

Pengembangan Aplikasi E-School Dengan Pendekatan *Evolutionary Prototype* Studi Kasus SMP Negeri 1 Cikarang Barat

Alawiyah Nurmalia¹, Bayu Priyambadha², Denny Sagita Rusdianto³

Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Brawijaya
Email: ¹alawiyahnurmalia60@gmail.com, ²bayu_priyambadha@ub.ac.id, ³denny.sagita@ub.ac.id

Abstrak

Perkembangan teknologi saat ini telah mencapai ke segala bidang termasuk pendidikan. Dengan adanya dukungan dari teknologi, kegiatan belajar mengajar dapat meningkat ke taraf yang lebih baik. SMP Negeri 1 Cikarang Barat memiliki fasilitas yang cukup memadai dari segi teknologi, namun fasilitas tersebut belum sepenuhnya di manfaatkan untuk kegiatan belajar mengajar. Pada penelitian ini, dikembangkan sebuah aplikasi pembelajaran bernama *E-School* yang memiliki fitur cek plagiarisme terhadap jawaban tugas maupun kuis siswa yang berjenis *essay*. Fitur cek plagiarisme dikembangkan dengan menerapkan algoritme *knuth-morris-pratt*. Pengembangan aplikasi menggunakan pendekatan *Evolutionary Prototype* yang dapat membantu *stakeholder* serta pengembang sistem dalam menggali spesifikasi kebutuhan sistem sesuai yang diinginkan oleh *stakeholder*. Dengan diterapkannya pendekatan *Evolutionary Prototype*, *stakeholder* dapat bersama-sama menentukan seperti apa sistem yang akan dikembangkan dan hal ini akan berdampak terhadap beberapa penambahan serta perubahan alur fungsional yang membedakan antara *prototype* pertama dengan *prototype* kedua. *Stakeholder* serta pengembang dapat mengevaluasi *prototype* aplikasi yang dikembangkan, hal ini dapat memudahkan komunikasi penggalan kebutuhan sistem serta dapat menghindari beberapa kesalahpahaman yang ada antara *stakeholder* dengan pengembang dalam membangun aplikasi. Hasil dari pengujian didapatkan semua fungsionalitas yang terdapat diaplikasi *prototype* satu dan *prototype* dua telah berjalan dengan baik dan *prototype* kedua sudah dapat digunakan oleh *stakeholder*.

Kata kunci: *learning management system, evolutionary prototype, algoritme knuth-morris-pratt, deteksi plagiarisme, pencocokan string*

Abstract

Technology development nowadays has reached all areas including education. With the support of technology, teaching and learning activities can be increase to a better level. SMP Negeri 1 Cikarang Barat has adequate technology facilities, but the facility is not fully utilized yet for teaching and learning activities. In this research, developed a learning application called *E-School* which has a feature to detect plagiarism on the answers of tasks and student quizzes which type is *essay*. The plagiarism detection feature will be developed by applying *knuth-morris-pratt* algorithm. The application applying the *Evolutionary Prototype* approach that could be help stakeholders and system developers in exploring system requirement specification as desired by stakeholder. By applying *Evolutionary Prototype* approach, stakeholders can work together as to what will be developed and this will impact some additional and functional changes between the first prototype and the second prototype. Stakeholder and developer can evaluated the prototype of the application that is being developed, this can be simplify the communication in the aim for system elicitation and this can avoid some of misunderstanding between stakeholder and developer in builiding application. The results of the test obtained all the functionality contained in the application first prototype and second prototype has been running well and the second prototype can be used by stakeholder.

Keywords: *learning management system, evolutionary prototype, knuth-morris-pratt algorithm, plagiarism detection, string matching*

1. PENDAHULUAN

Seiring dengan berkembangnya dunia teknologi, dunia pendidikan dapat pula ditunjang oleh teknologi agar kegiatan belajar mengajar dapat meningkat ke taraf yang lebih baik. SMP Negeri 1 Cikarang Barat merupakan salah satu sekolah menengah pertama yang masih menggunakan cara mengajar yang konvensional, padahal fasilitas yang ada di SMP ini terbilang memadai. Dari hasil observasi dan wawancara, sekolah ini belum sepenuhnya memanfaatkan fasilitas yang ada untuk mendukung kegiatan belajar mengajar. Dalam kegiatan belajar mengajar sehari-hari khususnya dalam pengerjaan tugas serta kuis seluruhnya masih mengandalkan kertas.

Permasalahan di sisi siswa yaitu tak jarang terdapat siswa yang kurang memiliki motivasi untuk membawa dan mengerjakan tugasnya. Di sisi lain, guru tidak jarang menumpuk buku tugas siswa dan kertas hasil kuis ketika guru tersebut belum memiliki waktu untuk mengecek hasil pekerjaan tersebut, hal ini menimbulkan permasalahan dimana buku tugas siswa dapat hilang, terselip maupun tertukar dengan kelas lain yang masih diajar oleh guru yang sama begitu pula halnya dengan kertas hasil jawaban kuis.

Ditinjau dari permasalahan tersebut, sekolah ini memerlukan sebuah aplikasi yang dapat menunjang kebutuhan pembelajaran para siswa dan guru. Terdapat beberapa fitur aplikasi yang akan mendukung kegiatan belajar mengajar salah satunya adalah fitur cek plagiarisme yang diterapkan dengan menggunakan salah satu algoritma *string matching* yaitu algoritme *knuth-morris-pratt*. Algoritme *knuth-morris-pratt* merupakan algoritme *string matching* yang tidak terlalu rumit namun lebih cepat dan efisien dibandingkan dengan algoritme *brute force* (Rudi, 2016). Algoritme *knuth-morris-pratt* memiliki beberapa kelebihan seperti running time pada algoritme ini adalah $O(m+n)$ yang mana ini termasuk cepat (Sa'diah, 2017) dan algoritme *knuth-morris-pratt* tidak perlu untuk bergeser mundur dalam melakukan pencocokan string, ini bermanfaat pada pencocokan string pada file string yang berukuran besar (Janani & Vijayarani, 2016).

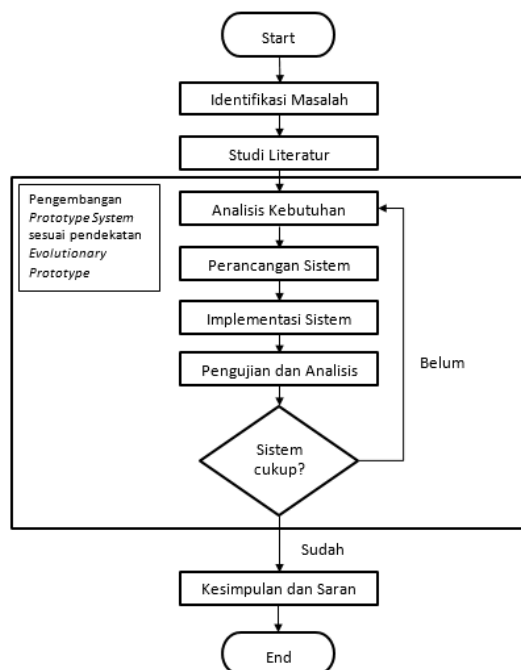
Untuk melakukan pengembangan aplikasi, Penelitian ini menggunakan sebuah model pengembangan perangkat lunak bernama *Evolutionary Prototype*, dimana pendekatan

prototype dianggap sebagai pendekatan yang paling baik untuk membantu tim pengembang serta *stakeholder* dalam memahami spesifikasi kebutuhan yang belum jelas (Pressman, 2010). *Stakeholder* dapat berdiskusi dengan pengembang selama pengembangan sistem berlangsung sehingga pada akhirnya akan didapatkan suatu sistem maupun aplikasi yang dikembangkan secara *evolusioner* sesuai dengan keinginan *stakeholder*.

Dilihat dari permasalahan serta keuntungan dari metode pengembangan *evolutionary prototype* dan algoritme *knuth-morris-pratt* dalam mengembangkan fitur cek plagiarisme pada aplikasi, maka penulis mengangkat penelitian dengan judul “Pengembangan Aplikasi E-School Dengan Pendekatan Evolutionary Prototype Studi Kasus SMP Negeri 1 Cikarang Barat”.

2. METODOLOGI

Penelitian dilakukan sesuai dengan diagram alir metodologi penelitian yang menggunakan pendekatan *Evolutionary Prototype*, berikut ini merupakan gambaran dari diagram bagaimana aplikasi *E-School* dikembangkan, yaitu sebagai berikut :



Gambar 1. Diagram alir metodologi penelitian

Dari gambar 1 dapat diketahui penelitian dimulai dengan melakukan identifikasi pada permasalahan yang terdapat dalam lingkup tempat studi kasus yaitu SMP Negeri 1 Cikarang

Barat. Dari hasil identifikasi, didapatkan beberapa masalah yang harus difokuskan terhadap satu permasalahan yang kemudian permasalahan tersebut akan menjadi dasar dalam melakukan penggalian kebutuhan dalam mengembangkan perangkat lunak.

Tahap kedua dalam penelitian ini adalah melakukan studi literatur, studi literatur bertujuan untuk mencari referensi pendukung penelitian terkait dengan *learning management system*, *evolutionary prototype* dan algoritme *knuth-morris-pratt*.

Tahap ketiga merupakan tahap dimana aplikasi mulai dikembangkan sesuai dengan pendekatan *evolutionary prototype* yang digunakan seluruh tahapan ini akan berulang jika stakeholder memiliki keinginan baru selama pengembangan sistem berlangsung, berikut tahapannya :

1. Analisis kebutuhan, merupakan tahapan penggalian kebutuhan yang sesuai dengan *scope* permasalahan yang telah diidentifikasi. Analisis kebutuhan dapat dilakukan berulang sesuai dengan perulangan pada pengembangan *prototype system*. Hasilnya akan didapatkan identifikasi dan spesifikasi kebutuhan.
2. Perancangan sistem, dilakukan setelah kebutuhan didapatkan dan tahap ini dapat berulang disesuaikan dengan adanya penambahan atau perubahan pada kebutuhan.
3. Implementasi sistem, dapat dilakukan beriringan dengan perancangan dan dapat pula dilakukan setelah perancangan selesai. Tahap ini akan berulang jika perancangan berubah.
4. Pengujian dan analisis, hasil implementasi diuji dan dilakukan analisis terhadap hasilnya. *Prototype system* yang telah selesai kemudian ditunjukkan kepada *stakeholder* untuk selanjutnya dapat dievaluasi serta didiskusikan untuk pengembangan fungsional tambahan, namun jika tidak terdapat penambahan atau perubahan fungsional maka *prototype system* tersebut dapat disebut dengan sistem.

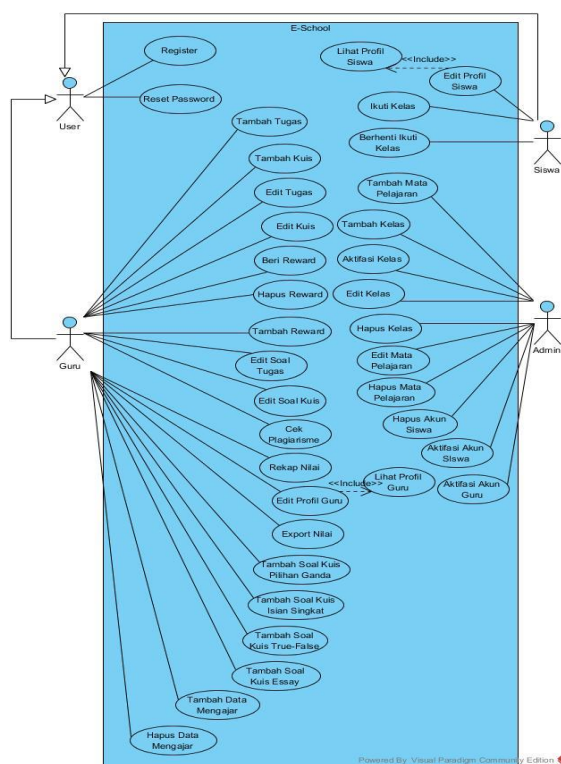
Tahap terakhir dalam penelitian adalah melakukan pengambilan kesimpulan serta saran yang akan berguna dalam pertimbangan pengembangan selanjutnya. Dalam menggunakan pendekatan *evolutionary prototype* sebenarnya proses dalam pengembangan bersifat tumpang tindih

(Sommerville, 2003).

3. ANALISIS DAN PERANCANGAN

Dalam penelitian ini, dilakukan iterasi sebanyak dua kali untuk melakukan pengembangan aplikasi E-School. Iterasi pertama pada tahapan analisis dan perancangan menghasilkan 14 kebutuhan fungsional dengan hanya memiliki 2 aktor yaitu guru dan siswa. Dari kebutuhan fungsional tersebut dimodelkan kedalam diagram *use-case* dan kemudian perancangan dimodelkan dengan diagram *sequence* dan diagram *class*.

Pada iterasi kedua, didapatkan penambahan kebutuhan fungsional sebanyak 30 kebutuhan termasuk kebutuhan fungsional untuk melakukan cek plagiasi pada sisi guru dan 3 kebutuhan fungsional mengalami perubahan dalam pengoperasian berjalannya fungsi. Kemudian pada iterasi kedua ini, aktor dibagi menjadi 3 yaitu guru, siswa dan admin. Adanya iterasi kedua ini disebabkan oleh permintaan penambahan atau perubahan kebutuhan fungsional yang telah dipertimbangkan oleh pengembang.



Gambar 2. Pemodelan *use case* diagram *prototype* 2

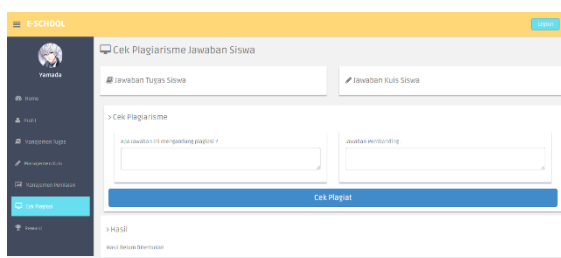
Kebutuhan fungsional pada iterasi kedua ini tetap dimodelkan dalam diagram *use-case* seperti pada gambar 2 dan perancangan tetap


```

21 }
22
23 //Fungsi untuk mencari
24 kecocokan pattern dengan teks
25
26 public function
27 KMPSearch($p,$t){
28 $resultAngka = array();
29 $pattern = str_split($p);
30 $text = str_split($t);
31 $prefix = $this-
32 >preKMP($pattern);
33 $i = $j = 0;
34 $num=0;
35 while($j<count($text)){
36 while($i>-1 &&
37 $pattern[$i]!=$text[$j]){
38 $i = $prefix[$i];
39 }
40 $i++;
41 $j++;
42 if($i>=count($pattern)){
43 $result[$num++]=$j-
44 count($pattern);
45 $resultAngka[$num++] +=1;
46 $i = $prefix[$i];
47 }
48 }
49 return $resultAngka;
50 }
51

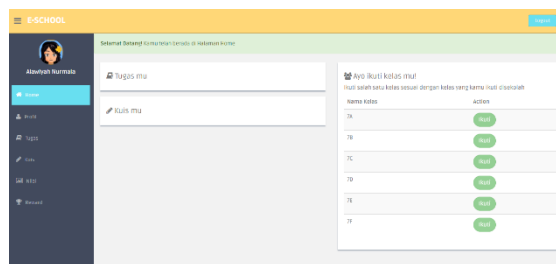
```

Algoritme *knuth-morris-pratt* pada tabel 1 merupakan algoritme inti dari *knuth-morris-pratt* yang diimplementasikan dalam bahasa pemrograman PHP. Algoritme ini akan membandingkan *pattern* dengan *text*, dalam penelitian ini cek plagiasi terhadap teks dengan teks dimana teks yang akan dibandingkan dijadikan sebagai *pattern* dan kemudian akan dilakukan pencocokan string menggunakan algoritme ini.



Gambar 4. Implementasi halaman cek plagiarisme

Gambar 4 merupakan salah satu tampilan dari halaman yang diambil dari *prototype 2*. Halaman tersebut merupakan halaman yang digunakan oleh guru untuk melakukan pengecekan plagiarisme terhadap jawaban tugas maupun kuis siswa yang berjenis *essay*.



Gambar 5. Implementasi halaman *home* siswa

Gambar 5 juga merupakan hasil implementasi di *prototype 2* yang merupakan halaman *home* siswa yang merupakan salah satu halaman yang dapat diakses oleh siswa yang telah memiliki akun aplikasi E-School.



Gambar 6. Implementasi halaman admin

Gambar 6 merupakan salah satu halaman yang hanya dapat diakses oleh admin. Halaman ini merupakan halaman yang digunakan oleh admin untuk melakukan manajemen mata pelajaran.

5. STRATEGI PENGUJIAN

Sesuai dengan metodologi yang dideskripsikan pada bagian kedua, pengujian yang dilakukan untuk menguji aplikasi E-School adalah pengujian jenis *whitebox* dan *blackbox*, kemudian untuk menguji kebutuhan non-fungsional digunakan developer tools yaitu *Performance-Analyzer*. Pengujian dilakukan pada *prototype* sistem pertama dan pada *prototype* sistem kedua yang selanjutnya dapat disebut sistem. Pengujian *whitebox* mengambil 3 sample dari *prototype* pertama dan 3 sample dari *prototype* kedua. Pengujian *whitebox* menghasilkan alur algoritme yang teruji secara benar sesuai dengan perhitungan *cyclometric complexity*. Untuk pengujian *blackbox*, dilakukan pengujian fungsional sistem pada sistem *prototype* pertama dan juga pada sistem *prototype* kedua. Pengujian *blackbox* membantu penulis untuk mengetahui apakah fungsionalitas sistem telah berjalan sesuai dengan apa yang dideskripsikan pada kebutuhan. Pengujian *whitebox* dan *blackbox* dilakukan sebanyak iterasi dari pembuatan *prototype* sistem.

Untuk pengujian dari kebutuhan non-fungsional yaitu *performance*, dilakukan pengambilan 3 sampel halaman yang akan diujikan dari masing-masing *prototype*. Hasil pengujian yang diharapkan adalah halaman dapat diakses dalam waktu kurang dari 3 detik.

6. HASIL

Hasil pengujian mendapatkan hasil dimana semua kebutuhan fungsional yang diimplementasikan pada *prototype* pertama serta *prototype* kedua berjalan dengan baik dan sesuai dengan kebutuhan fungsional yang didefinisikan, berikut merupakan tabel sampel hasil pengujian *blackbox* untuk *prototype* pertama dan *prototype* kedua.

Tabel 2. Kasus Uji *Requirement* Tambah Tugas *Prototype* 1

Kasus dan hasil uji (Kondisi normal)			
Prosedur pengujian	Hasil yang diharapkan	Hasil pengamatan	Kesimpulan
Guru melakukan input pada form yang tersedia di menu tugas dengan lengkap dan menekan tombol submit.	Tugas baru akan tersimpan dan siswa yang melakukan add pada guru tersebut dapat melihat tugas tersebut.	Tugas baru tersimpan dan setiap siswa yang menambahkan guru tersebut sebagai gurunya, tugas tersebut terlihat pada halaman tugas siswa	Valid
Kasus dan hasil uji (Kondisi alternatif)			
Guru tidak melakukan input apapun pada form tugas dan menekan tombol submit.	Sistem akan menampilkan pemberitahuan bahwa form tersebut wajib diisi.	Form menampilkan pemberitahuan bahwa harus diisi lengkap.	Valid

Tabel 2 merupakan tabel yang menunjukkan uji *requirement* pada kebutuhan fungsional tambah tugas yang ada pada *prototype* pertama. Pada *prototype* pertama kebutuhan fungsional tambah tugas telah berjalan dengan baik.

Tabel 3. Kasus Uji *Requirement* Tambah Kuis *Prototype* 1

Kasus dan hasil uji (Kondisi normal)			
Prosedur pengujian	Hasil yang diharapkan	Hasil pengamatan	Kesimpulan
Guru melengkapi form tambah kuis pada halaman	Kuis baru berhasil dibuat.	Kuis baru berhasil dibuat dan dapat ditambahkan soal.	Valid

yang tersedia di menu kuis dan menekan tombol submit.			
Kasus dan hasil uji (Kondisi alternatif)			
Guru tidak melengkapi form tambah kuis dan menekan tombol submit.	Terdapat pemberitahuan bahwa field-field tertentu harus diisi.	Pemberitahuan muncul pada field yang harus diisi.	Valid

Tabel 3 merupakan hasil dari pengujian *requirement* tambah kuis yang ada di *prototype* 1, hasilnya menunjukkan bahwa kebutuhan fungsional tambah kuis juga sudah sesuai dengan apa yang didefinisikan di kebutuhan fungsional.

Tabel 4. Kasus Uji *Requirement* Ikuti Kelas *Prototype* 2

Kasus dan hasil uji (Kondisi normal)			
Prosedur pengujian	Hasil yang diharapkan	Hasil pengamatan	Kesimpulan
Siswa telah berada di halaman home dan belum mengikuti kelas manapun dan menekan tombol ikuti pada salah satu kelas	Siswa tidak dapat mengikuti kelas lainnya lagi setelah memilih untuk mengikuti satu kelas.	Tombol ikuti pada daftar kelas yang tersedia menjadi tidak aktif sehingga siswa tidak dapat mengikuti kelas lain setelah mengikuti satu kelas.	Valid

Tabel 4 merupakan tabel kasus uji *requirement* ikuti kelas yang ada di *prototype* 2. Dari tabel tersebut dapat diketahui bahwa kebutuhan fungsional ikuti kelas telah berhasil diimplementasikan pada *prototype* 2 yang menghasilkan hasil uji yang valid.

Tabel 5. Kasus Uji *Requirement* Aktifasi Akun Siswa *Prototype* 2

Kasus dan hasil uji (Kondisi normal)			
Prosedur pengujian	Hasil yang diharapkan	Hasil pengamatan	Kesimpulan
Admin melakukan login dan masuk ke halaman home admin. Pada data pengguna baru yang termasuk	Data siswa tersebut langsung ke halaman data pengguna siswa dan nama siswa tersebut tidak muncul kembali pada	Setelah admin menekan tombol aktifkan, akun siswa yang telah diaktifkan tersebut terhapus dalam daftar pengguna	Valid

siswa, admin menekan tombol aktifkan	halaman daftar pengguna baru siswa.	baru siswa dan masuk kedalam daftar pengguna siswa	
--	--	---	--

Tabel 5 merupakan kasus uji yang dibuat untuk menguji fungsi aktivasi akun siswa di sisi admin. Dari tabel tersebut dengan skenario kasus uji seperti yang tertera pada tabel 5 maka fungsionalitas untuk aktivasi akun siswa adalah valid.

Untuk pengujian *blackbox*, dari seluruh kebutuhan fungsional di prototype ke 1 hingga prototype ke 2 menghasilkan uji yang valid sesuai dengan skenario pengujian yang didefinisikan pada tabel.

Pada pengujian *whitebox*, semua sample algoritme fungsi yang diujikan mendapatkan hasil yang valid dari hasil pengujian *basis path*. Pada pengujian *whitebox* untuk *prototype 1* menghasilkan ketiga algoritme fungsi yang valid berdasarkan hasil hitung menggunakan rumus *cyclometric complexity*. Pengujian *whitebox* juga dilakukan pada *prototype 2*, dari ketiga sampel algoritme yang diujikan menggunakan pengujian jenis *basis path* hasilnya ketiga sampel tersebut menunjukkan hasil yang valid sesuai dengan rumus perhitungan *cyclometric complexity*. Berikut merupakan tabel 6 yaitu salah satu sample yang dilakukan pengujian *whitebox* di *prototype 2*.

Tabel 6. Kasus Uji Jalur Independen Cek Plagiarisme

No	Jalur	Data Input	Hasil yang diharapkan	Hasil yang diperoleh	Status
1	1 – 2 – 11	Sesi on = 7200 detik	Sistem akan mengeluarkan guru kembali ke halaman login karena session time habis.	Session time di set 7200 detik atau setara dengan 2 jam. Setelah 2 jam dan guru memilih untuk melakukan cek plagiarisme tapi tidak berhasil karena sistem mengarahkan ke halaman login.	Valid
2	1 – 3 – 4 – 5 –	Teks 1 dan Teks 2	Sistem dapat mengeluarkan hasil kesamaan	Sistem berhasil menampilkan tingkat	Valid

6 –		teks yang	kemiripan
7 –		dibandingka	teks 1
8 –		n dalam	dengan teks
9 –		bentuk	2 dalam
10 –		angka	bentuk
11		persentase.	angka
			persentase
			pada kolom
			hasil.

Pada tabel 6, terdapat kolom jalur dimana jalur tersebut didapatkan pada saat pengubahan *pseudocode* menjadi bentuk *flow graph* yang kemudian dihitung menggunakan rumus *cyclometric complexity*. Hasil dari perhitungan tersebut kemudian didapatkan jalur independen yang digunakan untuk menentukan kasus uji.

Pengujian non-fungsional sistem pada penelitian ini juga dilakukan. Pengujian masing-masing dilakukan dengan mengambil 3 sampel halaman dari *prototype 1* dan *prototype 2*. Pada penelitian ini, pengujian non-fungsional yang diangkat adalah pengujian *performance* pengaksesan tiap halaman, dengan mengandalkan sebuah *developer tools* bernama *Performance-Analyzer*. Pada *prototype 1*, diambil 3 sampel halaman untuk dilakukan pengaksesan pada setiap halaman. Hasil yang didapatkan rata-rata pengaksesan halaman masih berkisar 0-1 detik. Berikut ini gambar 1 yang merupakan salah satu sample dari halaman di *prototype 1* yang diujikan.

Requests	Domains	Subdomains of TLD	Requests to Host
12	2	0	11
TLD Subdomain Requests	Total	Time to First Byte	DOM Content Loading
11	1965ms	1746ms	153ms
DOM Processing	Slowest Call	Average Call	
204ms	10ms	6ms	

Gambar 7. Hasil pengujian *performance* halaman tugas siswa *prototype 1*

Pada gambar 7, menunjukkan bahwa terdapat 12 *requests* untuk setiap kali pemanggilan halaman tugas siswa dan rata-rata pemanggilan halaman ini didapatkan waktu 10 ms.

Untuk *prototype 2*, diambil pula 3 sampel halaman untuk diuji *performance* akses halamannya. Halaman sampel yang diambil pada *prototype 2* merupakan halaman sampel yang berbeda dengan apa yang telah diujikan pada *prototype 1*. Hasil waktu rata-rata yang didapatkan pada pengaksesan ketiga halaman yang ada di *prototype 2* yaitu berkisar 0-1 detik. Berikut ini merupakan gambar halaman salah satu sampel yang dilakukan uji *performance* pada *prototype 2*.

Requests	Domains	Subdomains of TLD	Requests to Host
19	3	0	15
TLD & Subdomain Requests	Total	Time to First Byte	DOM Content Loading
15	1215ms	305ms	542ms
DOM Processing	Slowest Call	Average Call	
889ms	414ms	302ms	

Gambar 8. Hasil pengujian *performance* halaman home admin *prototype 2*

Pada gambar 8, memiliki 19 *requets* untuk setiap kali halaman ini dipanggil dan mendapatkan waktu total pemanggilan yaitu 1,215 detik.

Banyak faktor yang mempengaruhi cepat-lambatnya pengaksesan halaman aplikasi ini, salah satunya adalah kecepatan akses internet. Selain kecepatan internet, banyaknya file yang diakses dalam halaman juga menjadi pengaruh cepat-lambatnya halaman dapat terakses.

Dari pengujian *blackbox*, *whitebox* dan non-fungsional yang telah dilakukan, seluruh kebutuhan fungsional dan kebutuhan non-fungsional yang diujikan sesuai dengan kasus uji masing-masing mendapatkan hasil yang valid. Untuk *prototype 2* dapat selanjutnya disebut sistem jadi karena seluruh kebutuhan fungsional yang telah terimplementasikan dan diujikan telah sesuai dengan apa yang diinginkan oleh *stackholder*.

7. KESIMPULAN DAN SARAN

Pada penelitian ini, digunakan pendekatan *evolutionary prototype* untuk mengembangkan aplikasi E-School yang dapat menunjang kegiatan belajar mengajar di SMP Negeri 1 Cikarang Barat. Dengan digunakannya pendekatan *evolutionary prototype*, dalam penelitian ini dilakukan 2 kali iterasi dalam membangun aplikasi E-School untuk menyesuaikan dengan apa yang diinginkan oleh *stakeholder*. *Prototype* aplikasi yang kedua telah dapat digunakan oleh *stakeholder* dan tidak ada penambahan maupun perubahan alur fungsionalitas.

Dari 14 spesifikasi kebutuhan yang terdefinisi pada *prototype 1*, hasil keseluruhan pengujian *blackbox* menyatakan valid. Sedangkan untuk di *prototype 2* terdapat 30 spesifikasi kebutuhan fungsional yang di implementasikan di *prototype 2* dan 5 spesifikasi kebutuhan fungsional dari *prototype 1* yang berubah alur kerjanya, hasil pengujian *blackbox* yang dilakukan menghasilkan hasil uji yang

valid pada setiap spesifikasi kebutuhan yang telah di implementasikan tersebut.

Untuk pertimbangan pengembangan aplikasi selanjutnya pada strukturasi kode program dapat diperbaiki agar memiliki struktur kode yang baik serta mengurangi penggunaan kode program yang tidak perlu. Sedangkan dari sisi fitur, untuk kedepannya mungkin *stakeholder* memiliki keinginan untuk menambahkan fungsi baru yang selanjutnya dapat dipertimbangkan.

DAFTAR PUSTAKA

- Ajrina, D.S., 2016. Pengembangan sistem manajemen data pasien klinik gigi berbasis web (studi kasus : klinik Drg. Damayanti bogor). Skripsi. Universitas Brawijaya.
- Arkorful, V. & Abaidoo, N., 2014. The role of e-learning, the advantages and disadvantages of its adoption in Higher Education. *International Journal of Education and Research*, 2(12), pp. 397-410.
- Barber, S., 2006. *How Fast Does a Website Need To Be?*. PerfTestPlus, Inc. USA
- Fatansyah. 2012. Basis data edisi revisi. Informatika. Bandung.
- Gartzen, T., Brambring, F. Dan Basse, F., 2016. *Target-oriented prototyping in highly iterative product development*. 3rd International Conference on Ramp-up Management (ICRM).
- Griffiths, A., 2010. *CodeIgniter 1.7 professional development*. Packet Publishing. Birmingham.
- Haryanto, W., Adhi, A. & Purwinarko, A., 2016. Usability Laman Penerimaan Mahasiswa Baru. *Scientific Journal of Informatics*, 3(1).
- Janani, R. & Vijayarani, S. 2016. *An Efficient Text Pattern Matching Algorithm for Retrieving Information from Desktop*. Indian Journal of Science and Technology. Vol 9(43).
- Lethbridge, T. & Laganier, R., 2005. *Object-Oriented Software Engineering: Practical Software Development Using UML and Java*. 2nd edition. London: McGraw-Hill.
- Madhavi, D., 2016. A White Box Testing

- Technique in Software Testing: Basis Path Testing. *Journal for Research*, 02(04), pp. 12-17.
- Nugroho, A., 2010. Rekayasa Perangkat Lunak Berorientasi Objek dengan Metode USDP. Jakarta : Andi Offset.
- Oludare Jethro, O., Moradeke Grace, A. & Kolawole Thomas, A., 2012. E-Learning and Its Effects on Teaching and Learning in a Global Age. *International Journal of Academic Research in Business and Social Sciences*, 2(1), pp. 203-210.
- Oxford, 2017. *Indonesian Oxford Living Dictionaries BETA*. [Online] Available at: <https://id.oxforddictionaries.com/> [Diakses 13 08 2017].
- Pratama, Y.C., 2016. Pembangunan sistem informasi manajemen restaurant dan pemesanan terintegrasi dengan menggunakan metode pendekatan prototype. Skripsi. Universitas Brawijaya.
- Pressman, R. S. 2010. *Software engineering : a practitioner's approach 7th ed*. McGraw-Hill., New York.
- Rudi, C. M. 2016. Menentukan Tingkat Kecocokan Teks Menggunakan Algoritme KMP. Makalah IF2211 Strategi Algoritme. Institut Teknologi Bandung.
- Rumbaugh, J., Jacobson, I. dan Booch, G., 2005. *The Unified Modeling Language reference manual 2nd ed*. Addison-Wesley. Boston.
- Sa'diah, H. T. 2017. Implementasi Algoritme Knuth-Morris-Pratt Pada Fungsi Pencarian Judul Tugas Akhir Repository. *Jurnal Komputasi*. Vol 14 No 1. pp 115-124.
- Sommerville, I., 2000. *Software engineering 6th ed*. Addison-Wesley. Harlow.
- Sommerville, I., 2003. *Software engineering : rekayasa perangkat lunak jilid 1 edisi 6*. Erlangga. Jakarta.
- Sommerville, I., 2007. *Software engineering 8th ed*. Addison-Wesley. Harlow.
- Sommerville, I., 2011. *Software engineering 9th ed*. Addison-Wesley. London.
- Susanto, R. dan Andriana, A.D. 2016. Perbandingan Model Waterfall Dan Prototyping Untuk Pengembangan Sistem Informasi. [majalah ilmiah] Vol.14 No.1 bidang teknik. Universitas Komputer Indonesia.
- Syuhada, I., 2015. Inspirasi codeIgniter. Elex Media Komputindo. Jakarta
- Terho, H., Suonsyrjä, S., Systä, K. & Mikkonen, S., 2017. *Understanding the Relations Between Iterative Cycles. Proceedings of the 50th Hawaii International Conference on System Sciences*.
- Wibowo, A., Akhlis, I. & Nugroho, S., 2014. Pengembangan LMS (Learning Management System) Berbasis Web untuk Mengukur Pemahaman Konsep dan Karakter Siswa. *Scientific Journal of Informatics*, 1(2).