkungan asli karena simulator berupaya menyediakan lingkungan yang sedekat mungkin dengan kondisi lingkungan asli. Relevansi di dalam kegiatan penyiapan

lingkungan teknis pada pembuatan halaman *web* dinamis adalah bahwa aplikasi layanan seperti WampServer atau XAMPP memang telah memiliki spesifikasi yang sama dengan aplikasi layanan internet yang asli. Siswa diharapkan mampu memasang, mengenali karakteristik, dan menangani permasalahan yang berpotensi muncul di aplikasi layanan melalui simulator. Aplikasi layanan secara umum terdiri dari PHP Engine dan MySQL, sehingga bagian-bagian aplikasi layanan diharapkan tertuang di dalam isi materi pembelajaran penyiapan lingkungan teknis.

Ada tiga cakupan materi dalam pembelajaran pada materi penyiapan lingkungan teknis. Pertama, penyiapan aplikasi layanan, terdiri dari instalasi Apache, instalasi PHP Engine, instalasi MySQL, instalasi WampServer/XAMPP pada Windows, instalasi LAMPP pada Unix, dan instalasi MAMPP pada Macintosh. *Local web server* merupakan aplikasi untuk menyimpan dan mensimulasikan halaman *web* pada komputer (Powers, 2010:11). Contoh aplikasi aplikasi layanan adalah XAMPP untuk berbagai sistem operasi, LAMPP untuk sistem operasi Linux, atau WampServer untuk sistem operasi Windows. Kedua, pengujian persiapan melalui perambah *web*, terdiri dari verifikasi aplikasi layanan melalui *web browser*, verifikasi PHP Engine dengan membuka berkas PHP melalui perambah *web*, dan instalasi alat tambahan pada perambah *web* yang berfungsi sebagai *debugger* tambahan.

Perambah *web* adalah aplikasi untuk mengakses halaman *web* yang telah di simpan di aplikasi layanan (Chen & Heath, 2005:78). Contoh aplikasi perambah *web* adalah Mozilla Firefox, Internet Explorer, atau Google Chrome. Ketiga, penyiapan bahasa pemrograman layanan, terdiri dari perbandingan berbagai pengkodean layanan, gaya penulisan kode PHP, dan penggunaan variabel, operator, dan ekspresi pada PHP. Guru diharapkan dapat menyiapkan materi yang telah dideskripsikan, sehingga sajian materi dapat mendukung ketercapaian tujuan pembelajaran.

Tujuan penyiapan lingkungan teknis adalah siswa dapat mengevaluasi, mengatur, dan menguji kebutuhan teknis dalam pembuatan halaman *web* dinamis tingkat dasar yang terdiri dari aplikasi layanan lokal, perambah *web*, dan lingkungan bahasa pemrograman layanan. Siswa juga diharapkan mampu memahami bagaimana aturan dasar penulisan kode PHP seperti gaya kode (*code style*), variabel, konstanta, dan ekspresi. Mengevaluasi memiliki arti bahwa siswa memiliki mampu membuat keputusan untuk menggunakan atau mengkombinasikan berbagai perangkat keras dan lunak yang diperlukan untuk mengembangkan

web dinamis sekaligus menyempurnakan prototipe kebutuhan yang telah dikembangkan. Perangkat lunak yang dipasang adalah berupa aplikasi layanan lokal, perambah *web*, dan lingkungan kerja (*workspace*) untuk pengkodean.

Mengatur memiliki arti bahwa siswa diharapkan memiliki kemampuan untuk merangkai atau memasang berbagai perangkat lunak yang dibutuhkan sekaligus melakukan konfigurasi pada fitur perangkat lunak. Menguji memiliki arti bahwa siswa mampu melakukan diagnosa terhadap kesalahan-kesalahan konfigurasi berbagai perangkat yang dibutuhkan untuk mengembangkan halaman *web* dinamis. Beberapa perangkat lunak presentasi bisa digunakan guru untuk menunjukkan tata cara pengaturan perangkat lunak di komputer. Gambar atau video dapat ditampilkan dengan bantuan perangkat lunak presentasi untuk memperjelas demonstrasi mengenai proses-proses teknis (Barker, 2004:127). Demonstrasi adalah kegiatan wajib di dalam pembelajaran rumpun teknologi, karena demonstrasi adalah tahap siswa mengadopsi semua keterampilan teknis yang ditunjukkan oleh guru (Kolling & Barnes, 2008:26).

SKKNI TIK subsektor penulis kode komputer dengan kode dokumen TIK.PR04.002.01 menyatakan bahwa mempersiapkan lingkungan teknis merupakan elemen kompetensi yang harus dikuasai oleh pembuat halaman *web* dinamis tingkat dasar. Inti dari kriteria unjuk kerja yang diharapkan adalah siswa mampu menyiapkan berbagai perangkat keras dan perangkat lunak yang diperlukan untuk mengembangkan halaman *web* dinamis. Unjuk kerja juga dapat didokumentasikan dalam prototipe sehingga memudahkan guru dalam memilih variasi teknik penilaian. Penyiapan lingkungan teknis berisi kegiatan yang kompleks sehingga diperlukan berbagai teknik penilaian untuk mengukur ketercapaian tujuan yang telah dideskripsikan.

Teknik penilaian pembelajaran adalah cara guru untuk memperoleh data mengenai kemajuan yang dicapai oleh siswa baik selama maupun sesudah mengikuti proses pembelajaran. Peraturan Menteri Pendidikan dan Kebudayaan Republik Indonesia nomor 66 tahun 2013 tentang standar penilaian pendidikan menyatakan bahwa ruang lingkup penilaian terdiri dari pengetahuan, sikap, keterampilan, dan dipersyaratkan relevan dengan tujuan pembelajaran. Teknik penilaian menyesuaikan aspek yang dinilai. Aspek sikap dapat dilakukan dengan cara observasi, penilaian diri, penilaian antar siswa, atau jurnal. Aspek pengetahuan dapat dinilai dengan cara tes tulis, tes lisan, atau penugasan. Aspek keterampilan dapat dinilai de-

ngan cara tes praktik, proyek, atau portofolio. Teknik tes lisan, tes tertulis, uji kelayakan produk, dan tes praktik adalah beberapa teknik penilaian yang dapat digunakan oleh guru rumpun rekayasa perangkat lunak (Borstler, Nordstrom, Westin, Mostrom, & Eliasson, 2008:91; Bolanos & Sierra, 2009:237)

Guru diharapkan mampu memilih berbagai teknik penilaian yang relevan dengan tujuan penyiapan lingkungan teknis, terutama teknik penilaian yang dapat mengukur produk berupa protipe horisontal. Beberapa hal yang bisa dijadikan indikator penilaian pembelajaran adalah instalasi Apache, instalasi PHP Engine, instalasi MySQL, instalasi WampServer, instalasi LAMP, instalasi MAMP, verifikasi aplikasi klien, verifikasi aplikasi PHP, aplikasi tambahan untuk verifikasi klien, gaya penulisan kode, variabel, konstanta, dan ekspresi.

Pembuatan Halaman Web Dinamis Sederhana

Pelaksanaan indikator yang harus dipertahankan adalah materi halaman *web* sederhana, prasarana pembuatan *web* sederhana, sarana pembuatan *web* sederhana, sumber belajar pembuatan *web* sederhana, dan penilaian halaman *web* dinamis sederhana. Pelaksanaan indikator yang harus ditingkatkan adalah mengembangkan prototipe vertikal, menganalisis fungsionalitas halaman *web*, menganalisis keluaran halaman *web*, mengevaluasi *web* dinamis sederhana, prakonsepsi pembuatan *web* sederhana, demonstrasi prototipe vertikal, dan konsolidasi pembuatan halaman *web*. Maknanya adalah temuan di lapangan secara garis besar menunjukkan bahwa pelaksanaan pembuatan halaman *web* dinamis sederhana didominasi oleh aktifitas menulis kode saja tanpa mengakomodasi siswa untuk melatih keterampilan berfikir tingkat tinggi dan bertentangan dengan teori metode formal (*formal method*) yang ada di disiplin ilmu rekayasa *web* (*web engineering*).

Teori yang telah dikaji dan dijadikan sebagai landasan konfirmasi temuan di lapangan menyatakan bahwa indikator primer di dalam pelaksanaan pembuatan halaman *web* dinamis sederhana adalah pengembangan prototipe vertikal. Prototipe merupakan cetak biru sistem yang bersifat operasional dan bukti perupaan pengetahuan konseptual ke model teknis yang dapat diukur (Brinck, Gergle, & Wood, 2002:238). Kriteria sistem vertikal untuk halaman *web* dinamis secara substansial tercantum pada konstruk pembuatan halaman *web* dinamis sederhana yang dapat diekspresikan dalam bentuk pengembangan prototipe vertikal.

Prototipe vertikal merupakan gambaran sistem yang bersifat mendalam. Mendalam memiliki arti bahwa kinerja algoritma menjadi aspek utama yang harus disajikan. Pertautan antar formulir, struktur masukan, struktur proses, dan struktur keluaran membentuk fungsi dasar dan harus dideskripsikan dengan jelas. Pertautan antar fungsi dasar disebut dengan fungsi inti. Fungsi inti mempunyai sifat paralel, yaitu menyebarkan sudut pandang antar penulis kode secara kolektif. Halaman *web* dinamis dalam bentuk prototipe vertikal dapat dikatakan halaman *web* yang sudah bisa beroperasi bagi pengguna meskipun hanya memiliki fungsi inti. Fungsi inti dapat digunakan untuk menghubungkan lebih dari satu halaman *web* dinamis. Kumpulan halaman *web* dinamis yang saling terhubung dan memiliki fungsi inti disebut dengan papan alur (*storyboard*).

Kegiatan pembelajaran pembuatan halaman *web* dinamis sederhana sudah termasuk ke dalam kegiatan penulisan kode, dimana siswa diharapkan memiliki kemampuan untuk memecahkan permasalahan dengan dasar logika. Apabila keputusan yang diberikan oleh siswa memiliki nilai kebenaran, maka siswa telah melalui serangkaian proses berfikir logis dan kreatif yang disebut dengan cara berfikir algoritmik. Berfikir algoritmik merupakan kemampuan inti yang perlu dikuasai siswa dalam pembelajaran pemrograman komputer (Dagiene, Dzemyda, & Sapagovas, 2006:7).

Keterampilan berpikir algoritmik merupakan bagian keterampilan berpikir tingkat tinggi yang ada unsur aktivitas menganalisis dan mengevaluasi untuk memecahkan masalah. Kemampuan berfikir tingkat tinggi diharapkan ada pada hasil pembelajaran bidang teknologi karena merupakan indikator pencapaian kemampuan teknis tertinggi yang dibutuhkan oleh dunia kerja modern (Rojewski, 2009:23). Kemampuan memecahkan masalah merupakan bentuk kemampuan profesional yang wajib dimiliki oleh siswa agar dapat berkompetisi di pasar kerja dan direkomendasikan untuk diintegrasikan ke dalam tujuan pembelajaran (Rauner, 2012:447). Aktivitas pembelajaran pemrograman tidak berarti aktivitas siswa dititikberatkan pada aktivitas menulis kode pemrograman dalam jumlah yang banyak dan kemahiran siswa dalam menggunakan perangkat lunak. Pembelajaran pemrograman merupakan aktivitas kompleks yang mengharuskan siswa untuk memiliki keterampilan berpikir tingkat tinggi dan melibatkan kreatifitas guru dalam menyajikan berbagai permasalahan dengan tingkat kesulitan bervariasi untuk dipecahkan oleh siswa (Pedroni & Meyer, 2010:155). Pembelajaran pemro-

graman memiliki muatan pemecahan masalah dan menuntut siswa untuk dapat menggunakan logika sehingga terbentuk keterampilan berpikir algoritmik. Pembelajaran disituasikan agar siswa beraktivitas di dalam berbagai ruang permasalahan yang disajikan oleh guru dan permasalahan yang ditemukan oleh siswa sendiri selama proses pembelajaran, sehingga siswa dapat mengekspresikan pengetahuan yang didapatkan dalam bentuk prototipe. Variasi tingkat kesulitan permasalahan yang disediakan oleh guru akan meningkatkan motivasi siswa di dalam pembelajaran (Bannedsen & Caspersen, 2008b:124).

Aktivitas pembelajaran pemrograman memiliki indikator bahwa siswa difasilitasi untuk menggunakan logika. Sasaran pengukuran guru terhadap aktivitas siswa bukan pada kuantitas kode pemrograman yang dihasilkan oleh siswa, tetapi dititikberatkan pada kualitas algoritma yang disusun siswa untuk memecahkan masalah (Szlavi & Zsako, 2006:52). Indikator lebih detail mengenai kegiatan pembelajaran pemrograman, yaitu ada aktivitas untuk mengembangkan keterampilan analisis, desain, menulis kode, pengujian, mendaur ulang, dan pengambilan keputusan (Hadjerrouit, 2005b:169). Peraturan Menteri Pendidikan dan Kebudayaan Republik Indonesia nomor 65 tahun 2013 tentang standar proses pendidikan dasar dan menengah menyatakan bahwa kriteria proses pembelajaran pada gradasi keterampilan harus ada unsur menalar dan mencipta, sehingga pembelajaran rumputan pemrograman komputer dititikberatkan pada kualitas proses algoritmik yang ditampilkan oleh siswa, penampilan prototipe yang dikembangkan, dan bukan hanya sekedar menulis kode.

Tujuan pembuatan halaman *web* dinamis sederhana adalah memfasilitasi siswa untuk berpikir algoritmik dalam memecahkan masalah tentang pemrosesan formulir, logika perulangan, logika kontrol, dan membuat sebuah prototipe vertikal sebagai bukti keberhasilan penanaman konsep teoritis dan praktis yang dapat diukur oleh guru. Aktivitas pembelajaran difokuskan untuk memberikan siswa berbagai studi kasus sehingga siswa bisa mengelaborasi permasalahan dan membuat keputusan individual maupun kolektif bersama dengan siswa lain. Guru diharapkan mampu menunjukkan tahap demi tahap penulisan kode, cara membaca kode, dan menyajikan materi yang mu-takhir.

Argumentasi yang dirumuskan didukung oleh hasil penelitian yang dilakukan oleh Stephen, Franklin, Elizabeth, Jurna, & Patrick (2011:251) dan Bannedsen & Caspersen (2008a:6), yaitu pembelajaran pemrograman akan lebih baik bila didukung oleh piranti

yang bersifat visual. Visualisasi memiliki arti bahwa guru menunjukkan tahap demi tahap proses pemrograman. Siswa diharapkan dapat mengadopsi dengan mudah keterampilan pemrograman melalui visualisasi yang dilakukan oleh guru. Siswa juga harus terfasilitasi dengan materi pembelajaran yang bersifat mutakhir. Pemutakhiran materi pembelajaran pemrograman harus dilakukan karena materi pemrograman komputer selalu berkembang sesuai dengan perkembangan teknologi komputer dari waktu ke waktu. Apabila siswa tetap diberi materi pembelajaran yang bersifat monoton dari waktu ke waktu atau bahkan dari tahun ke tahun, maka secara logis dapat diputuskan bahwa materi pemrograman bersifat obsolet.

Argumentasi yang telah dirumuskan peneliti juga relevan dengan SKKNI TIK. SKKNI TIK subsektor penulis kode komputer dengan kode dokumen TIK.PR04.002.01. Kriteria unjuk kerja yang diutamakan adalah bagaimana cara manajemen berkas kode pemrograman dan penambahan bahasa markah di halaman *web* dinamis berposisi sebagai keterampilan dasar (prasyarat) yang harus dikuasai. Kriteria unjuk kerja siswa dalam pembuatan halaman *web* dinamis sederhana dapat dibuktikan dengan pengembangan prototipe jenis vertikal.

Prototipe vertikal merupakan pengembangan prototipe horisontal dimana siswa dituntut untuk menguasai bahasa markah. Prototipe vertikal merupakan gambaran sistem yang bersifat mendalam. Mendalam memiliki arti bahwa kinerja algoritma menjadi aspek utama yang harus disajikan. Pertautan antar berkas, fungsi, struktur masukan, proses, dan keluaran harus dideskripsikan dengan jelas. Makna yang dapat dirumuskan adalah guru harus cermat dalam memilih teknik penilaian dengan pertimbangan bahwa sasaran ukur pada penilaian bukan kuantitas kode, melainkan kualitas kode yang relevan untuk memecahkan permasalahan.

Hasil penelitian yang dilakukan oleh Stephen, Franklin, Elizabeth, Jurna & Patrick (2011:251), yaitu penilaian untuk pembelajaran pemrograman harus dilakukan dengan hati-hati. Sasaran penilaian bukan pada kuantitas kode yang ditulis oleh siswa atau kemahiran siswa dalam menggunakan perangkat lunak untuk menulis kode pemrograman. Sasaran penilaian terletak pada bagaimana siswa memaknai kode yang ditulis sehingga dapat memecahkan permasalahan. Makna yang dapat dirumuskan adalah sasaran penilaian guru tidak hanya kemampuan siswa dalam menulis kode pemrosesan formulir, logika kontrol, dan logika perulangan. Sasaran utama penilaian adalah bagaimana siswa mengkombinasikan dan memaknai

kode pemrosesan formulir, logika kontrol, dan logika perulangan, sehingga kode yang telah disusun siswa membentuk suatu algoritma yang dapat memecahkan permasalahan dengan tepat. Indikator yang dapat dijadikan sebagai indikator penilaian pembuatan halaman *web* dinamis sederhana, yaitu fungsi masukan dan keluaran, metode kirim, metode balasan, pernyataan jika, pernyataan ganti, pernyataan ketika, dan pernyataan untuk.

Penambahan Fungsi Halaman Web Dinamis

Pelaksanaan indikator yang harus dipertahankan adalah materi penambahan fungsi, prasarana penambahan fungsi, sarana penambahan fungsi, sumber belajar penambahan fungsi, dan penilaian penambahan fungsi. Pelaksanaan indikator yang harus ditingkatkan adalah mengembangkan prototipe fungsional, menganalisis kompatibilitas fungsi, menganalisis keluaran fungsi, mengevaluasi fungsi, prakonsepsi penambahan fungsi, demonstrasi prototipe fungsional, dan konsolidasi penambahan fungsi. Maknanya, secara garis besar temuan di lapangan menunjukkan bahwa pelaksanaan penambahan fungsi halaman *web* dinamis hanya fokus pada kegiatan menulis kode tanpa ada kegiatan yang membimbing siswa untuk melatih kemampuan berfikir tingkat tinggi. Temuan penelitian di lapangan kurang relevan dengan *frame* disiplin ilmu rekayasa *web* (*web engineering*), apabila secara teoretis diungkap, maka akan menunjukkan bahwa pelaksanaan penambahan fungsi halaman *web* dinamis identik dengan aktivitas-aktivitas yang menuntut siswa untuk terbiasa berpikir tingkat tinggi.

Teori yang telah dikaji dan dijadikan sebagai landasan konfirmasi temuan di lapangan menyatakan bahwa indikator primer di dalam pelaksanaan penambahan fungsi halaman *web* dinamis adalah pengembangan prototipe fungsional. Prototipe dapat digunakan sebagai objek pratinjau struktur aplikasi *web* secara keseluruhan sebelum benar-benar membangun aplikasi *web* pada tahap implementasi sehingga peluang kegagalan sistem bisa diperkecil (Kusmaull & Jack, 2008: 193). Kriteria fungsional isi halaman *web* dinamis secara substansial tercantum pada konstruk penambahan fungsi halaman *web* dinamis sederhana yang dapat diekspresikan dalam bentuk pengembangan prototipe fungsional. Prototipe dapat digunakan sebagai objek pratinjau struktur aplikasi *web* secara keseluruhan sebelum benar-benar membangun aplikasi *web* pada tahap implementasi sehingga peluang kegagalan sistem bisa diperkecil (Kusmaull & Jack, 2008:193).

Prototipe fungsional merupakan pengembangan dari prototipe vertikal dimana setiap navigasi halaman *web* dinamis sudah memiliki fungsi inti yang lebih konkret dan papan alur yang lebih banyak. Papan alur di dalam fungsional prototipe terdiri dari bingkai-bingkai antar muka. Prototipe fungsional terdiri dari beberapa bagian halaman *web* dinamis yang penting, langkah-langkah operasional dalam menjelajahi isi halaman *web* dinamis, memiliki bingkai antar muka dengan petunjuk yang jelas, dan bingkai yang terurut sesuai dengan kebutuhan pengguna. Prototipe fungsional merupakan cerminan asli dari sistem halaman *web* dinamis yang dikembangkan siswa di dalam kegiatan pembelajaran.

Kegiatan yang dilaksanakan dalam penambahan fungsi halaman *web* dinamis bertujuan memfasilitasi siswa untuk menganalisis struktur fungsi dan larik, menganalisis hasil eksekusi fungsi dan larik, mengevaluasi fungsi dan larik yang digunakan di berbagai halaman *web* dinamis, dan mengembangkan prototipe *web* dinamis sebagai bukti bahwa siswa telah menguasai materi tentang fungsi dan larik. Argumentasi yang telah dirumuskan peneliti relevan dengan SKKNI TIK. SKKNI TIK subsektor penulis kode komputer dengan kode dokumen TIK.PR04.002.01 menyatakan bahwa menambahkan fungsi-fungsi internal merupakan kompetensi yang harus dikuasai oleh pembuat halaman *web*. Siswa perlu didukung dengan materi-materi yang relevan sehingga siswa dapat mengelaborasi bermacam fungsi yang harus dipasang di halaman *web* dinamis. Materi yang harus dipelajari pada bahasan fungsi adalah tentang struktur fungsi, pendeklarasian fungsi, pengelolaan nilai balik, pereferensian, fungsi rekursif, dan fungsi dinamis. Siswa diharapkan juga menguasai materi tentang larik. Materi yang harus dipelajari pada bahasan larik adalah tentang penggunaan larik, yang berisi materi tentang anatomi larik, jenis-jenis larik, pembuatan larik, pengelolaan larik, larik multidimensi, dan fungsi larik dimana semua substansi materi mensituasikan siswa dalam suasana menulis kode (pemrograman).

Prinsip pembelajaran bidang teknologi adalah kebermaknaan pengetahuan dalam belajar sehingga bisa mengantarkan siswa untuk memperoleh pengalaman yang bermakna di dalam aktivitas pembelajaran (Gagnon, 2009:271). Kebermaknaan dalam pembelajaran memiliki relevansi dengan sifat pelaksanaan pembelajaran pemrograman. Sifat pelaksanaan pembelajaran pemrograman adalah mensituasikan siswa agar siswa memecah permasalahan kompleks menjadi pecahan permasalahan sederhana, menemukan solusi yang tepat atas permasalahan dengan da-

sar berpikir algoritmik, dan menterjemahkan solusi ke dalam kode pemrograman yang siap dieksekusi (Carbone, Hurst, Mitchell, & Gunstone, 2009:26). Argumentasi yang dirumuskan relevan dengan penelitian Douglas (2013:1) tentang kegiatan pembelajaran rumpun teknologi, yaitu pembelajaran akan lebih baik apabila siswa disituasikan agar dapat bekerja di dalam tim, dibiasakan saling berkomunikasi dua arah, dan berpikir tingkat tinggi sehingga mampu menemukan berbagai variasi pemecahan masalah. Relevansi penelitian dengan argumentasi yang telah dirumuskan peneliti terletak pada sifat kegiatan pemrograman dalam memecahkan masalah adalah kerja kolektif. Kerja kolektif memiliki arti bahwa solusi konseptual untuk setiap permasalahan yang muncul di dalam penulisan kode pemrograman akan lebih cepat ditemukan. Penemuan pemecahan masalah secara kolektif akan memberi kesempatan siswa untuk saling berkomunikasi dan akan mempercepat penulisan pengetahuan antar siswa. Pembelajaran penambahan fungsi *web* dinamis akan lebih baik bila disituasikan di dalam kegiatan belajar berkelompok atau tim. Tugas yang bersifat analisis dan pengembangan kode pemrograman harus diselesaikan di dalam kerja tim (Kolling, 2008:145). Kerja tim akan merangsang siswa untuk saling berkomunikasi, menghargai pendapat, dan memutuskan permasalahan berdasarkan musyawarah bersama. Umpan balik antar siswa dapat menantang siswa untuk selalu mencari solusi terbaik untuk memecahkan masalah sehingga secara tidak langsung keterampilan berpikir tingkat tinggi yang dimiliki oleh siswa akan berkembang.

Penelitian Douglas (2013:1) menyatakan bahwa penilaian yang sesuai dengan sifat pembelajaran untuk mensituasikan siswa di dalam tim, berkomunikasi, dan berfikir tingkat tinggi dapat dilakukan dengan pendekatan informal berupa pelaporan hasil belajar yang disusun oleh siswa. Relevansi dalam pembelajaran penambahan fungsi halaman *web* dinamis adalah pelaporan hasil belajar siswa dapat dikontekskan ke dalam bentuk prototipe halaman *web* dinamis. Argumentasi yang disimpulkan relevan dengan penelitian Brady, Seigel, Vosecky, & Wallace (2009:92), yaitu penilaian pada pembelajaran rumpun rekayasa perangkat lunak dititikberatkan ke produk yang dikembangkan siswa dalam bentuk prototipe. Prototipe dapat dibelah menjadi beberapa bagian dan dijadikan sebagai bahan penilaian, baik penilaian ranah pengetahuan teoretik maupun praktik.

Guru dapat melakukan banyak variasi dalam penilaian pembelajaran penambahan fungsi halaman *web* dinamis. Guru juga dapat menggolongkan ke-

mampuan siswa dalam merespon permasalahan ke dalam golongan abstrak, relasional, multistruktural, unistruktural, prastruktural, atau tidak bisa sama sekali (Shuhaida, Hamilton, & D'Souza, 2009:149; Manilla, 2007:140; Lister, Simon, Thompson, Whalley, & Prasad, 2006:119). Penggolongan kemampuan siswa dalam penilaian pembelajaran penambahan fungsi halaman *web* dinamis dapat membantu guru untuk memetakan kemampuan algoritmik siswa yang tertuang di dalam kode pemrograman yang telah tersusun. Indikator yang dapat dijadikan sebagai indikator butir penilaian yaitu, penggunaan larik, anatomi larik, jenis larik, pembuatan larik, pengelolaan larik, larik multidimensional, fungsi larik, fungsi, pendeklarasian fungsi, nilai balikan fungsi, nilai referensi, fungsi rekursif, dan fungsi dinamis.

Guru harus memiliki kemampuan untuk menggunakan berbagai media pendidikan untuk mendukung pembelajaran pemrograman (Ragonis & Levinz, 2011:114). Media pendidikan yang bisa digunakan adalah berupa *handouts* dan perangkat lunak aplikasi. *Handouts* dapat digunakan siswa sebagai pedoman untuk beraktivitas di dalam kegiatan pembelajaran (Bryan & Hayes, 2007:50). *Handouts* dapat diisi dengan beberapa studi kasus dalam bentuk kode pemrograman rumpang. Perangkat lunak presentasi dapat digunakan guru dan siswa untuk menyajikan hasil analisis atau ide di dalam pembelajaran (Marshall, 2007:93). Perangkat lunak presentasi bisa digunakan guru untuk menunjukkan demonstrasi pengkodean, cara eksekusi kode, dan hasil eksekusi kode tentang pemrosesan formulir, logika kontrol, dan logika perulangan.

Pengujian Halaman Web Dinamis

Pelaksanaan indikator yang harus dipertahankan adalah materi pengujian halaman *web* dinamis, prasarana pengujian *web* dinamis, dan sarana pengujian *web* dinamis. Pelaksanaan indikator yang harus ditingkatkan adalah mengembangkan alat ukur halaman *web*, menganalisis kontrol kualitas *web* dinamis, manajemen pengujian *web* dinamis, menguji *web* dinamis, sumber belajar pengujian *web* dinamis, prakonsepsi pengujian *web* dinamis, demonstrasi alat ukur halaman *web*, konsolidasi pengujian *web* dinamis, dan penilaian pengujian halaman *web* dinamis. Maknanya, secara garis besar temuan di lapangan menunjukkan bahwa pelaksanaan pengujian halaman *web* dinamis belum melibatkan siswa untuk mengembangkan alat ukur. Alat ukur ditentukan oleh guru secara

penuh atau pengujian dilakukan secara sepihak oleh guru. Keadaan tersebut kurang relevan dengan teori disiplin ilmu rekayasa *web* (*web engineering*) dimana setiap pengembang aplikasi *web* diwajibkan membuat alat ukur untuk menguji *web* yang dihasilkan sendiri. Pembuatan alat ukur dapat meringankan kerja pengembang untuk melakukan perawatan halaman *web* dalam skema *prerelease-release-postrelease*.

Teori yang telah dikaji dan dijadikan sebagai landasan konfirmasi temuan di lapangan menyatakan bahwa tujuan pengujian halaman *web* dinamis adalah siswa mampu mengontrol kualitas dan melakukan manajemen pengujian halaman *web* dinamis. Kontrol kualitas *web* merupakan kegiatan untuk memeriksa sistem halaman *web* dinamis secara vertikal dan fungsional. Sistem secara vertikal telah dipelajari siswa pada pelaksanaan pembuatan halaman *web* dinamis sederhana dan diekspresikan ke dalam bentuk prototipe vertikal. Sistem secara fungsional telah dipelajari siswa pada pelaksanaan penambahan fungsi halaman *web* dinamis. Siswa akan lebih mudah mengelaborasi materi kontrol kualitas *web* apabila telah memiliki pengalaman pembuatan prototipe vertikal dan fungsional.

Manajemen pengujian halaman *web* dinamis merupakan kegiatan untuk menyiapkan segala kebutuhan pengujian halaman *web* dinamis. Kebutuhan terdiri dari alat ukur, perangkat keras, dan perangkat lunak komputer yang disesuaikan dengan konsep atau arsitektur halaman *web* dinamis yang hendak diuji. Kebutuhan pengujian relevan dengan isi pelaksanaan penyediaan lingkungan teknis yang diekspresikan ke dalam bentuk prototipe horisontal. Identifikasi atau pengenalan terhadap konsep atau arsitektur halaman *web* dinamis yang hendak diuji memiliki relevansi dengan isi pelaksanaan pengenalan konsep halaman *web* dinamis yang diekspresikan ke dalam prototipe konsep dan kebutuhan. Siswa akan lebih mudah mengelaborasi manajemen pengujian halaman *web* dinamis apabila telah memiliki pengalaman membuat prototipe konsep, prototipe kebutuhan, dan prototipe horisontal.

Target kegiatan pengujian halaman *web* adalah siswa dapat berperan sebagai pihak penguji *web* dan pihak yang diuji. Pihak penguji *web* bertugas mengevaluasi dan memberikan rekomendasi terkait dengan pemutakhiran halaman *web* yang sedang diuji. Pihak yang diuji bertugas untuk menyajikan dan memutakhirkan halaman *web* dinamis yang telah dikembangkan sesuai dengan rekomendasi yang diberikan oleh pihak penguji. SKKNI TIK subsektor penulis kode komputer dengan kode dokumen TIK.PR04.002.01 menyatakan bahwa menguji halaman *web* di-

namis merupakan salah satu elemen kompetensi pembuatan halaman *web* dinamis tingkat dasar. Kriteria unjuk kerja yang diutamakan adalah penampilan halaman *web* di aplikasi klien, sehingga penilaian pengujian halaman *web* dinamis difokuskan pada kegiatan siswa dalam menguji halaman *web* dinamis dan hasil akhir pengembangan halaman *web* dinamis yang dilengkapi dengan pedoman pengujian *web*.

Argumentasi yang dirumuskan peneliti relevan dengan penelitian Edwards (2003:1), yaitu pengujian perangkat lunak akan memberikan timbal balik yang lebih konkret terhadap kemampuan pemrograman siswa. Siswa akan lebih memahami di mana saja letak kesalahan di dalam aplikasi yang telah dikembangkan melalui berbagai sistem pengujian, baik pengujian yang dilakukan sendiri atau pengujian yang dilakukan oleh pihak lain. Kritik dan saran akan membangun keterampilan siswa untuk berkomunikasi. Fokus pembelajaran pengujian perangkat lunak adalah memperoleh timbal balik (*feedback*) dan melakukan perbaikan dengan solusi yang konkret (*improvement*). Pembelajaran pengujian perangkat lunak dapat menemukan kerusakan kode sebesar 28% dari ribuan baris kode yang ditulis oleh siswa, sehingga dapat dikatakan bahwa pembelajaran pengujian perangkat lunak memberikan efisiensi yang besar dalam hal penemuan kesalahan yang dilakukan siswa dalam pengembangan perangkat lunak.

Relevansi penelitian Edwards (2003:1) dengan pembelajaran pengujian halaman *web* dinamis adalah siswa juga perlu mengetahui kualitas *web* yang telah dikembangkan dan siswa diharapkan dapat mengembangkan alat ukur untuk menguji kualitas *web* dinamis. Pengujian halaman *web* dinamis dapat dilakukan oleh guru bersama siswa. Guru menguji alat ukur yang dikembangkan oleh siswa untuk menguji halaman *web* dinamis. Siswa diminta untuk menguji halaman *web* dinamis yang dikembangkan oleh siswa yang lain dengan menggunakan alat ukur dan perangkat lunak. Tujuan pengujian halaman *web* dinamis adalah agar siswa mampu mengembangkan halaman *web* dinamis yang siap beroperasi dan siap uji yang dilengkapi dengan alat ukur. Tujuan pengembangan relevan dengan taksonomi Bloom yang diterapkan di dalam pembelajaran pemrograman, yaitu siswa diharapkan mampu menggabungkan berbagai algoritma sehingga menghasilkan sebuah solusi baru untuk memecahkan masalah (Thompson, Reilly, Whalley, Hu, & Robbins, 2008:157). Aktivitas pembelajaran pengujian halaman *web* dinamis merupakan aktivitas siswa yang berpotensi menjumpai kendala teknis. Siswa sebagai subjek belajar harus mampu mendiagnosa

dan memecahkan masalah teknis yang ditemukan selama melakukan pengujian melalui berbagai perangkat lunak maupun uji coba eksekusi kode pemrograman secara abstrak (tidak disituasikan oleh guru). Kemampuan siswa dalam memecahkan berbagai masalah teknis yang bersifat abstrak dengan pengetahuan yang telah dimiliki disebut dengan *knowledge worker* atau *abstract reasoning* (Fien & Wilson, 2009:20).

Aktivitas yang dapat terjadi di awal pembelajaran adalah guru menggugah motivasi siswa melalui tayangan konkret mengenai bagaimana uji coba terhadap halaman *web* dinamis. Tayangan konkret akan lebih meningkatkan motivasi siswa di dalam pembelajaran daripada contoh abstrak (Caspersen & Christensen, 2008:131; Lucas, Spencer, & Claxton, 2012:48). Siswa dibentuk ke dalam kelompok karena di dalam kegiatan pembelajaran akan ada interaksi antara kelompok siswa yang menjadi penguji dan siswa yang diuji. Pengelompokan siswa juga membantu guru agar siswa fokus kepada kegiatan pembelajaran (Burd, 2004:70; Setiawan, Fitrajaya, & Mardiyanti, 2010:8). Guru juga bertugas menangkap prakonsepsi siswa mengenai pengujian halaman *web* dinamis. Kegiatan prakonsepsi merupakan kegiatan guru meminta siswa untuk menyiapkan terlebih dahulu perangkat lunak apa saja yang akan diperlukan dan dipasang di perangkat keras (Zhang & Olfman, 2010:189). Siswa juga akan memerlukan bimbingan dari guru untuk mengembangkan alat ukur halaman *web* dinamis, sehingga demonstrasi dari guru tentang tata cara mengembangkan alat ukur akan membantu siswa untuk mengadopsi beberapa keterampilan yang diperlukan. Guru menjadi pemimpin skenario pengujian halaman *web* dinamis, artinya guru bertugas memeriksa kelayakan alat ukur dan menengahi siswa yang menjadi pihak penguji dan yang diuji. Siswa juga dapat diminta oleh guru untuk melaporkan hasil pengujian, membuat refleksi, dan mengerjakan beberapa tugas untuk mempertahankan pengetahuan siswa. Pemberian tugas dapat membantu guru untuk mengkonsolidasikan pengetahuan siswa yang diperoleh selama kegiatan inti (Duckett & Tatarkowski, 2005:25).

Strategi yang perlu diketahui untuk pengujian halaman *web* dinamis sama dengan strategi pengujian perangkat lunak, yaitu pengujian kotak putih dan pengujian kotak hitam. Pengujian kotak putih berasumsi bahwa penguji mengetahui struktur kode yang digunakan untuk membangun halaman *web* dinamis. Pengujian kotak hitam berasumsi bahwa penguji tidak mengetahui struktur kode, sehingga hanya menguji sebatas masukan dan keluaran yang dihasilkan oleh halaman

web. Guru dapat mengatur peran siswa dalam pengujian halaman *web* dinamis dan membagi kelompok mana saja yang termasuk penguji kotak putih dan kelompok penguji kotak hitam. Kelompok penguji kotak putih fokus pada kode pemrograman dan kelompok penguji kotak putih fokus pada kualitas eksternal halaman *web*. Joy (2004:78) menyebutkan beberapa dimensi penilaian kode pemrograman, yaitu komentar pada kode, tata letak kode, ketepatan penulisan kode, struktur kode, pengujian kode, penggunaan kode eksternal (*library*), dokumentasi desain, dokumentasi penggunaan, dokumentasi sistem, efisiensi kode, pemilihan algoritma, dan efisiensi algoritma.

Batasan materi pengujian halaman *web* dinamis ada dua, yaitu kontrol kualitas halaman *web* dinamis dan manajemen pengujian halaman *web* dinamis. Materi kontrol kualitas halaman *web* dinamis terdiri dari batasan kualitas halaman *web* dan tipe-tipe kesalahan di dalam *web*. Materi manajemen pengujian halaman *web* dinamis terdiri dari strategi pengujian *web*, perencanaan pengujian *web*, dan alat pengujian *web*. Indikator penilaian pengujian *web* dinamis, yaitu kualitas halaman *web*, tipe-tipe kesalahan, strategi pengujian, perencanaan pengujian, dan alat pengujian. Batasan materi yang jelas diharapkan dapat membantu guru untuk menentukan dimensi apa saja yang akan dinilai. Penilaian terhadap kemampuan pemrograman siswa paling tidak harus melalui dua jenis penilaian, yaitu pengetahuan teori dan pengetahuan teknis (Yehezkel & Haberman, 2010:196).

SIMPULAN DAN SARAN

Simpulan

Simpulan yang dapat dirumuskan berdasarkan hasil penelitian, yaitu pertama, pelaksanaan pengenalan konsep halaman *web* dinamis berkategori sedang dengan rentangan kategori berupa kuat, sedang, dan lemah. Konten pengetahuan dianggap bersifat teoretis saja dan tidak memerlukan skema penilaian khusus.

Kedua, pelaksanaan penyiapan lingkungan teknis berkategori sedang dengan rentangan kategori berupa sangat kuat, kuat, sedang, dan lemah. Konten pengetahuan dianggap bersifat prosedural yang sederhana dan tidak memerlukan skema penilaian khusus.

Ketiga, pelaksanaan pembuatan halaman *web* dinamis sederhana berkategori sedang dengan rentangan kategori berupa sangat kuat, kuat, sedang, dan lemah. Aktivitasnya hanya menulis kode saja tanpa mengakomodasi siswa untuk melatih keterampilan berpikir tingkat tinggi dan bertentangan dengan teori

metode formal (*formal method*) yang ada di disiplin ilmu rekayasa *web* (*web engineering*).

Keempat, pelaksanaan penambahan fungsi halaman *web* dinamis berkategori sedang dengan rentangan kategori berupa sangat kuat, kuat, sedang, dan lemah. Aktivitasnya hanya menulis kode saja tanpa mengakomodasi siswa untuk melatih keterampilan berpikir tingkat tinggi dan bertentangan dengan teori metode formal (*formal method*) yang ada di disiplin ilmu rekayasa *web* (*web engineering*).

Kelima, pelaksanaan pengujian halaman *web* dinamis berkategori sedang dengan rentangan kategori berupa kuat, sedang, dan lemah. Siswa belum dilibatkan untuk mengembangkan alat ukur kualitas halaman *web* dinamis.

Saran

Saran untuk SMK RPL, yaitu pertama, pelaksanaan pengenalan konsep halaman *web* dinamis perlu disempurnakan melalui kegiatan mengembangkan prototipe konsep dan kebutuhan, menganalisis komponen fragmentasi, menganalisis pengelolaan konten, menganalisis arsitektur halaman *web* dinamis, sarana pengenalan konsep halaman *web* dinamis, sarana pengenalan konsep halaman *web* dinamis, sumber belajar pengenalan konsep halaman *web* dinamis, konsolidasi konsep *web* dinamis, dan penilaian konsep *web* dinamis. Kedua, pelaksanaan penyiapan lingkungan teknis perlu disempurnakan melalui kegiatan mengembangkan prototipe horisontal, menginstal aplikasi layanan, mengatur *web* browser, prakonsepsi penyiapan lingkungan teknis, demonstrasi prototipe horisontal, konsolidasi penyiapan lingkungan teknis, dan penilaian penyiapan lingkungan teknis. Ketiga, pelaksanaan pembuatan halaman *web* dinamis sederhana perlu disempurnakan melalui kegiatan mengembangkan prototipe vertikal, menganalisis fungsionalitas halaman *web*, menganalisis keluaran halaman *web*, mengevaluasi *web* dinamis sederhana, prakonsepsi pembuatan *web* sederhana, demonstrasi prototipe vertikal, dan konsolidasi pembuatan halaman *web*.

Keempat, pelaksanaan penambahan fungsi halaman *web* dinamis perlu disempurnakan melalui kegiatan mengembangkan prototipe fungsional, menganalisis kompatibilitas fungsi, menganalisis keluaran fungsi, mengevaluasi fungsi, prakonsepsi penambahan fungsi, demonstrasi prototipe fungsional, dan konsolidasi penambahan fungsi. Kelima, pelaksanaan pengujian halaman *web* dinamis perlu disempurnakan melalui kegiatan mengembangkan alat ukur

halaman *web*, menganalisis kontrol kualitas *web* dinamis, manajemen pengujian *web* dinamis, menguji *web* dinamis, sumber belajar pengujian *web* dinamis, prakonsepsi pengujian *web* dinamis, demonstrasi alat ukur halaman *web*, konsolidasi pengujian *web* dinamis, dan penilaian pengujian halaman *web* dinamis. Sekolah juga diharapkan dapat melakukan pelatihan untuk meningkatkan kompetensi pedagogik dan profesional guru kelompok produktif dengan dasar disiplin ilmu rekayasa *web* (*web engineering*) sehingga dapat mengurangi peluang munculnya indikator berkategori sedang dan lemah.

Saran untuk pemerintah, yaitu perlu membuat kebijakan mengenai susunan isi pembelajaran pembuatan halaman *web* dinamis tingkat dasar, yaitu terdiri dari pengelolaan konten, arsitektur, fragmentasi komponen, bahasa pemrograman, aplikasi layanan lokal, pengujian *web browser*, penyiapan bahasa pemrograman layanan, pemrosesan formulir, logika kontrol, logika perulangan, larik, fungsi, kontrol kualitas, dan manajemen pengujian. Pemerintah juga perlu memperhatikan bahwa pelaksanaan indikator-indikator pembelajaran halaman *web* dinamis tingkat dasar di dalam penelitian menunjukkan ada yang berkategori sangat kuat dan ada yang masih berkategori lemah, sehingga pemerintah perlu mengupayakan tindak lanjut masalah yang ditemukan di dalam penelitian melalui kegiatan pengawasan atau supervisi intensif. Harapan dari pengawasan atau supervisi yang dilakukan oleh pemerintah adalah agar mengecilkan peluang kemunculan indikator-indikator yang berkategori sedang dan lemah.

Saran untuk penelitian selanjutnya perlu memperhatikan bahwa kekurangan dari penelitian ini adalah data hanya diambil dari perspektif siswa dengan menggunakan alat ukur berupa angket berskala dikotomi. Penelitian selanjutnya diharapkan dapat dilakukan dengan cara mengambil data dengan teknik observasi terstruktur, wawancara dengan berbagai pihak yang terlibat di dalam pelaksanaan pembelajaran, dan alat ukur yang digunakan berskala interval atau rasio dengan nilai standar *error* (s_e) yang relatif kecil. Penelitian lebih lanjut juga perlu dilakukan untuk mengkaji dispersi terbaru mengenai kompetensi pengembangan halaman *web* dinamis tingkat dasar, mengembangkan bagan pengetahuan baku (*body of knowledge*) tentang pengembangan halaman *web* dinamis tingkat dasar yang disepakati oleh SMK RPL dengan industri rekayasa perangkat lunak, mengembangkan berbagai metode pembelajaran yang efektif dan efisien di dalam pembelajaran *web* dinamis tingkat dasar, memetakan spesifikasi sarana dan prasarana

pendukung pembelajaran *web* dinamis tingkat dasar yang dimiliki oleh SMK RPL, mengembangkan sumber pembelajaran *web* dinamis tingkat dasar yang mudah diakses oleh siswa, dan mengembangkan berbagai teknik penilaian yang relevan dengan karakter pembelajaran *web* dinamis tingkat dasar.

DAFTARRUJUKAN

- Atkinson, L. 2001. *Core PHP Programming Using PHP to Build Dynamic Websites: Second Edition*. London: Prentice Hall.
- Bannedsen, J. & Caspersen, M. E. 2008a. Exposing the Programming Process. Dalam Bannedsen, J., Caspersen, M. E., & Kolling, M. (Eds.). *Reflections on the Teaching of Programming: Method and Implementations* (hlm. 116 - 129). Berlin: Springer.
- Bannedsen, J. & Caspersen, M. E. 2008b. Model Driven Programmings. Dalam Bannedsen, J., Caspersen, M. E., & Kolling, M. (Eds.). *Reflections on the Teaching of Programming: Method and Implementations* (hlm. 116 - 129). Berlin: Springer.
- Barker, P. 2004. Technology in Support of Learning. Dalam Baillie, C. & Moore I. (Eds.). *Effective Learning and Teaching in Engineering*. New York: Routledge Falmer.
- Bolanos, D. & Sierra, A. 2009. Integrated Software Testing Environment for Training Senior - Level Computer Science Students. Dalam Ellis, H. J. C., Demurjian, S. A., & Naveda, J. F. (Eds.). *Software Engineering: Effective Teaching and Learning Approaches and Practices* (hlm. 233 - 250). USA: IGI Global.
- Borstler, J., Nordstrom, M., Westin, L. K., Mostrom, J. E., & Eliasson, J. 2008. Transitioning to OOP/ Java - A Never Ending Story. Dalam Bannedsen, J., Caspersen, M. E., & Kolling, M. (Eds.). *Reflections on The Teaching of Programming: Methods and Implementations* (hlm. 80 - 97). New York: Springer.
- Brady, A. Seigel, M., Vosecky, T., & Wallace, C. 2009. Speaking of Software: Case Studies in Software Communication. Dalam Ellis, H. J. C., Demurjian, S. A., & Naveda J. F. (Eds.). *Software Engineering: Effective Teaching and Learning Approaches and Practices* (hlm. 75-97). USA: IGI Global.
- Brinck, T., Gergle, D., & Wood, S. D. 2002. *Designing Web Sites that Work: Usability for The Web*. USA: Academic Press.
- Bryan, J. & Hayes, D. 2007. The McDonaldization of Further Education. Dalam Hayes, D., Marshall, T., & Turner, A. (Eds.). *A Lecturer's Guide to Further Education* (hlm. 49-65). London: Open University Press.
- Burd, L. 2004. Groupwork for Computing Students. Dalam Irons, A. & Alexander, S. (Eds.). *Effective Teaching and Learning in Computing* (hlm. 69 - 75). New York: Routledge Falmer.
- Carbone, A., Hurst, J., Mitchell, I., & Gunstone, D. 2009. An Exploration of Internal Factors Influencing Student Learning of Programming. *Proceeding of the Eleventh Australasian Computing Education Conference*. 31 (5):25 – 34.
- Caspersen, M. E. & Christensen, H. B. 2008. Computer Science 1: Getting Started. Dalam Bannedsen, J., Caspersen, M. E., & Kolling, M. (Eds.). *Reflections on the Teaching of Programming: Method and Implementations* (130 - 144). Berlin: Springer.
- Casteleyn, S., Daniel, F., Dolog, P., & Matera, M. 2009. *Engineering Web Applications*. New York: Springer.
- Chen, J. Q. & Heath, R. D. 2005. *Web Application Development Technologies*. Dalam Suh, W. (Ed.). *Web Engineering: Principles and Techniques* (hlm. 76 - 96). USA: Idea Group Inc.
- Christodoulou, S. P., & Papatheodorou, T. S. 2005. *Web Engineering Resources Portal (WEP): A Reference Model and Guide*. Dalam Suh, W. (Ed.). *Web Engineering: Principles and Techniques* (hlm. 31 - 75). USA: Idea Group Inc.
- CNN. 2013. *Best Jobs For Fast Growth: Web Developer*. (Online), (<http://money.cnn.com/magazines/moneymag/best-jobs/2011/fast-growing-jobs/18.html>, diakses 1 Januari 2013).
- Converse, T., Park, J., & Morgan, C. 2004. *PHP 5 and MySQL Bible*. Canada: Willey Publishing.
- Dagiene, V., Dzemyda, G., & Sapagovas, M. 2006. Evolution of the Cultural Based Paradigm for Informatics Education. Dalam Mittermeir, R. T. (Ed.). *Informatics Education: The Bridge between Using and Understanding Computers in Secondary Schools* (hlm. 1-12). Berlin: Springer.
- Douglas, E. P. 2013. Implementation of Process Oriented Guided Learning (POGIL) in Engineering. *Advances in Engineering Education*. 3 (3): hlm. 1-16.
- Downey, A. B. & Stein, L. A. 2006. Designing a Small Footprint Curriculum in Computer Science. *Proceedings of the 36th Frontiers in Education Conference*. IEEE Computer Society: hlm. 21-26.
- Doyle, M. 2010. *Beginning PHP 5.3*. USA: Wiley Publishing, Inc.
- Duckett, I. & Tatarkowski, M. 2005. *Practical Strategies for Learning and Teaching on Vocational Programmes*. London: Learning and Skills Development Agency.

- Edwards, S. H. 2003. Improving Student Performance by Evaluating How Well Students Test Their Own Programs. *ACM: Journal of Educational Resource in Computing*, 3 (3): hlm. 1-24.
- Fabro, M. D. D., Almeida, E. C. D., & Sluzarski, F. 2012. Teaching *Web* Application Development: A Case Study in a Computer Science Course. *Informatics in Education*, 11 (1):29 – 44.
- Fien, J. & Wilson, D. 2009. Advancing Social Sustainability through Vocational Education and Training. Dalam Willis, P., McKenzie, S., & Harris, R. (Eds.). *Rethinking Work and Learning* (hlm. 13-24). Australia: Springer.
- Gagnon, R. 2009. Competency, Meaningful Learning and Learning Styles in TVET. Dalam Maclean, R. & Wilson, D. N. (Eds.). *International Handbook of Education for the Changing World of Work* (hlm. 1382 - 1392). USA: Springer.
- Gilbuena, D. M. & Kirsch, F. A. 2012. Use of an Authentic, Industrially Situated Virtual Laboratory Project to Address Engineering Design and Scientific Inquiry in High School. *Advances in Engineering Education*, 3 (2):1 – 32.
- Gilmore, W. J. 2008. *Beginning PHP and MySQL: From Novice to Professional, Third Edition*. USA: Apress.
- Greer, T. H. 2002. Critical Success Factor in Developing, Implementing, and Teaching a *Web* Development Course. *Journal of Information System Education*, 13(1):17 – 20.
- Hadjerrouit, S. 2005a. Designing a Pedagogical Model for *Web* Engineering Education: An Evolutionary Perspective. *Journal of Information Technology Education*, 4 (1):116-140.
- Hadjerrouit, S. 2005b. Object Oriented Software Development Education: A Constructivist Framework. *Informatics in Education*, 4 (2):167 - 192.
- Hauser, K., Olsen, D., & Fadel, K. 2010. An Integrated Approach to Teaching *Web* Development. *Review of Business Information System – First Quarter 20-10*, 14 (1):43 – 60.
- Hazzan, O., Lapidot, T., & Ragonis, N. 2011. *Guide to Teaching Computer Science: An Activity Based Approach*. London: Springer.
- Joy, M. 2004. Automating the Process of Skills-Based Assessment. Dalam Irons, A. & Alexander, S. (Eds.). *Effective Teaching and Learning in Computing* (hlm. 76 - 85). New York: Routledge Falmer.
- Judowati, S. A. 2010. *Keefektifan Pelaksanaan Program Unit Produksi Sekolah pada Sekolah Menengah Kejuruan di Kota Blitar*. Tesis tidak diterbitkan. Malang: Program Pascasarjana Universitas Negeri Malang.
- Kelly, W. R. 2012. *Indonesia Employment Outlook and Salary Guide 2009/2010*. (Online), (<http://www.kellyservices.co.id>, diakses 1 Mei 2012).
- Kolling, M. & Barness, D. J. 2008. Apprentice Based Learning Via Integrated Lectures and Assignments. Alam Bennedsen, J., Caspersen, M. E., & Kolling, M. (Eds.). *Reflections on the Teaching of Programming: Method and Implementations* (hlm. 17 - 29). Berlin: Springer.
- Kolling, M. 2008. Teaching Software Engineering Issues. Dalam Bennedsen, J., Caspersen, M. E., & Kolling, M. (Eds.). *Reflections on The Teaching of Programming: Methods and Implementations* (hlm. 145 - 146). New York: Springer.
- Kussmaull, C. & Jack, R. 2008. Prototyping in the *Web* Development. Dalam Brandon, D. M. (Ed.). *Software Engineering for Modern Web Applications: Methodologies and Technologies* (hlm. 191 - 206). USA: Information Science Reference.
- Lecky, G. W. & Thompson. 2008. *Just Enough Web Programming With XHTML, PHP, and MySQL*. Australia: Course Technology.
- Lister, R., Simon, B., Thompson, E., Whalley, J. L., & Prasad, C. 2006. Not Seeing the Forest for Trees: Novice Programmers and the Solo Taxonomy. *Special Interest Group on Computer Science Education Bulletin*, 38(3):118–122.
- Lucas, B., Spencer, E., & Claxton, G. 2012. *How to Teach Vocational Education: A Theory of Vocational Pedagogy*. London: City & Guilds-Center for Skills Development.
- Malmi, L. & Korhonen, A. 2008. Active Learning and Examination Methods in a Data Structures and Algorithms Course. Dalam Caspersen, E. M. & Kolling, M. (Eds.). *Reflections on the Teaching of Programming: Methods and Implementations* (hlm. 210-227). Berlin: Springer.
- Manilla, L. 2007. Novices' Progress in Introductory Programming Courses. *Informatics in Educations*, 6 (1): 139 – 152.
- Meloni, J. C. 2012. *Sams Teach Yourself PHP, MySQL, and Apache: All in One*. USA: Sams.
- Mitchell, M. L. & Jolley, J. M. 2010. *Research Design Explained: Seventh Edition*. USA: Wadsworth.
- Mueller. 2013. *HTML5 Programming with JavaScript for Dummies*. New Jersey: John Wiley & Sons, Inc.
- National Research Council. 2010. *Standards for K-12 Engineering Education*. Washington: The National Academies Press.
- Nixon, R. 2009. *Learning PHP, MySQL, & Javascript*. USA: O'Reilly.
- Pedoman Penilaian Ujian Praktik Kejuruan Kompetensi Keahlian Rekayasa Perangkat Lunak tahun 2014*.

2014. Direktorat Pembinaan Sekolah Menengah Kejuruan. (Online), (<http://ditpsmk.net>, diakses 1 Februari 2014).
- Pedroni, M. & Meyer, B. 2010. Object Oriented Modeling of Object Oriented Concepts: A Case Study in Structuring an Educational Domain. Dalam Hromkovic, J., Kralovic, R., & Vahrenhold, J. (Eds.). *Teaching Fundamental Concepts of Informatics* (hlm. 155 - 170). Berlin: Springer.
- Peraturan Menteri Pendidikan dan Kebudayaan Republik Indonesia Nomor 65 Tahun 2013 tentang Standar Proses Pendidikan Dasar dan Menengah. Badan Standar Nasional Pendidikan, (Online), (<http://bsnp-indonesia.org/>, diakses 1 Juli 2013).
- Peraturan Menteri Pendidikan dan Kebudayaan Republik Indonesia Nomor 66 Tahun 2013 tentang Standar Penilaian Pendidikan. Badan Standar Nasional Pendidikan, (Online), (<http://bsnp-indonesia.org/>, diakses 1 Juli 2013).
- Peraturan Menteri Pendidikan Nasional Republik Indonesia Nomor 2 Tahun 2008 tentang Buku. Badan Standar Nasional Pendidikan, (Online), (<http://bsnp-indonesia.org/>, diakses 1 Juli 2013).
- Peraturan Menteri Pendidikan Nasional Republik Indonesia Nomor 22 Tahun 2006 tentang Standar Isi SMK/ MAK. Badan Standar Nasional Pendidikan, (Online), (<http://bsnp-indonesia.org/>, diakses 8 Juli 2014).
- Peraturan Menteri Pendidikan Nasional Republik Indonesia Nomor 23 Tahun 2006 tentang Standar Kompetensi Lulusan SMK/ MAK. Badan Standar Nasional Pendidikan, (Online), (<http://bsnp-indonesia.org/>, diakses 8 Juli 2014).
- Peraturan Menteri Pendidikan Nasional Republik Indonesia Nomor 40 Tahun 2008 tentang Standar Sarana dan Prasarana SMK/ MAK. Badan Standar Nasional Pendidikan, (Online), (<http://bsnp-indonesia.org/>, diakses 1 Juli 2013).
- Powers, D. 2010. *PHP Solutions: Dynamics Web Design Made Easy, Second Edition*. USA: Friends of.
- Pressman, R. S., & Lowe, D. 2009. *Web Engineering: A Practitioner's Approach*. USA: Mc Graw Hill.
- Rauner, F. 2012. Demarcations between Vocational and Academic Education And How to Overcome Them. Dalam Philz, M. (Ed.). *The Future of Vocational Education and Training in a Changing World* (hlm. 433-454). Berlin: Springer.
- Rencana Pembangunan Daerah Jangka Menengah Daerah Kabupaten Bojonegoro Tahun 2008-2013. Pemerintah Kabupaten Bojonegoro, (Online), (<http://www.bojonegorokab.go.id/documents/2/renstra.pdf>, di-akses 1 Desember 2011).
- Rojewski, J. W. 2009. A Conceptual Framework for Technical and Vocational Education and Training. Dalam Maclean, R. & Wilson, D. N (Eds.). *International Handbook of Education for the Changing World of Work* (19 - 40). USA: Springer.
- Ruping, A. 2009. *Where Code and Content Meet: Design Pattern for Web Content Management and Delivery, Personalisation, and User Participation*. United Kingdom: Willey.
- Setiawan, S., Fitrajaya, E., & Mardiyanti, T. 2010. Penerapan Pengajaran Langsung (*Direct Instruction*) untuk Meningkatkan Pemahaman Belajar Siswa dalam Pembelajaran Rekayasa Perangkat Lunak (RPL). *Jurnal Pendidikan Teknologi Informasi dan Komunikasi (PTIK)*, 3(1):7-10.
- Sodikin. 2007. *Pelaksanaan Praktik Kerja Industri Sekolah Menengah Kejuruan Teknologi Industri di Kabupaten Banyumas*. Tesis tidak diterbitkan. Malang: Pascasarjana Universitas Negeri Malang.
- Shuhaida, S., Hamilton, M., & D'Souza, D. A. 2009. Taxonomic Study of Novice Programming Summative Assessment. *Proceeding of the Eleventh Australasian Computing Education Conference*, 31 (5):147-156.
- Simmons, C. & Hawkins, C. 2009. *Teaching ICT*. London: Sage Publications.
- Singh, A. & Ranjan, R. 2010. A Dynamic Web Caching Technique for Using URL Rewriting. *Journal of Physical Sciences*, 14(1):181 - 189.
- SKKNI TIK. 2014. *Lembaga Sertifikasi Profesi Telematika: SKKNI Programmer*, (Online), (<http://www.lsp-telematika.or.id>, diakses 1 Januari 2014).
- Souza, V. E. S., Falbo, A. R., & Guizzardi, G. 2010. Designing Web Information Systems for a Framework-Based Construction. Dalam Tatnall, A. (Ed.). *Web Technologies: Concept, Methodologies, Tools, and Applications* (310-343). New York: Information Science Reference.
- Steindl, C., Ramler, R., & Altmann, J. 2006. Testing Web Applications. Dalam Kappel, G, Proll, B., Reich, S., & Retschitzegger, W. (Eds.). *Web Engineering: The Discipline of Systematic Development of Web Applications* (hlm. 133 - 152). London: John Wiley & Sons, Ltd.
- Stephen, M., Franklin, W., Elizabeth, A., Juma, K., & Patrick, O. 2011. Teaching Computer Programming in 21st Century. *International Journal of Science and Technology*, 1(6):247 - 252.
- Suh, W. 2005. *Web Engineering: Principles and Techniques*. United Kingdom: Idea Group Publishing.
- Szlavi, P., & Zsako, L. 2006. Programming Versus Application. Dalam Mittermeir, R. T. (Ed.). *Informatics Education: The Bridge between Using and Understanding Computers in Secondary Schools* (hlm. 48-58). Berlin: Springer.
- Tarr, A. 2012. *PHP and MySQL: 24 Hour Trainer*. USA: John Wiley & Sons, Inc.
- Tatro, K., McIntyre, P., & Lerdorf, R. 2013. *Programming PHP*. USA: O'Reilly.
- Taylor, C. S. 2013. *Validity and Validation*. United Kingdom: Oxford Press.
- Taylor, T. 2006. *Web Competencies for IT Students*. Braun, R. (Ed.). *Proceedings 7th International Conference*

- on Information Technology Based Higher Education and Training (hlm. 563 - 570). New South West: IEEE.
- Thompson, E., Reilly, A. L., Whalley, J. L., Hu, M., & Robbins, P. 2008. Bloom's Taxonomy for Computer Science Asssment. *Conference in Research and Practice in Information Technology*, 30(5):155-162.
- Valade, J. 2005. *PHP & MySQL Everyday Apps for Dummies*. Canada: Wiley Publishing, Inc.
- Valade, J., Ballad, T., & Ballad, B. 2008. *PHP & MySQL Web Development: All in One Reference for Dummies*. Canada: Wiley Publishing, Inc.
- Vaswani, V. 2005. *How to Do Everything with PHP and MySQL*. USA: McGraw-Hill.
- Walker, E. L., & Browne, L. 1999. *Teaching Web Development with Limited Resources*. The Proceedings of the Thirtieth SIGCSE Technical Symposium on Computer Science Education, 31(1):1 – 5.
- Wang, X. 2006. A Practical Way to Teach Web Programming in Computer Science. *Journal of Computing Sciences in Colleges*, 22(1):211-220.
- Wetfeet. 2013. *Career Overview: Web Design*, (Online), (<https://www.wetfeet.com/articles/career-overview-web-design>, diakses 1 Januari 2013).
- Yehezkel, C. & Haberman, B. 2010. Long Term Development of Software Projects - Students' Self Appreciation and Expectations. Dalam Hromkovic, J., Kralovic, R., & Vahrenhold, J. (Eds.). *Teaching Fundamental Concepts of Informatics* (hlm. 194 - 206). Berlin: Springer.
- Zhang, X. S., & Olfman, L. 2010. Studios, Mini Lectures, Project Presentation. Class Blog and Wiki: A New Approach to Teaching Web Technologies. *Journal of Information Technology Education: Innovations in Practice*, 9(1):187 - 199.