

# PENGEMBANGAN SISTEM INFORMASI *EVENT* MAHASISWA BERBASIS WEB DI UNIVERSITAS NEGERI YOGYAKARTA

## *THE DEVELOPMENT OF WEB BASED STUDENT EVENTS INFORMATION SYSTEM IN YOGYAKARTA STATE UNIVERSITY*

Oleh: Nanang Wisnu Pambudi,  
Universitas Negeri Yogyakarta,  
pambudinw@gmail.com

### **Abstrak**

Tujuan dari penelitian ini adalah untuk mengembangkan sistem informasi *event* mahasiswa di Universitas Negeri Yogyakarta, serta mengetahui tingkat kualitas sistem informasi *event* mahasiswa Universitas Negeri Yogyakarta berdasarkan standar pengujian kualitas perangkat lunak ISO/IEC 25010. Metode yang digunakan dalam penelitian ini adalah *Research and Development (R&D)*. Pengembangan perangkat lunak menggunakan model pengembangan *waterfall* yang terdiri atas 4 tahapan yaitu analisis kebutuhan, desain atau perancangan, implementasi, dan pengujian. Hasil dari penelitian ini adalah 1) Sistem Informasi *Event* Mahasiswa di Universitas Negeri Yogyakarta yang memiliki fungsi untuk membantu penyelenggara *event* menyampaikan informasi *event* kepada mahasiswa. Aplikasi ini dikembangkan dengan *framework* Laravel. 2) Pada aspek *functional suitability* diperoleh bahwa semua fungsi telah berjalan dengan baik dan benar. Pada aspek *performance efficiency* diperoleh bahwa aplikasi mendapatkan skor PageSpeed adalah 99 dan YSlow adalah 82 dengan waktu respon rata-rata halaman adalah 2,78 detik yang ada di bawah waktu respon maksimal 10 detik menurut Nielsen(1993). Pada aspek *usability* diperoleh rata-rata skor angket SUS dari 20 responden adalah 80,13 yang berarti di atas rata-rata skor *System Usability Scale(SUS)* yaitu 68. Pada aspek *maintainability* diketahui dengan *code duplication* sebesar 0% aplikasi termasuk dalam kategori A yang berarti sangat baik. Pada aspek *portability* diketahui aplikasi dapat berjalan baik dan tanpa cacat pada *Google Chrome, Mozilla Firefox, dan Opera* maka aplikasi dapat dikategorikan sebagai *Green Application*. Pada aspek *security*, aplikasi dinyatakan bebas dari kerentanan dengan tingkat resiko tinggi seperti XSS dan SQLi. Pada aspek *realibility*, aplikasi dinyatakan sukses dalam pengujian dengan skenario 20 pengguna mengakses aplikasi secara bersamaan dalam 10 menit.

Kata kunci: sistem informasi, *event* mahasiswa, ISO/IEC 25010

### **Abstract**

*The goals of this study were to develop student event information system in Yogyakarta State University, and to know about the level of quality of the student event information system based on ISO/IEC 25010. The method was used in this study is Research and Development (R&D). The software development model used waterfall development model that consists of 4 stages. They are analysis of needs, design, implementation, and examination. The results of this study were 1) Student Event Information System in Yogyakarta State University which has a function to assist event organizers to convey information to students. This application was developed with Laravel framework. 2) In functional suitability aspect of ISO/IEC 25010, every functions in this application had worked well and correctly. In performance efficiency aspect, application got score 99 in PageSpeed standard and 82 in YSlow with average response time of pages are 2.78 seconds which was below the maximum response time of 10 seconds by Nielsen(1993). In usability aspect, the application obtained an average score 80.13 in System Usability Scale(SUS) questionnaire from 20 respondents which means the scores above 68, the average score of SUS. In maintainability aspect, the application has 0% code duplication which means the application can be categorized in category A or very good by TIOBE. In portability aspect, the application runs well and flawlessly in Google Chrome, Mozilla Firefox, and Opera which means the application is categorized as green application in those web browsers. In security aspect, the application is free from high risk vulnerability such as XSS and SQLi. In realibility aspect, the application succeeds in examination with scenario of 20 users accessing the application simultaneously in 10 minutes*

Keywords: information system, student events, ISO/IEC 25010

## PENDAHULUAN

Selain kegiatan perkuliahan, di Universitas Negeri Yogyakarta terdapat banyak kegiatan atau *event* lain yang dapat diikuti oleh mahasiswa. Berbagai *event* tersebut antara lain berbentuk lomba, seminar, festival, dll. Penyelenggara berbagai *event* tersebut pun ada berbagai macam. Antara lain adalah dari instansi pemerintah seperti Direktorat Jendral Pendidikan Tinggi dan Kementrian Pemuda dan Olahraga, pihak kampus, organisai mahasiswa, dan swasta.

Mahasiswa akan memperoleh berbagai hal yang berguna dengan mengikuti *event* di luar perkuliahan. Misalnya dengan mengikuti seminar tentang kepemimpinan, mahasiswa akan memperoleh materi-materi tentang kepemimpinan yang belum tentu diajarkan di kelas. Selain itu dengan mengikuti *event* di luar kelas, mahasiswa akan mendapatkan relasi yang lebih luas. Adapun kelebihan lain adalah mahasiswa yang aktif di luar kegiatan kelas akan memiliki nilai lebih dalam Surat Keterangan Pendamping Ijazah (SKPI). Berdasarkan wawancara dengan Bapak Drs. Mujiran selaku Kepala Bagian Kemahasiswaan Universitas Negeri Yogyakarta pada tanggal 28 April 2016, SKPI yang diberikan kepada mahasiswa di akhir kuliah memuat rekam jejak mahasiswa selama di kampus. Rekam jejak mahasiswa tersebut juga meliputi prestasi dan partisipasi atas kegiatan-kegiatan di luar kelas yang dilakukan mahasiswa. Bahkan untuk mendukung hal tersebut, pihak universitas tengah menyiapkan sistem informasi untuk melakukan pencatatan sertifikat-sertifikat yang diperoleh mahasiswa dari berbagai *event* tersebut.

Ada berbagai cara untuk mendapatkan informasi tentang adanya *event* di Universitas Negeri Yogyakarta. Poster *event* di papan informasi adalah salah satu diantaranya. Papan informasi adalah papan yang disediakan untuk menempelkan berbagai macam informasi. Papan informasi tersebar di berbagai tempat seperti di rektorat, dekanat, kantor jurusan, kantor Unit Pelaksana Tugas, Student Center, dll. Namun ukuran papan informasi yang terbatas kadangkala tidak memungkinkan untuk menempelkan semua informasi *event* yang ada. Adapun cara lain yang

digunakan untuk menginformasikan *event* adalah dengan media sosial. Namun terbatasnya akun yang mahasiswa yang dimiliki membuat tidak semua mahasiswa mendapat informasi. Sementara itu dari sisi penyelenggara *event*, jumlah sarana informasi *event* yang banyak dan beragam membuat penyelenggara *event* memerlukan lebih banyak usaha maupun biaya untuk menyampaikan informasi *event*. Sebagai contoh penyelenggara *event* akan memerlukan banyak usaha dan biaya untuk membuat dan menyebarkan poster di setiap papan informasi yang ada di kampus. Sehingga akan lebih baik jika ada sebuah media untuk berbagi dan memperoleh informasi *event* secara terpadu.

Dalam satu tahun ada banyak sekali *event* yang bisa diikuti mahasiswa Universitas Negeri Yogyakarta. Bahkan kadang kala dalam satu waktu yang berdekatan ada dua *event* yang bisa diikuti. Hal tersebut membuat mahasiswa harus mampu cermat dalam mengatur jadwalnya sendiri. Terlebih mahasiswa juga memiliki tugas pokok untuk mengikuti kegiatan perkuliahan. Sehingga diperlukan media untuk membantu pengelolaan jadwalevent yang akan diikuti mahasiswa.

Pemanfaatan teknologi informasi dapat dijadikan sebagai solusi dalam permasalahan media untuk informasi *event*. Hal ini dapat kita buktikan bahwa dengan berkembangnya teknologi informasi saat ini telah banyak kegiatan di Universitas Negeri Yogyakarta yang terbantu dengan pemanfaatan teknologi informasi. Berbagai contohnya antara lain adalah Sistem Informasi Akademik (SIKAD), e-learning Universitas Negeri Yogyakarta (Be Smart), Sistem Informasi Registrasi Mahasiswa, dll.

Jenis teknologi informasi yang dapat digunakan untuk mengembangkan sistem informasi *event* tersebut adalah aplikasi *web*. Aplikasi ini memiliki keunggulan yaitu fleksibel karena hanya bergantung dengan *web browser* yang dimiliki pengguna. Hal ini membuat mahasiswa tidak perlu repot memasang aplikasi pada perangkat yang mereka miliki. Mahasiswa cukup membuka aplikasi pada *web browser* yang ada pada perangkat .

Penggunaan teknologi informasi akan menjadi sia-sia apabila dalam aplikasi yang dibuat banyak terjadi kesalahan setelah didistribusikan pada pengguna. Sehingga agar mendapatkan hasil berupa perangkat lunak yang berkualitas, dalam proses pengembangan aplikasi perlu dilakukan pengujian kualitas perangkat lunak. Diantara standar pengujian tersebut, ISO/IEC 25010:2011 merupakan standar internasional pengujian kualitas perangkat lunak. ISO/IEC 25010 dibuat oleh *International Organization for Standardization and International Electrotechnical Commission* sebagai revisi atau pengganti ISO/IEC 9126 (ISO, 2011).

Oleh karena itu penelitian ini bertujuan untuk mengembangkan sistem informasi berupa aplikasi *web* di Universitas Negeri Yogyakarta yang berfungsi untuk menginformasikan *event* kepada mahasiswa sekaligus memudahkan mahasiswa untuk mengelola berbagai *event* yang mereka agendakan melalui penelitian dan pengembangan.

## METODE PENELITIAN

### Model Pengembangan

Penelitian ini menggunakan metode *Research and Development* (R&D). Menurut Sugiyono (2013:407), metode penelitian dan pengembangan atau *Research and Development* adalah metode penelitian yang digunakan untuk menghasilkan produk tertentu, dan menguji keefektifan produk tersebut. Pada penelitian ini, produk yang dikembangkan oleh peneliti adalah berupa perangkat lunak. Model pengembangan perangkat lunak yang digunakan oleh peneliti adalah model *waterfall*.

### Prosedur Pengembangan

Karena penelitian ini menggunakan model pengembangan perangkat lunak *waterfall*, maka prosedur pengembangan pada penelitian ini menggunakan empat tahapan model pengembangan perangkat lunak *waterfall*. Keempat tahapan tersebut adalah analisis kebutuhan, desain, implementasi, dan pengujian.

Tahapan analisis kebutuhan adalah tahapan dimana pengembang perangkat lunak

menganalisis dan mendokumentasikan spesifikasi kebutuhan perangkat lunak agar perangkat lunak sesuai dengan apa yang dibutuhkan pengguna.

Proses analisis kebutuhan dilakukan dengan melakukan observasi terhadap proses informasi *event* di Universitas Negeri Yogyakarta dan wawancara dengan Kepala Bagian Kemahasiswaan Universitas Negeri Yogyakarta, Bapak Drs. Mujiran. Selain itu, peneliti juga melakukan studi literatur tentang aplikasi *web* dan standar kualitas aplikasi *web*.

Tahapan desain adalah tahapan dimana pengembang membuat rancangan pembuatan program seperti struktur data, arsitektur perangkat lunak, representasi antarmuka dan prosedur pengodean. Desain yang dilakukan pada penelitian ini meliputi perancangan diagram *Unified Model Language* yaitu *class diagram*, *activity diagram*, *use case diagram*, dan *sequence diagram*, serta perancangan database dan perancangan antarmuka.

Tahap implementasi merupakan tahapan dimana desain yang telah dibuat pada tahap sebelumnya ditranslasikan ke dalam bentuk perangkat lunak. Tahap ini juga disebut dengan tahap pengodean karena bentuk translasi ke perangkat lunak melalui kode-kode bahasa pemrograman.

Tahap pengujian adalah tahapan dimana perangkat lunak yang dihasilkan melalui tahap sebelumnya diuji kelayakannya sebelum disampaikan kepada pengguna. Pada penelitian ini, pengujian yang dilakukan menggunakan standar kelayakan perangkat lunak ISO/IEC 25010. ISO/IEC 25010 sendiri memiliki berbagai aspek yaitu *functional suitability*, *performance efficiency*, *usability*, *portability*, *security*, *maintainability*, *realibility*, dan *compatibility*.

Pengujian *functional suitability* menggunakan *checklist* pada *test case* yang berisi fungsi-fungsi berdasarkan analisis kebutuhan. *Test case* berguna untuk memastikan tidak ada kesalahan dalam perangkat lunak yang dikembangkan. Pengujian *functional suitability* dilakukan oleh responden ahli yang dalam pengembangan web.

Pengujian *performance efficiency* menggunakan perangkat lunak untuk mengukur performa dari perangkat lunak yang dikembangkan. Untuk pengujian aplikasi web secara umum mengukur kecepatan akses, dan kecepatan *request* dan *response* data.

Pengujian aspek *usability* menggunakan instrumen System Usability Scale(SUS). SUS adalah instrumen penelitian untuk *system usability* berbentuk kuesioner yang diciptakan oleh John Brooke.

Pengujian *portability* menggunakan berbagai *browser* dan sistem operasi untuk menguji kualitas perangkat lunak saat berjalan pada lingkungan perangkat lunak yang berbeda.

Pengujian *security* menggunakan perangkat lunak khusus untuk menemukan dan mengeksploitasi kerentanan aplikasi *web*.

Pengujian *maintainability* menggunakan *tools* untuk menguji *code duplication*. Menurut Jansen (2014), *code duplication* berpengaruh negatif terhadap aspek *maintainability* perangkat lunak. Dengan banyaknya *code duplication* dalam sebuah perangkat lunak akan membuat perubahan bagian *code* misalnya untuk mengatasi *bug* atau untuk menambahkan fungsi menjadi mempengaruhi bagian *code* lain yang merupakan hasil duplikasinya.

Pengujian *reability* menggunakan perangkat lunak khusus untuk menguji kehandalan sistem dalam kondisi tertentu selama jangka waktu yang ditentukan.

Aspek *compatibility* adalah karakteristik perangkat lunak yang memuat tentang sejauh mana sebuah produk, atau sistem dapat bertukar informasi dengan produk, sistem atau komponen dan/atau menjalankan fungsi lain yang diperlukan secara bersamaan ketika berbagi perangkat keras dan environment perangkat lunak yang sama. Sehingga pengujian untuk karakteristik ini tidak dilakukan karena sistem ini tidak berbagi perangkat keras dan lingkungan perangkat lunak yang sama dengan sistem lain.

#### **Sumber Data/Subjek Penelitian**

Subjek penelitian untuk menguji aspek *functional suitability* adalah tiga orang responden yang terdiri dari dosen, mahasiswa, dan ahli

dalam pengembangan perangkat lunak. Subjek penelitian untuk menguji aspek *performance efficiency*, *portability*, *security*, dan *maintainability* adalah perangkat lunak Sistem Informasi *Event* Mahasiswa. Sedangkan untuk aspek *usability* adalah 20 responden yang merupakan sasaran pengguna aplikasi. Jakob Nielsen (2012) menjelaskan bahwa dalam penelitian kuantitatif untuk mendapatkan angka yang signifikan secara statistik setidaknya diperlukan pengujian terhadap 20 responden. 20 orang responden tersebut dibagi menjadi 17 mahasiswa, 2 responden dari organisasi mahasiswa, dan 1 staf Bagian Kemahasiswaan Universitas Negeri Yogyakarta. Responden dari organisasi mahasiswa digunakan karena organisasi mahasiswa adalah termasuk penyelenggara *event* di Universitas Negeri Yogyakarta. Sedangkan responden dari Bagian Kemahasiswaan Universitas Negeri Yogyakarta untuk menguji sistem admin. Teknik sampling yang digunakan adalah sampling kuota. Menurut Sugiyono (2013:124), sampling kuota adalah teknik untuk menentukan sampel dari populasi yang mempunyai ciri-ciri tertentu sampai kuota yang diinginkan.

#### **Alat Pengumpul Data**

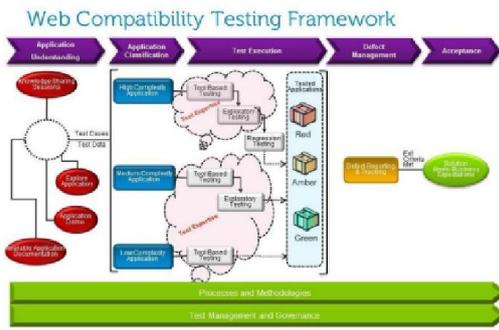
Alat atau instrumen pengumpul dalam penelitian ini adalah berupa angket dan perangkat lunak pengukuran.

Pengujian pada aspek *functional suitability* menggunakan *test case* yang berisi kriteria-kriteria yang memuat kebutuhan fungsionalitas aplikasi.

Pengujian pada aspek *performance efficiency* dilakukan dengan *load test* menggunakan aplikasi GTMetrix. Dengan aplikasi GTMetrix dapat diketahui nilai YSlow dan PageSpeed. YSlow dan PageSpeed adalah aturan penilaian performa aplikasi *web*. YSlow dikembangkan oleh Yahoo Developer Network, sedangkan PageSpeed dikembangkan oleh Google. GTMetrix juga memberikan informasi tentang waktu yang diperlukan untuk *load* halaman, besar ukuran halaman dan total request.

Pengujian pada aspek *usability* menggunakan kuesioner SUS (*System Usability*

Scale). SUS adalah instrumen untuk mengukur aspek *usability* sebuah sistem yang dikembangkan oleh John Brooke pada 1986.



Gambar 1. *Web Compatibility Testing Framework* (Kaalra & Gowthaman, 2014)

Pengujian *portability* dilakukan dengan melakukan tes kepada aplikasi web untuk berjalan di berbagai *web browser*. Pengujian pada aspek ini menggunakan *Web Compatibility Testing Framework* (Kaalra & Gowthaman, 2014). *Web browser* yang digunakan dalam penilaian ini adalah Google Chrome, Opera, dan Mozilla Firefox. Penilaian dilakukan dengan menjalankan semua halaman yang ada pada aplikasi. Kemudian diamati apakah setiap halaman berjalan tanpa cacat di *web browser*. Hasil pengamatan akan diklasifikasikan menjadi *Red*, *Amber*, atau *Green Application*.

Instrumen yang digunakan untuk menguji aspek *security* adalah *tool* OWASP ZAP. Dengan OWASP ZAP akan diuji kerentanan *web* sistem informasi terhadap serangan seperti XSS dan *SQL Injection*.

Pengujian *maintainability* menggunakan PHP Copy/Paste Detector untuk menguji duplikasi kode.

Pengujian aspek *reliability* menggunakan *tool* Wapt 9.3. Wapt 9.3 dapat digunakan untuk melakukan *stress testing*. *Stress testing* adalah suatu cara untuk menguji kehandalan suatu *web* dengan memberi beban lebih pada *web* tersebut dan menguji seberapa kuat *web* berjalan dalam kondisi tertentu dalam waktu tertentu.

### Teknik Analisis Data

Pengukuran untuk aspek *functional suitability* menggunakan skala Guttman. Menurut Sugiyono (2013:139), data yang diperoleh dalam skala Guttman dapat berupa data interval atau rasio dikhotomi (dua alternatif). Hasil pengujian

*functional suitability* dihitung menggunakan rumus dari matriks *Feature Completeness* (Acharya dan Sinha, 2013). Matriks *Feature Completeness* adalah matriks untuk mengukur sejauh mana fitur yang ada di desain dapat benar-benar diimplementasikan. Rumus yang dimaksud sebagai berikut:

$$X = \frac{I}{P}$$

Keterangan :

P = Jumlah fitur yang dirancang

I = Jumlah fitur yang berhasil diimplementasikan

Instrumen *functional suitability* diisi dengan pernyataan “YA” dan “TIDAK”. Nilai P dalam pengujian ini merupakan jumlah semua fungsi yang dirancang berdasarkan analisis kebutuhan. Sedangkan nilai I adalah jumlah fungsi yang berhasil diimplementasikan atau dalam instrumen berarti yang berisi “YA”. Dalam pengujian ini perangkat lunak dinyatakan memenuhi aspek *functional suitability*, jika nilai X adalah 1.

Analisis data *performance efficiency* dilakukan dengan menghitung rata-rata skor semua halaman dan waktu respon yang diuji menggunakan aplikasi GTMetrix. GTMetrix akan langsung mengkategorikan nilai YSlow dan PageSpeed menjadi *grade* A, B, C, D, atau E. Sementara itu untuk waktu respon, menurut Jakob Nielsen(1993), 10 detik adalah batas waktu untuk menjaga perhatian pengguna pada halaman. Waktu respon dinyatakan baik jika kurang dari 10 detik.

Pengukuran yang digunakan dalam instrumen *usability* adalah dengan skala Likert. Menurut Sugiyono (2013:134), skala Likert digunakan untuk mengukur sikap, pendapat, dan persepsi seseorang atau sekelompok orang tentang fenomena sosial. Jawaban setiap item instrumen yang menggunakan skala Likert mempunyai gradasi dari sangat positif sampai sangat negatif.

SUS menggunakan lima skala dalam pengukurannya. Kelimanya merupakan gradasi dari sangat setuju sampai dengan sangat tidak setuju atas pernyataan yang diajukan. Untuk menghitung skor dalam SUS, pertama kali yang

harus dilakukan adalah menjumlahkan semua kontribusi skor dari setiap item pernyataan. Kontribusi skor setiap item adalah 0 sampai dengan 4. Untuk item ganjil yaitu 1, 3, 5, 7, dan 9 kontribusi skornya berlaku pernyataan paling positif atau dalam hal ini paling setuju bernilai 4 dan seterusnya sampai yang paling tidak setuju bernilai 0. Sedangkan untuk item genap berlaku sebaliknya yaitu pernyataan paling tidak setuju justru bernilai 4 dan seterusnya sampai yang paling setuju bernilai 0. Kemudian total skor yang diperoleh dari penjumlahan tiap item dikalikan 2,5. Sehingga nilai maksimal yang mungkin diperoleh adalah 100.

Menurut Sauro (2011), skor SUS dianggap di atas rata-rata jika di atas 68 dan dianggap buruk atau di bawah rata-rata jika di bawah 68. Hal ini didasarkan pada 500 studi bahwa skor rata-rata SUS adalah 68.

Dalam pengujian *portability*, *Web Compatibility Testing Framework* akan menghasilkan aplikasi terkelompok menjadi tiga yaitu *green application* (aplikasi berjalan sempurna, tidak ada cacat), *amber application* (perlu *minor tweak* supaya aplikasi berjalan dengan baik), dan *red application* (tidak kompatibel, perlu dilakukan perubahan yang menyeluruh di dalam kode program agar dapat berjalan dengan baik).

Aplikasi akan dianggap lolos dari pengujian aspek ini jika mampu terklasifikasi ke dalam *green application*.

Pengujian *security* menggunakan aplikasi OWASP ZAP. Sehingga hasil analisis data untuk aspek *security* diperoleh dari hasil pengujian menggunakan aplikasi OWASP ZAP. OWASP ZAP akan melakukan penetrasi kepada aplikasi kemudian melaporkan ancaman apa saja yang mungkin terjadi pada aplikasi dan juga tingkat resiko ancaman tersebut.

Analisis data untuk aspek *maintainability* menggunakan pengukuran berdasarkan hasil dari pengujian *code duplication* dari *source code* perangkat lunak yang diperoleh dengan *tool* PHP Copy/Paste Detector. Hasil dari pengujian *code duplication* tersebut menurut Jansen (2016), kemudian digunakan rumus:

$$\text{Skor} = \min(-30 \times \log_{10}(C) + 70, 100)$$

Keterangan :

$C$  = Persentase hasil pengukuran *code duplication*

Skor kualitas *code* perangkat lunak sebagai indikator penilaian aspek *maintainability* menurut Jansen (2016), dapat dipetakan sebagai berikut:

Tabel 1. Kategori Penilaian *Maintainability* (Jansen, 2016)

| Code Duplication | TQI Score | Category |
|------------------|-----------|----------|
| <= 0.22%         | >= 90%    | A        |
| <= 0.46%         | >= 80%    | B        |
| <= 1.00%         | >= 70%    | C        |
| <= 4.64%         | >= 50%    | D        |
| <= 10.00%        | >= 40%    | E        |
| > 10.00%         | < 40%     | F        |

Pengujian pada aspek *reability* dilakukan dengan skenario sistem berjalan pada 10 menit dan terdapat 20 pengguna yang mengakses secara bersamaan. Setelah melalui proses perekaman, verifikasi, dan tes, Wapt akan memberikan hasil yaitu *success* atau *failure*. Parameter yang diukur *tool* ini adalah *session*, *pages*, dan *hits*.

## HASIL PENELITIAN DAN PEMBAHASAN

### Analisis Kebutuhan

Analisis kebutuhan adalah tahapan pertama pada model pengembangan perangkat lunak *waterfall*. Pada tahapan ini pengembang perangkat lunak menganalisa hal-hal yang dibutuhkan oleh perangkat lunak secara intensif dan mendokumentasikannya. Analisis kebutuhan dibagi menjadi dua bagian yaitu analisis kebutuhan fungsional serta perangkat keras dan lunak.

### Desain

Pada tahapan desain dalam penelitian ini dihasilkan desain UML yang meliputi diagram *use case*, *activity*, *class*, dan *sequence*. Serta desain antarmuka dan basis data.

### Implementasi

Pada tahapan implementasi ini terjadi proses translasi desain yang telah dibuat pada tahap sebelumnya ke dalam bentuk perangkat lunak.

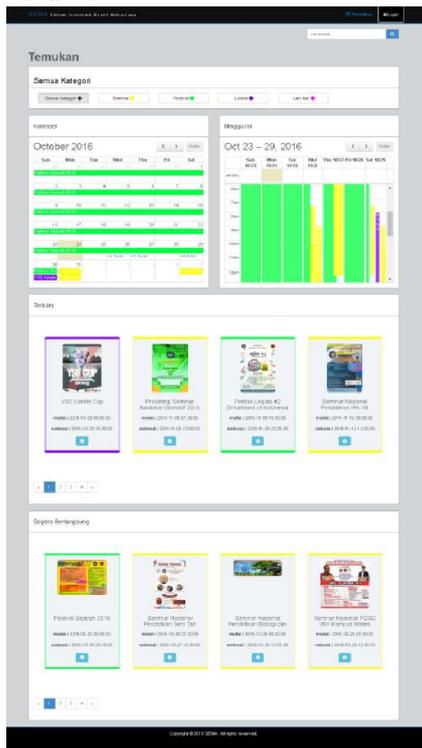
Sistem *Database* yang digunakan dalam Sistem Informasi *Event* Mahasiswa ini adalah MySQL. MySQL adalah sistem manajemen basis

data SQL yang bersifat *open source*. Database MySQL mendukung beberapa fitur seperti *multithreaded*, *multi-user*, dan *SQL Database Management System* atau DBMS. Berikut adalah hasil implementasi *database* pada aplikasi ini.

| Tabel           | Tindakan  | Baris | Jenis  | Penyartiran     | Ukuran | Beban |
|-----------------|---|-------|--------|-----------------|--------|-------|
| admin           | elajahi Struktur Cari Tambahkan Kosongkan Hapus | 1     | InnoDB | utf8_unicode_ci | 32 KB  |       |
| agenda          | elajahi Struktur Cari Tambahkan Kosongkan Hapus | 3     | InnoDB | utf8_unicode_ci | 32 KB  |       |
| cran_jnh        | elajahi Struktur Cari Tambahkan Kosongkan Hapus | 0     | InnoDB | utf8_unicode_ci | 32 KB  |       |
| cran_manager    | elajahi Struktur Cari Tambahkan Kosongkan Hapus | 0     | InnoDB | utf8_unicode_ci | 32 KB  |       |
| event           | elajahi Struktur Cari Tambahkan Kosongkan Hapus | 6     | InnoDB | utf8_unicode_ci | 32 KB  |       |
| fakultas        | elajahi Struktur Cari Tambahkan Kosongkan Hapus | 7     | InnoDB | utf8_unicode_ci | 32 KB  |       |
| groups          | elajahi Struktur Cari Tambahkan Kosongkan Hapus | 2     | InnoDB | utf8_unicode_ci | 32 KB  |       |
| jenis_kontak    | elajahi Struktur Cari Tambahkan Kosongkan Hapus | 5     | InnoDB | utf8_unicode_ci | 32 KB  |       |
| jumlah          | elajahi Struktur Cari Tambahkan Kosongkan Hapus | 1     | InnoDB | utf8_unicode_ci | 32 KB  |       |
| kategori_event  | elajahi Struktur Cari Tambahkan Kosongkan Hapus | 4     | InnoDB | utf8_unicode_ci | 32 KB  |       |
| komentar        | elajahi Struktur Cari Tambahkan Kosongkan Hapus | 13    | InnoDB | utf8_unicode_ci | 32 KB  |       |
| kontak          | elajahi Struktur Cari Tambahkan Kosongkan Hapus | 4     | InnoDB | utf8_unicode_ci | 32 KB  |       |
| kritik_sarun    | elajahi Struktur Cari Tambahkan Kosongkan Hapus | 1     | InnoDB | utf8_unicode_ci | 32 KB  |       |
| mahasiswa       | elajahi Struktur Cari Tambahkan Kosongkan Hapus | 7     | InnoDB | utf8_unicode_ci | 32 KB  |       |
| migrations      | elajahi Struktur Cari Tambahkan Kosongkan Hapus | 15    | InnoDB | utf8_unicode_ci | 32 KB  |       |
| password_resets | elajahi Struktur Cari Tambahkan Kosongkan Hapus | 3     | InnoDB | utf8_unicode_ci | 32 KB  |       |
| penyelenggara   | elajahi Struktur Cari Tambahkan Kosongkan Hapus | 7     | InnoDB | utf8_unicode_ci | 32 KB  |       |
| predi           | elajahi Struktur Cari Tambahkan Kosongkan Hapus | 2     | InnoDB | utf8_unicode_ci | 32 KB  |       |
| users           | elajahi Struktur Cari Tambahkan Kosongkan Hapus | 3     | InnoDB | utf8_unicode_ci | 32 KB  |       |

Gambar 2. Implementasi Database

Proses pengembangan Sistem Informasi *Event Mahasiswa* menggunakan *Sublime Text 2* dan *framework Laravel 5.2*. *Sublime Text 2* sendiri merupakan *cross-platform code editor*. *Sublime Text* mendukung banyak sekali bahasa pemrograman. Sementara itu *framework Laravel 5.2* memiliki kelebihan seperti *framework* ini sudah terintegrasi dengan *ORM (Object Relation Mapper)* yaitu *Eloquent* dan *Template Engine* yaitu *Blade*. Sehingga kita tidak perlu menginstal *ORM* seperti *Doctrine* ataupun *Template Engine* seperti *Twig* secara terpisah seperti di beberapa *framework* lain.



Gambar 3. Implementasi Sistem

### Pengujian

Pengujian pada aspek *functional suitability* menggunakan tiga orang pengujian yang terdiri dari dosen, mahasiswa, dan *software developer* profesional. Hasil dari pengujian adalah semua pengujian menyatakan semua fungsi yang total berjumlah 59 dalam instrumen penelitian berfungsi dengan benar.

Dengan diperolehnya hasil penelitian *functional suitability* maka dapat diketahui matriks *feature completeness*. Berdasarkan rumus matriks *feature completeness* maka hasil perhitungannya adalah sebagai berikut.

$$X = \frac{I}{P} = \frac{59}{59} = 1$$

Keterangan :

X = matriks *feature completeness*

P = Jumlah fitur yang dirancang

I = Jumlah fitur yang berhasil diimplementasikan

Pengujian pada aspek *performance efficiency* menggunakan *tool* dari *GTmetrix*. Sebelum memulai menganalisis performa *web*, *GTmetrix* akan membutuhkan pengaturan lokasi *server* dan *web browser*. Pengujian pada penelitian ini akan dilakukan dengan *setting* lokasi di *Hongkong, China* dan menggunakan *web browser* *Mozilla Firefox*. Berikut adalah hasil pengujian aspek *performance efficiency* dengan *GTmetrix*.



Gambar 4. Hasil Pengujian Dengan GTmetrix

Pengujian ini dilakukan di setiap halaman dalam aplikasi. Tabel di bawah ini menunjukkan hasil pengujian selengkapny.

Tabel 2. Hasil Pengujian *Performance Efficiency*

| No | Halaman              | Waktu Respon (s) | PageSpeed |       | YSlow |       |
|----|----------------------|------------------|-----------|-------|-------|-------|
|    |                      |                  | Skor      | Grade | Skor  | Grade |
| 1  | Temukan <i>Event</i> | 2.7              | 99        | A     | 82    | B     |

|                  |                      |      |    |   |    |   |
|------------------|----------------------|------|----|---|----|---|
| 2                | Login                | 3.0  | 99 | A | 82 | B |
| 3                | Registrasi           | 2.7  | 99 | A | 82 | B |
| 4                | Lupa Password        | 2.7  | 99 | A | 82 | B |
| 5                | Detail Event         | 2.9  | 99 | A | 82 | B |
| 6                | Profil Mahasiswa     | 2.7  | 99 | A | 82 | B |
| 7                | Agendaku             | 2.7  | 99 | A | 82 | B |
| 8                | Profil Penyelenggara | 2.8  | 99 | A | 82 | B |
| 9                | Event -ku            | 2.7  | 99 | A | 82 | B |
| 10               | Tambah Event         | 2.9  | 99 | A | 82 | B |
| 11               | Ubah Event           | 2.8  | 99 | A | 82 | B |
| <b>Rata-rata</b> |                      | 2.78 | 99 |   | 82 |   |

Pengujian *usability* dilakukan dengan menggunakan kuesioner *System Usability Scale* atau SUS. Kuesioner ini diisi oleh 20 responden yang merupakan calon pengguna Sistem Informasi *Event* Mahasiswa. Hasil dari pengujian ini adalah skor SUS rata-rata 80,13.

Pengujian pada aspek *security* menggunakan *tool* yang bernama OWASP Zed Attack Proxy(ZAP). OWASP ZAP merupakan alat *penetration testing* untuk menemukan vulnerabilitas pada aplikasi *web*.

Pada pengujian aspek *security*, Sistem Informasi *Event* Mahasiswa masih terdapat empat peringatan keamanan. Satu merupakan peringatan keamanan dengan tingkat resiko *medium* sementara tiga yang lain memiliki tingkat resiko *low* atau rendah. Jadi tidak ada peringatan keamanan dengan tingkat resiko tinggi seperti *Cross Site Scripting(XSS)* atau *SQL Injection(SQLi)*.

Pengujian *portability* dilakukan dengan mengamati Sistem Informasi *Event* Mahasiswa yang dijalankan pada *web browser* berbeda. Kemudian hasil pengamatan diklasifikasikan menjadi *green*, *amber*, dan *red application* sesuai dengan kesempurnaan atau ketidakcacatan tampilan *user interface* setiap halaman. Pada

penelitian ini, *web browser* yang digunakan adalah Google Chrome, Opera, dan Mozilla Firefox. Hasil dari pengujian ini adalah perangkat lunak berjalan dengan lancar dan tanpa cacat pada ketiga *web browser*.

Pengujian pada aspek *maintainability* menggunakan bantuan *tool* PHP Copy/Paste Detector. *Tool* ini tidak secara langsung menyatakan nilai aspek *maintainability*. PHP Copy/Paste Detector hanya memberikan informasi *duplication code* pada *source code* aplikasi. Selanjutnya adalah dengan memasukkan nilai *duplication code* ke dalam rumus dari TIOBE Quality Indicator yang dikemukakan oleh Jansen(2016).

Nilai *duplication code* yang didapat dari aplikasi PHP Copy/Paste Detector adalah 0%. Berikut adalah perhitungan nilai aspek *maintainability*.

$Skor = \min(-30 \times \log_{10}(0) + 70, 100) = \text{tidak terdefinisi}$

Pengujian pada aspek *reliability* dilakukan dengan *tool* Wapt 9.3 dengan skenario pengguna berjumlah 20 orang yang mengakses aplikasi secara bersamaan selama 10 menit. Dari hasil pengujian, skenario pada sistem dinyatakan *success*.

```

Test execution parameters:
Test status: finished
Test started at: 13/12/2016 14:05:52
Scenario name:
Test run comment:
Test executed by: Nanang (NANANG-PC)
Test executed on: NANANG-PC
Test duration: 0:10:00

Test result: SUCCESS

```

Gambar 5. Hasil Pengujian *Reliability*

## Pembahasan Hasil Penelitian

Berdasarkan hasil pengujian pada aspek *functional suitability* didapatkan bahwa semua fungsi yang diperoleh dari analisis kebutuhan fungsionalitas dinyatakan berjalan semua. Sehingga pada perhitungan matriks *feature completeness* diperoleh nilai X adalah 1. Sehingga dapat disimpulkan bahwa perangkat lunak telah memenuhi aspek *functional suitability*.

Pada pengujian kinerja perangkat lunak berdasarkan aspek *performance efficiency*, Sistem Informasi *Event* Mahasiswa mendapatkan 99 pada skor PageSpeed dan 82 pada YSlow. Dalam huruf, aplikasi mendapatkan *grade A* pada PageSpeed dan B pada YSlow. Sehingga

performa sistem dapat dikatakan baik. Sementara itu, rata-rata waktu respon aplikasi adalah 2,78 detik. Waktu tersebut masih kurang dari syarat minimal waktu respon 10 detik menurut Nielsen(1993). Sehingga dapat disimpulkan bahwa aplikasi telah memenuhi aspek *performance efficiency*.

Berdasarkan pengujian perangkat lunak pada aspek *usability* didapatkan rata-rata skor SUS sebesar 80,13. Skor tersebut telah melebihi skor rata-rata SUS menurut Sauro (2011). Sehingga dapat disimpulkan bahwa aplikasi telah memenuhi aspek *usability*.

Berdasarkan pengujian perangkat lunak pada aspek *security* menggunakan perangkat lunak OWASP ZAP didapatkan bahwa aplikasi memiliki beberapa peringatan keamanan dengan tingkat resiko rendah dan satu peringatan keamanan dengan tingkat resiko menengah. Perangkat lunak dinyatakan bebas dari ancaman dengan tingkat resiko tinggi seperti *SQL Injection* dan *Cross Site Scripting*.

Berdasarkan pengujian pada aspek *portability* ditemukan bahwa Sistem Informasi *Event Mahasiswa* dapat berjalan dengan baik pada *Google Chrome*, *Mozilla Firefox*, dan *Opera*. Sehingga dapat disimpulkan bahwa aplikasi memenuhi aspek *portability* pada ketiga *web browser* tersebut.

Pada pengujian pada berdasarkan aspek *maintainability*, perhitungan nilai aspek *maintainability* adalah tidak terdefinisi. Hal ini dikarenakan nilai *code duplication* adalah 0. Namun merujuk pada tabel 5, aplikasi dinyatakan masuk pada *grade A*. Hal tersebut dikarenakan nilai *code duplication* 0% memenuhi syarat masuk *grade* tersebut yaitu kurang dari atau sama dengan 0,22%. Sehingga dapat disimpulkan aplikasi memenuhi aspek *maintainability*.

Pada pengujian pada berdasarkan aspek *realibility*, hasil laporan dari pengujian dengan skenario 20 pengguna mengakses aplikasi secara bersamaan dalam waktu 10 menit adalah *success*. Sehingga dapat disimpulkan aplikasi memenuhi aspek *realibility*.

## SIMPULAN DAN SARAN

### Simpulan

Berdasarkan hasil penelitian dan pembahasan maka peneliti dapat mengambil kesimpulan sebagai berikut:

Penelitian ini menghasilkan Sistem Informasi *Event Mahasiswa* di Universitas Negeri Yogyakarta. Aplikasi ini memiliki fungsi untuk membantu penyelenggara *event* menyampaikan informasi *event* kepada mahasiswa. Pengembangan aplikasi menggunakan model *waterfall* dan dikembangkan dengan *framework* *Laravel*.

Aplikasi diuji dengan ISO/IEC 25010 berdasar aspek *functional suitability*, *performance efficiency*, *usability*, *maintainability*, *portability*, dan *security*. Pada aspek *functional suitability* diperoleh bahwa semua fungsi telah berjalan dengan baik dan benar. Pada aspek *performance efficiency* diperoleh bahwa aplikasi mendapatkan skor *PageSpeed* adalah 99 dan *YSlow* adalah 82 dengan waktu respon rata-rata halaman adalah 2,78 detik yang ada di bawah waktu respon maksimal 10 detik menurut Nielsen(1993). Pada aspek *usability* diperoleh rata-rata skor angket SUS dari 20 responden adalah 80,13 yang berarti di atas rata-rata skor SUS yaitu 68. Pada aspek *maintainability* diketahui dengan *code duplication* sebesar 0% aplikasi termasuk dalam kategori A yang berarti sangat baik. Pada aspek *portability* diketahui aplikasi dapat berjalan baik dan tanpa cacat pada *Google Chrome*, *Mozilla Firefox*, dan *Opera* maka aplikasi dapat dikategorikan sebagai *Green Application*. Pada aspek *security*, aplikasi dinyatakan bebas dari kerentanan dengan tingkat resiko tinggi seperti *XSS* dan *SQLi*. Pada aspek *realibility*, aplikasi dinyatakan sukses dalam pengujian dengan skenario 20 pengguna mengakses aplikasi secara bersamaan dalam 10 menit.

### Saran

Penulis memberikan saran untuk pengembangan aplikasi yang akan datang sebagai berikut:

1. Halaman *web* dibuat lebih responsif lagi sehingga dapat dengan baik berjalan pada

ukuran layar yang lebih baik seperti pada ponsel pintar.

2. Pengujian aspek *portability* dilakukan dengan melibatkan lebih banyak *web browser pc* maupun *mobile*.

#### DAFTAR PUSTAKA

Acharya, Anal dan Devadatta Sinha. (2013). Assessing the Quality of M-Learning Systems using ISO/IEC 25010. *International Journal of Advanced Computer Research*, 3(3).

Brooke, John. *SUS – A Quick and Dirty Usability Scale*. Diakses tanggal 19 Mei 2016 dari <http://hell.meiert.org/core/pdf/sus.pdf>

Brooke, John. (2013). SUS: A Retrospective. *Journal of Usability Studies*. Vol. 8, Issue 2

International Organization for Standardization. (2011). *ISO/IEC 25010*. Diakses pada tanggal 07 Januari 2016 dari [http://www.iso.org/iso/catalogue\\_detail.htm?csnumber=35733](http://www.iso.org/iso/catalogue_detail.htm?csnumber=35733)

Jansen, Paul. (2016). *The TIOBE Quality Indicator*. Diakses tanggal 26 Mei 2016 dari [http://www.tiobe.com/files/TIOBEQualityIndicator\\_v3\\_8.pdf](http://www.tiobe.com/files/TIOBEQualityIndicator_v3_8.pdf)

Kaalra, B., & Gowtman, L. (2014). Browser Compatibility Testing Using Manual Testing Methods and Test Tools. *International Journal of Advanced Studies in Computer Science and Engineering*, Volume 3, Issue 10

Nielson, Jacob. (2012). *How Many Test Users in a Usability Study?*. Diakses pada tanggal 23 Mei 2016 dari <https://www.nngroup.com/articles/how-many-test-users/>

Pressman, Roger. S. (2012). *Rekayasa Perangkat Lunak Buku I : Pendekatan Praktisi (Edisi 7)*. Yogyakarta : ANDI.

Sauro, Jeff. (2011). *Measuring Usability with the System Usability Scale (SUS)*. Diakses pada 01 Maret 2016 dari <http://www.measuringu.com/sus.php>

Menyetujui,  
Penguji Utama,



Dessy Irmawati, M.T.  
NIP. 19791214 201012 2 002

Yogyakarta, 6 Januari 2017

Dosen Pembimbing,



Nurkhamid, Ph.D.  
NIP. 19680707 199702 1 001