

SISTEM UJIAN INTRANET DENGAN TEKNIK RANDOM MENGUNAKAN CODEIGNITER

Imam Husni Al Amin

Abstrak

Ujian merupakan salah satu cara untuk mengevaluasi proses belajar, namun faktor kecurangan masih menjadi permasalahan yang umum terjadi dalam pendidikan. Randomisasi dalam penyajian soal merupakan solusi yang sering digunakan dalam mengatasi kecurangan. Namun perlu ada metode tambahan sehingga peluang terjadi kecurangan lebih kecil yaitu dengan memunculkan beberapa soal yang berbeda tiap siswa dan klasifikasi tingkat kesulitan.

Dalam hal ini penulis mencoba menerapkan teknik semacam random matrix dimana menciptakan array bertingkat dan mengacak index element tiap level. Hasil dari penelitian ini menghasilkan perangkat lunak (software) dalam bentuk sistem informasi ujian intranet berbasis web menggunakan PHP dan MYSQL di SMK Negeri 7 Semarang.

Uji coba yang dilakukan fokus pada bagaimana soal ditampilkan berbeda – beda antar siswa tetapi memiliki bobot yang sama. Dari hasil uji coba menunjukkan sistem telah berjalan sesuai rencana.

Penggunaan RandomMatrix telah dapat menghasilkan soal acak dengan tingkat kesulitan yang sama tiap siswa. Untuk pengembangan kedepan tingkat kesulitan dapat dibuat secara fleksibel tidak hanya terdiri mudah, sedang dan sulit saja.

Kata Kunci -- *Ujian, Random soal, kecurangan siswa, Nilai ujian*

Abstrack

Exam is one way to evaluate the learning process, but cheating is still a factor common problem in education. Randomization in the presentation about a solution that is often used in dealing with fraud. But there needs to be an additional method that a smaller chance of fraud is to bring several different about each student and the difficulty level classification.

In this case the authors try to apply such techniques to create an array of random matrix where the index stratified and randomize elements of each level.

The results of this study resulted in the software (software) system in the form of intranet web-based test information using PHP and MYSQL in SMK 7 Semarang.

Experiments performed focused on how to display different matter - the difference between the student but has the same weight. From the test results show the system has been running according to plan.

The use RandomMatrix been able to generate random problems with similar levels of difficulty for each student. For the future development of the level of difficulty can be created flexibly composed not only easy, medium and difficult course.

Keywords -- *Exam, Random question, cheating students, exam scores.*

1. PENDAHULUAN

Teknologi komunikasi dan elektronik sudah berkembang sedemikian pesat, sehingga menyebabkan bidang pendidikan juga turut mengalami peningkatan dalam hal kualitas, kecepatan, kepraktisan dan juga kemudahan, ujian

konvensional pun bergeser ke arah komputerisasi, salah satu dengan adanya ujian intranet.

Ujian merupakan salah satu cara untuk mengevaluasi proses belajar. Dalam dunia pendidikan ujian dimaksudkan untuk mengukur taraf pencapaian suatu tujuan pengajaran oleh

siswa atau mahasiswa sebagai peserta didik, sehingga siswa atau mahasiswa dapat mengetahui tingkat kemampuannya dalam memahami bidang studi yang sedang ditempuh. Bila ternyata hasilnya belum maksimal, maka proses belajar harus ditingkatkan baik kualitas maupun kuantitas.

Keberadaan pengawas dan randomisasi dalam penyajian soal merupakan solusi yang sering digunakan dalam mengatasi kecurangan. Pada saat ini banyak content management system atau e-learning yang menyediakan randomisasi soal. Metode penyajian soal yang berbeda dan dapat membuat kuis yang sama muncul berbeda setiap kali ujian merupakan solusi yang bagus dalam ujian intranet namun sebagian besar sistem yang telah ada tidak mempunyai fitur pengawasan peserta ujian yang memadai, sehingga peserta masih bisa mencari celah untuk berbuat curang.

2. METODE PENELITIAN

Metode penelitian yang penulis gunakan adalah metodologi prototyping, yaitu metodologi yang menitik beratkan pada pendekatan aspek fungsi dan kepuasan user.

Sumber data dari data primer yaitu data yang paling utama, yang diperoleh secara langsung dari sumber atau obyek penelitian melalui interview dan observasi. Sumber data dari data sekunder yaitu data yang diperoleh dari literatur buku dan jurnal sebagai pelengkap untuk menganalisa masalah penelitian ini.

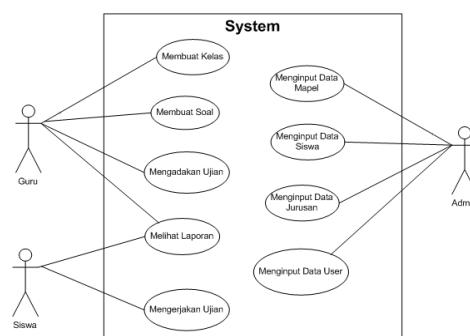
Metode pengumpulan data yang dipakai yaitu observasi, interview dan metode pustaka.

3. HASIL DAN PEMBAHASAN

3.1 Analisa dan Perancangan Sistem

Pada skripsi ini akan dikembangkan sistem ujian intranet dengan menggunakan metode klasifikasi tingkat kesulitan dengan memunculkan soal lebih sedikit dari bank soal yang dikombinasikan dengan random matrix sehingga memungkinkan siswa memperoleh soal berbeda – beda dengan

urutan yang berbeda pula tetapi memiliki tingkat kesulitan yang sama. Dengan metode seperti itu diharapkan tidak terjadi kecurangan dalam mengerjakan soal. Sistem ini dirancang untuk digunakan pada ulangan harian dalam sekolah reguler, adapun untuk mempermudah penggambaran sistem maka ditentukan implementasinya pada SMK Negeri 7 Semarang.

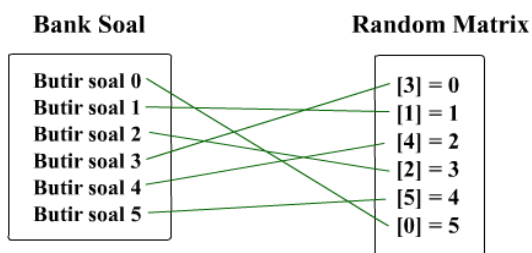


Gambar 1 Diagram UML Use Case Proses Umum

Sistem ini terdiri dari tiga buah modul program yaitu modul administrator, modul guru, dan modul siswa/peserta ujian. Modul administrator hanya dapat diakses oleh admin, administrator memiliki peranan tugas antara lain, menambah atau mengubah siswa, menambah atau mengubah jurusan, menambah atau mengubah user, menambah atau mengubah mata pelajaran. Modul guru memiliki peranan sebagai berikut: membuat atau mengubah kelas yang diampunya sendiri, membuat atau mengubah bank soal, membuat dan mengatur ujian, mengkoreksi jawaban essay, melihat laporan nilai hanya pada ujian yang diselenggarakan oleh guru yang bersangkutan saja.

3.2 Analisis Teknik dan Metode Random untuk Mengacak Soal dalam Satu Judul

Metode pengacakan yang terintegrasi dengan klasifikasi dan jumlah soal tersusun dalam pola matrix yang dipresentasikan dalam format array penulis menyebutnya Random Matrix.



Gambar 2 Pemetaan Random Matrix

Pada kotak random matrix pada gambar diatas menunjukkan index elemen tersusun secara acak. Soal akan dipetakan menurut index random matrix kemudian diurutkan menurut nilai random matrix.

Random matrix terdiri dari beberapa tingkat level array sebagai ilustrasi sebagai berikut

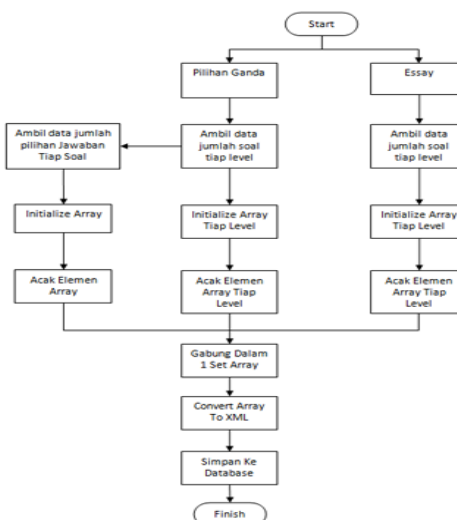
```

$result['Choice']['Easy']['IndexRandom'] = //(0 - jumlah soal level mudah);
$result['Choice']['Middle']['IndexRandom'] = //(0 - jumlah soal level sedang);
$result['Choice']['Hard']['IndexRandom'] = //(0 - jumlah soal level sulit);
$result['Choice']['Total']['IndexRandom'] = //(0 - jumlah seluruh soal);
$result['Choice']['Option']['ID']['IndexRandom'] = //(0 - jumlah pilihan gar
$result['Essay']['Easy']['IndexRandom'] = //(0 - jumlah soal level mudah);
$result['Essay']['Middle']['IndexRandom'] = //(0 - jumlah soal level sedang);
$result['Essay']['Hard']['IndexRandom'] = //(0 - jumlah soal level sulit);
$result['Essay']['Total']['IndexRandom'] = //(0 - jumlah seluruh soal);
    
```

Index random berupa angka urut mulai dari 0 sampai batas jumlah. Index random diacak sedemikian rupa kemudian soal dipetakan menurut matrix array tersebut.

3.3 Proses Random Matrix untuk Soal dalam Satu Judul

Setiap ujian yang dikerjakan setiap siswa memiliki random matrix sendiri – sendiri, random matrix disimpan di dalam database dalam bentuk XML sehingga memungkinkan soal ujian yang sudah dikerjakan dapat dibuka kembali persis ketika siswa pertama kali mengerjakannya, hal ini berguna untuk audit soal. Tujuan utama disimpan di database adalah untuk mengacak soal hanya ketika siswa mulai start ujian, jika soal diacak terus siswa akan sangat kesulitan dalam mengerjakan.

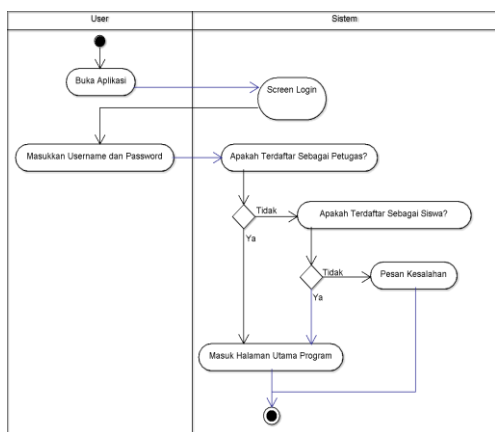


Gambar 3 Flow Chart Setup Random Matrix

Diagram diatas menjelaskan proses terbentuknya random matrix. Ketika siswa mengambil ujian salah satu objek pertama yang diproses sistem adalah membuat random matrix. Sistem mulai men-setup jumlah soal tiap level tiap jenis soal (pilihan ganda maupun essay), jumlah soal didapat dari data master soal. Tahap selanjutnya adalah meng-create array tiap level tiap jenis soal dengan elemen sebanyak jumlah soal. Create array juga dilakukan untuk pilihan jawaban pilihan ganda dengan elemen sebanyak jumlah pilihan jawaban. Setelah terbentuk array masing – masing array diacak elemennya. Array – array tersebut digabung dalam satu set array kemudian dikonversi ke dalam XML. Data XML inilah yang akan disimpan ke dalam database beserta informasi penunjang lainnya seperti id siswa, id ujian, data konfigurasi.

3.4 Perancangan Autentifikasi User

Sistem mempunyai 3 jenis user yaitu administrator, guru dan siswa. Seluruh user tersebut untuk dapat mengakses sistem diharuskan untuk melakukan login terlebih dahulu. Proses autentifikasi user dapat dilihat pada gambar berikut.

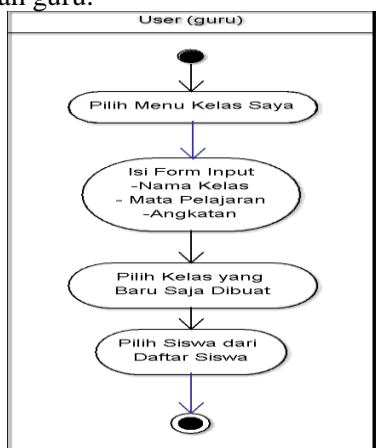


Gambar 4 Diagram UML Activity Proses Login

Untuk masuk ke dalam sistem user harus memasukkan username dan password terlebih dahulu, kemudian sistem akan memeriksa ke database apakah data yang dimasukkan adalah valid. Dari gambar diatas terdapat dua kali pengecekan yaitu Apakah terdaftar sebagai petugas dan Apakah terdaftar sebagai siswa, petugas disini maksudnya admin atau guru, hal ini dikarenakan antara petugas dan siswa memiliki tabel database terpisah. Jika terdaftar dalam database maka sistem akan menampilkan halaman utama program sesuai dengan jenis user masing – masing.

3.5 Perancangan Kelas

Otoritas yang berhak dalam hal ini adalah guru.



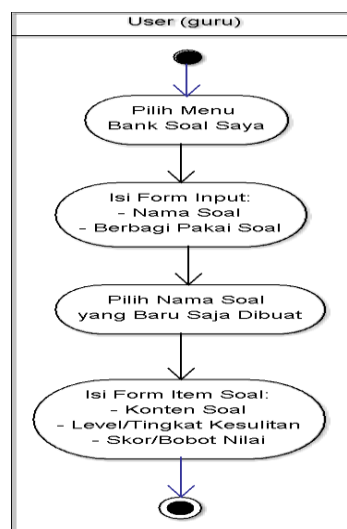
Gambar 5 Diagram UML Activity Membuat Kelas Baru

Pada halaman guru terdapat menu Kelas Saya, ketika guru ingin membuat kelas baru akan ditampilkan Form Input

Kelas, guru diharuskan memasukkan nama kelas, memilih mata pelajaran, memilih tahun angkatan. Ketika di-submit sistem akan mengecek validitas data yang dimasukkan jika data tidak valid akan muncul warning pada form input kelas, jika valid akan disimpan ke dalam database kemudian kelas baru tersebut langsung ditampilkan pada daftar kelas. Kelas yang baru dibuat tersebut masih kosong belum ada siswanya maka langkah selanjutnya guru meng-klik link siswa pada kelas tersebut sistem akan menampilkan data master siswa, pilih siswa yang akan dimasukkan ke kelas tersebut dan sistem akan meng-update ke database.

3.6 Perancangan Bank Soal

Otoritas yang berhak dalam hal ini adalah guru.



Gambar 6 Diagram UML Activity Proses Pembuatan Soal

Pada halaman guru terdapat menu Bank Soal Saya, ketika guru ingin membuat soal baru akan ditampilkan Form Input Nama Soal. Isi nama soal, pilih apakah soal boleh dipakai oleh guru lain. Ketika di-submit sistem akan cek validitas data yang dimasukkan. Validitas meliputi Nama soal harus diisi, jika data tidak valid akan muncul warning pada Form Input Nama Soal, jika valid akan disimpan ke dalam database kemudian soal baru tersebut langsung ditampilkan pada daftar soal.

Soal tersebut masih kosong belum berisi butir – butir soal maka langkah selanjutnya guru meng-klik link Item Soal pada soal tersebut sistem akan menampilkan Form Input Soal. Isi uraian soal, level/tingkat kesulitan, skor/bobot nilai. Ketika di-submit sistem akan cek validitas data yang dimasukkan. Validitas meliputi Uraian Soal, Level, Skor tidak boleh kosong, skor harus diisi angka. Jika data tidak valid akan muncul warning pada form input soal, jika valid akan disimpan ke dalam database. Pada soal pilihan ganda bedanya terdapat tombol Edit pada daftar soal, tombol Edit tersebut untuk menambah, mengubah, menghapus pilihan jawaban dan menentukan pilihan jawaban.

3.7 Perancangan Pembuatan Ujian

Otoritas yang berhak dalam hal ini adalah guru.



Gambar 7 Diagram UML Activity Membuat Ujian Baru

Pada halaman guru terdapat menu Manajemen Ujian Saya, ketika guru ingin membuat ujian baru akan ditampilkan Form Input Ujian, guru diharuskan memasukkan nama ujian, tanggal penutupan ujian. Ketika di-submit/next sistem akan mengecek validitas data yang dimasukkan jika data tidak valid akan muncul warning pada Form Input Ujian, jika valid akan disimpan ke dalam database kemudian guru langsung disodorkan daftar kelas untuk dipilih kelas yang akan mengikuti ujian. Klik Next untuk tahap selanjutnya yaitu memilih soal yang akan diujikan, Finish

untuk mengakhiri proses dan menuju ke halaman utama manajemen ujian.

3.8 Implementasi Teknik Random

Ketika siswa mulai start ujian sistem akan meng-create session menginisiasi skor, dan membuat random matrix. Berikut ini cuplikan source code random matrix untuk soal essay.

```

public function setRandomMatrixEssay($EID){
    $result = array();
    $num = array();
    $num['Easy'] = 0;
    $num['Middle'] = 0;
    $num['Hard'] = 0;
    $ds = StoreProc::SelectQuestznEssayCountLevelByExamID($EID);
    foreach($ds as $dr){
        $key = trim(ucwords($dr['Level']));
        $num[$key] = $dr['Num'];
    }
    $num['Total'] = $num['Easy'] + $num['Middle'] + $num['Hard'];
    $range['Easy'] = array();
    $range['Middle'] = array();
    $range['Hard'] = array();
    $range['Total'] = array();
    if($num['Easy'] > 0){
        $range['Easy'] = range(0, $num['Easy'] - 1);
    }
    if($num['Middle'] > 0){
        $range['Middle'] = range(0, $num['Middle'] - 1);
    }
    if($num['Hard'] > 0){
        $range['Hard'] = range(0, $num['Hard'] - 1);
    }
    if($num['Total'] > 0){
        $range['Total'] = range(0, $num['Total'] - 1);
    }
    shuffle($range['Easy']);
    shuffle($range['Middle']);
    shuffle($range['Hard']);
    shuffle($range['Total']);
    return $range;
}
    
```

Gambar 8 Source Code Random Matrix

Mula – mula variabel ds pada baris 228 mengambil data jumlah soal tiap level kemudian nilainya dimasukkan ke variabel num yang sudah diinisiasi sebelumnya di baris ke 223 – 226. Pada baris ke 240 – 251 menunjukkan sistem mulai meng-create elemen – elemen array dengan jumlah elemen sesuai dengan jumlah soal tiap level mengacu pada variabel num. Setelah terbentuk tiap level soal di random elemennya menggunakan fungsi shuffle. Pengacakan pada soal pilihan ganda dan pilihan jawaban memiliki metode proses yang sama dengan soal essay.

Penjelasan diatas baru membuat random matrix, soal akan diambil dari database kemudian ditampilkan sesuai urutan yang sudah dibuat sebelumnya, berikut cuplikan source code untuk menyajikan soal dalam hal ini soal berupa essay.

```

60
61 $i = 0;
62 foreach($data as $key => $val){
63     if($data[$key]['SumHard'] == $i AND $data[$key]['SumHard'] != '100%'){
64         break;
65     }
66     $randKey = $data[$key];
67     foreach($randKey as $key2 => $val2){
68         if($data[$key2]['SumLevel'] == 'yes'){
69             $data[$key][$key2] = $data[$key2];
70             $level[$key2] = $data[$key2];
71         }
72     }
73     $i++;
74 }
75
76 $dsMerge = array_merge_recursive($dsEasy, $dsMiddle, $dsHard);
77
78 return $dsMerge;
79
80
81 $i = 0;
82 foreach($data as $key => $val){
83     if($data[$key]['SumHard'] == $i AND $data[$key]['SumHard'] != '100%'){
84         break;
85     }
86     $randKey = $data[$key];
87     foreach($randKey as $key2 => $val2){
88         if($data[$key2]['SumLevel'] == 'yes'){
89             $data[$key][$key2] = $data[$key2];
90             $level[$key2] = $data[$key2];
91         }
92     }
93     $i++;
94 }
95
96 $dsMerge = array_merge_recursive($dsEasy, $dsMiddle, $dsHard);
97
98 return $dsMerge;

```

Gambar 9 Source Code Penyusun Soal

Mula – mula sistem mengambil data soal dari database diimplementasikan pada baris 25-27, baris tersebut mengambil data soal per level direpresentasikan dalam bentuk array tingkat 2. Baris 30-43 adalah proses memetakan antara data soal level mudah dengan random matrix. Baris 46-59 memetakan antara data soal sedang dengan random matrix. Baris 62-75 memetakan antara data soal sulit dengan random matrix. Kemudian ketiga proses tadi digabung dalam satu variable ditunjukkan pada baris ke 77 dengan nama variable dsMerge. Proses selanjutnya tinggal dirender ke dalam view.

3.9 Ruang Lingkup Uji Coba

Uji coba yang dilakukan fokus pada bagaimana soal ditampilkan berbeda – beda antar siswa tetapi memiliki bobot yang sama. Guru menyediakan enam belas butir soal tetapi dikonfigurasi hanya sepuluh soal saja yang diberikan ke siswa, dari sepuluh soal tersebut diambilkan dari enam belas soal tadi secara acak sehingga memungkinkan tiap siswa memperoleh beberapa soal yang berbeda dan dikonfigurasi pilihan ganda juga diacak.

Lebih detailnya lagi dari enam belas soal tersebut dibagi menjadi tiga level yaitu mudah, sedang dan sulit dengan rincian enam soal mudah, lima soal sedang, dan lima soal sulit. Kemudian guru mengharapkan soal yang

muncul hanya sepuluh dengan rincian empat soal mudah, tiga soal sedang, dan tiga soal sulit. Dari tiap level tersebut diacak sehingga meskipun diacak siswa tetap memperoleh jumlah soal tiap level sama.

3.10 Evaluasi Hasil Uji Coba

Diambil dua sampel peserta ujian yaitu atas nama Dimas dan Eko. Dilihat dari aspek persamaan soal antara kedua siswa tersebut antara lain:

- Jumlah soal sepuluh.
- Soal 1-4 termasuk level mudah, soal 5-7 termasuk level sedang, soal 8-10 termasuk level sulit.
- Memiliki urutan soal dan pilihan jawaban yang berbeda terhadap bank soal.

Dilihat dari aspek perbedaan soal antara kedua siswa tersebut antara lain:

- Memiliki urutan soal maupun pilihan jawaban yang berbeda antara soal milik Dimas dan Eko.
- Terdapat beberapa soal yang sama misalnya soal nomor 1 milik Dimas terdapat pada soal nomor 4 milik Eko, soal nomor 1 milik Eko terdapat pada soal nomor 2 milik Dimas.
- Terdapat beberapa soal yang berbeda misalnya soal nomor 3 milik Dimas tidak ada pada Eko, sebaliknya soal nomor 2 milik Eko tidak ada pada Dimas.

Dari *statement* diatas menunjukkan bahwa sistem berjalan sesuai rencana dan harapan.

4. KESIMPULAN

Dari sistem ujian intranet yang dibuat dapat diambil beberapa hal sebagai kesimpulan, yaitu:

1. Soal yang dikerjakan antar siswa tidak sama 100%, ada beberapa soal yang berbeda.
2. Urutan soal yang dikerjakan tiap siswa tidak sama, masing – masing memperoleh soal acak.
3. Meskipun soal antar siswa berbeda tetapi total soal dan total skor sama berarti menunjukkan bahwa tingkat kesulitan secara sistem sama.

4. Penggunaan RandomMatrix telah dapat menghasilkan soal acak dengan tingkat kesulitan yang sama tiap siswa.

5. SARAN

Untuk pengembangan lebih lanjut soal yang telah dikerjakan siswa dapat dibuka kembali dengan item dan urutan soal yang sama persis dengan yang dikerjakan siswa tersebut karena pemetaan random tersimpan didatabase.

Untuk pengembangan kedepan level kesulitan dapat dibuat secara dinamis tidak hanya terdiri dari mudah, sedang, sulit tetapi level kesulitan dapat dibuat tak terbatas sesuai kebutuhan guru masing – masing.

Memungkinkan penggabungan, perpindahan atau pertukaran soal antar bank soal.

DAFTAR PUSTAKA

- Arikunto, Suharsimi (2007) *Manajemen penelitian*, Jakarta, Rineka Cipta.
- Bowen, R. & Coar, K. (2000) *Apache Server Unleashed*, Sams Publishing, Indianapolis.
- Darin, E.H. (2001) *Selling E-Learning*, American Society for Training and Development.
- Desrizal (2014) *Panduan PHP dan MySQL Indonesia 1.0*, http://www.ajar.st-albertus.sch.id/TutorialTIK/panduan_php/array.html (diakses tanggal 28 Agustus 2014).
- Ellislab Inc. (2014) *CodeIgniter User Guide Version 2.2.0*, <https://ellislab.com/codeigniter/user-guide> (diakses tanggal 28 Agustus 2014).
- Jhonson P Harianja, dkk. (2013) *Sistem Ujian Adaptif Untuk Seleksi Penerimaan Mahasiswa Baru Pada Perguruan Tinggi Berbasis Web Melalui Penerapan Teknik MVC Dengan Menggunakan PHP Framework Codeigniter*, Skripsi, Fakultas Teknik, Universitas Maritim Raja Ali Haji, Tanjung Pinang.
- Jogiyanto (2005) *Analisis dan Desain Sistem Informasi*, Yogyakarta, Penerbit Andi.
- Kustiyaningsih, Yeni & R.A. Devie (2011) *Pemrograman Basis Data Berbasis Web Menggunakan PHP & MySQL*, Yogyakarta, Graha Ilmu.
- Adha, M.L. (2011) *Implementasi Aplikasi Ujian Online pada Training Karyawan Menggunakan Metode Computerized Classification Test Dengan Adaptive Feedback*, Skripsi, Fakultas Teknik. Universitas Indonesia. Depok.
- Mahyuzir, T.D. (1991) *Pengantar Analisis dan Perancangan Perangkat Lunak*, Elex Media Komputindo, Jakarta.
- Mujahidah (2009) *Perilaku Menyontek Laki-Laki dan Perempuan: Studimeta Analisis*, Skripsi, Fakultas Tarbiyah, Sekolah Tinggi Agama Islam Negeri, Samarinda.
- Nugroho, A. (2010) *Rekayasa Perangkat Lunak Menggunakan UML dan JAVA*, Andi Publisher.
- Oetomo & Budi S.D. (2002) *Perancangan dan Pembangunan Sistem Informasi*, Jakarta.
- Wahono, R.S. (2007) *Sistem eLearning Berbasis Model Motivasi Komunitas dan Penerapannya untuk eLearning Publik*, Pusat Dokumentasi dan Informasi Ilmiah LIPI, Jakarta.
- Widigdo & Anon K. (2003) *Dasar Pemrograman PHP dan MySQL*, Jakarta.
- Yuhfizar (2008) *10 JAM menguasai Internet: Teknologi dan Aplikasinya*, PT. Elex Media Komputindo, Jakarta.