

# PENGEMBANGAN WEB DIREKTORI KARYA MAHASISWA DENGAN IMPLEMENTASI DATA VERSIONING

## DEVELOPMENT OF STUDENTS' PRODUCTS WEB DIRECTORY WITH DATA VERSIONING IMPLEMENTATION

Oleh: Gagas Akar Ilalang<sup>1</sup>, Handaru Jati<sup>2</sup>, Nurkhamid<sup>3</sup>  
Prodi Pendidikan Teknik Informatika Fakultas Teknik Universitas Negeri Yogyakarta<sup>1,2,3</sup>  
gagas.akar2015@student.uny.ac.id<sup>1</sup>, handaru@uny.ac.id<sup>2</sup>, nurkhamid@uny.ac.id<sup>3</sup>

### Abstrak

Penelitian ini bertujuan untuk (1) Mengembangkan produk web direktori karya yang memiliki komponen *data versioning*, (2) Menjamin kualitas produk yang dikembangkan sesuai standar kualitas ISO/IEC 25010:2011. Pengembangan produk dilaksanakan mengikuti model pengembangan Rational Unified Process. Pengujian kualitas produk menggunakan 6 dari 8 aspek *product quality model* ISO/IEC 25010:2011. Hasil dari penelitian ini adalah: (1) Sistem berbasis web yang dikembangkan, bernama Fineprint, telah memenuhi kebutuhan platform *showcase* karya dan pencatatan data karya dengan fitur *data versioning*. (2) Sistem telah teruji dengan ISO/IEC 25010:2011 pada masing-masing aspek sebagai berikut: *functional suitability* 98% (sebagian besar fitur berhasil diimplementasikan); *performance efficiency* sub-aspek performa 88% (baik/cepat), sub-aspek struktur 96% (sangat baik); *usability* 85% (sangat layak); *reliability* 82% (sangat layak) dan 98% tahan tekanan; *security* 0 celah keamanan (sangat aman); *maintainability* menunjukkan bahwa *source code* sedikit kompleks dengan skor CCN 14,16 namun masih relatif mudah dipelihara dengan skor MI 91,96, Halstead 0,39, dan Kan's defects 0,86.

Kata kunci: direktori karya, data versioning, ISO/IEC 25010:2011

### PENDAHULUAN

Mahasiswa selama pembelajarannya dalam perguruan tinggi, akan menemui mata kuliah yang mengharuskan mahasiswa membuat karya ilmiah atau produk tertentu sebagai persyaratan tugas akhir mata kuliah, sehingga setiap mahasiswa mempunyai karya. Karya yang dibuat dapat mengambil dua bentuk, yaitu tertulis (fisik) dan non-tulis (non-fisik); hal tersebut terutama berlaku untuk mahasiswa program studi Pendidikan Teknik Informatika (PTI) yang karyanya dapat berupa karya tulis dari mata kuliah umum, dan produk digital dari mata kuliah kejuruan. Karya juga dapat berupa produk sebagai hasil hobi tertentu mahasiswa. Karya-karya tersebut umumnya tidak dipublikasi/dicatat dan hilang, karena hanya dibuat untuk memenuhi persyaratan saja, padahal karya bisa dijadikan portofolio oleh mahasiswa untuk masa depan, terutama untuk mencari kerja.

Direktori karya dapat dibuat untuk mengatasi keterbatasan platform untuk publikasi

karya. Direktori, atau *listing*, sederhananya berarti daftar. Terdapat banyak jenis *listing* berbasis web untuk mempermudah akses *online* oleh banyak orang. Salah satu contoh direktori adalah situs *marketplace* (jual-beli) seperti ThemeForest yang berisi tema tampilan *website*, atau Tokopedia yang berisi berbagai jenis produk untuk dijual. Sebuah direktori karya dapat menjadi portofolio bagi mahasiswa.

Pengembangan *software* dapat dilakukan dengan tahap yang berurutan (*waterfall*) dan dengan tahap yang berulang (*iterative*). Pengembangan berulang lebih fleksibel karena perubahan kebutuhan bisa muncul di tengah pengerjaan proyek. Salah satu model pengembangan berulang adalah Rational Unified Process (RUP), yang terdiri dari 4 tahap, yaitu: (1) *Inception*, yang berisi identifikasi kebutuhan dan persiapan proyek; (2) *Elaboration*, yang berisi penjabaran kebutuhan dan penyusunan desain & arsitektur sistem; (3) *Construction*, yang berisi pembuatan tiap komponen sistem sesuai desain dan testing internal; (4) *Transition*, yang berisi

pemasangan sistem secara operasional untuk digunakan dan dites oleh *stakeholders*. RUP menggunakan Unified Modeling Language (UML), terutama *use case*, untuk merepresentasikan spesifikasi kebutuhan serta arsitektur & desain sistem (Wazlawick, 2013: 7).

Sebuah platform yang kontennya berasal dari pengguna (*user-generated content*) harus memiliki moderasi yang memungkinkan tindakan dilakukan terhadap konten yang tidak pantas. Selain itu, sistem platform tersebut juga harus bisa mencatat tidak hanya data terbaru, namun data yang lama juga, sehingga ketika ada pengguna 'nakal' yang ingin menghilangkan jejaknya dengan mengubah data, hal tersebut akan ketahuan. Untuk itu, sistem harus mencatat setiap perubahan yang terjadi pada data, supaya asli atau tidaknya data dapat dipastikan. Pencatatan dapat dilakukan dalam bentuk *data versioning*, yaitu penyimpanan setiap perubahan data yang diberi nomor versi sebagai representasi keadaan data pada waktu tertentu. Implementasi *data versioning* dapat dilakukan dengan banyak cara, salah satunya yaitu dengan membandingkan setiap perubahan dengan data aslinya, lalu menyimpan perbedaan data (*diff*).

Pengembangan produk *software* tidak akan lepas dari *bug* atau *error*. Selalu terdapat kemungkinan kecacatan produk bisa lolos sampai *software* sudah operasional dan digunakan *stakeholders*, walaupun setiap fitur atau komponen yang sudah dikembangkan telah dites satu-persatu pada tahap pengembangan. Untuk menghindari kegagalan operasional, maka *software* harus diuji mengikuti model kualitas *software* yang sudah standar sehingga sebanyak mungkin kecatatan dapat dieliminasi sebelum masuk fase operasional. Salah satu standar kualitas *software* yang dapat digunakan untuk pengujian adalah *product quality model* dalam ISO/IEC 25010:2011.

Salah satu penelitian yang relevan dengan tema direktori karya adalah pengembangan produk Karyaku oleh Bintang Muhammad (2018). Karyaku adalah *website* yang juga berperan sebagai direktori karya mahasiswa, namun belum memiliki fitur/komponen pencatatan perubahan

data. Selain itu, Karyaku juga belum diuji pada sisi aspek kualitas *security*.

Berdasarkan permasalahan yang telah diuraikan, peneliti merumuskan masalah penelitian sebagai berikut: (1) Bagaimana pengembangan produk yang dapat mengatasi keterbatasan platform untuk publikasi karya mahasiswa? (2) Bagaimana implementasi pencatatan perubahan data pada direktori karya untuk mendukung keaslian data? (3) Bagaimana pengujian produk direktori karya supaya kualitasnya terjamin untuk menghindari kegagalan operasional?

Tujuan penelitian ini lahir dari permasalahan penelitian yang telah dirumuskan. Tujuan penelitian ini adalah: (1) Mengembangkan produk web direktori karya mahasiswa sebagai platform publikasi karya mahasiswa; (2) Menambahkan implementasi komponen *data versioning* ke direktori karya sebagai pencatatan perubahan data; (3) Menjamin kualitas produk *software* yang dikembangkan dengan melakukan pengujian menggunakan *product quality model* ISO/IEC 25010:2011.

## METODE PENELITIAN

### Jenis Penelitian

Penelitian ini adalah *Research and Development* (R&D). R&D terdiri dari aktivitas pengembangan dan penelitian produk hasil pengembangan. Penelitian pengembangan dipilih karena produk tidak hanya dikembangkan saja, namun hasilnya juga perlu dipastikan kualitasnya dengan penelitian supaya memenuhi standar.

### Waktu dan Tempat Penelitian

Tempat dilaksanakannya penelitian ini adalah Universitas Negeri Yogyakarta. Penelitian dilakukan pada semester gasal tahun ajaran 2022/2023; pengembangan pada bulan Juli sampai November 2022, dan pengujian produk pada bulan November sampai Desember 2022.

*Security* berisi seberapa aman *website* secara teknis. *Maintainability* berisi perkiraan seberapa mudah kode program dapat dipelihara.

## Target/Subjek Penelitian

Subjek penelitian ini adalah mahasiswa UNY yang masih kuliah sejumlah minimal 20 orang, dengan lebih banyak mengambil mahasiswa pada program studi Informatika. Alasan diambilnya lebih banyak mahasiswa Informatika adalah untuk mendapatkan data karya yang lebih beragam (tulis/non-tulis).

Selain mahasiswa, tenaga profesional juga menjadi subjek penelitian sebagai penguji kualitas produk yang bersifat lebih teknis. Subjek tersebut akan disebut ahli bidang, sejumlah minimal 3 orang, dan diambil dari tenaga profesional yang telah bekerja di berbagai perusahaan.

## Prosedur

Penelitian dibagi menjadi dua kegiatan utama, yaitu pengembangan produk dan pengujian kualitas produk.

Pengembangan produk mengikuti model RUP, yang membagi pengembangan *software* ke dalam empat tahapan. Tahap *inception* berisi pengumpulan kebutuhan sistem dan perencanaan awal proyek. Tahap *elaboration* berisi penjabaran kebutuhan sistem, penentuan arsitektur sistem, dan desain keseluruhan sistem. Tahap *construction* berisi sebagian besar aktivitas pembuatan kode program (*coding*) sekaligus tes fungsi tiap fitur yang dibuat; selain itu, implementasi *data versioning* dilakukan pada tahap ini. Tahap *transition* berisi pemasangan sistem supaya dapat diakses secara *online*. Implementasi komponen *data versioning* didesain pada tahap *elaboration*, dan dieksekusi pada tahap *construction*.

Pengujian kualitas produk yang telah dikembangkan mengikuti 6 dari 8 aspek kualitas *product quality model* ISO/IEC 25010:2011. *Functional suitability* berisi pencacahan tiap fungsi sistem dan apakah fungsi tersebut berjalan dengan benar. *Performance efficiency* berisi pengukuran performa *website* dari segi waktu proses (ukuran *website* dan lamanya eksekusi *script*) dan dari segi struktur (kesesuaian kode *website* dengan *best practices* umum). *Usability* berisi seberapa baik kegunaan *website* dirasakan oleh pengguna. *Reliability* berisi seberapa nyaman penggunaan *website* tanpa *error* atau *down*.

## Teknik dan Instrumen Pengumpulan Data

Observasi pada saat pengembangan produk dilakukan untuk mengumpulkan data mengenai fitur dan fungsi produk yang akan dikembangkan. Objek observasi untuk mengumpulkan fungsi-fungsi utama adalah *website* ThemeForest dan produk Karyaku.

Kuesioner berisi pertanyaan yang dijawab oleh responden mengenai pengalaman mereka dalam penggunaan sebuah produk. Kuesioner disuguhkan melalui media *online* (Google Forms). Kuesioner digunakan untuk menilai bagaimana produk tampil dan bekerja di mata pengguna.

Produk sistem web diuji menggunakan *software* (disebut *tools*) untuk menilai aspek kualitas tertentu. *Tools* digunakan untuk menyimulasikan pengujian yang membutuhkan pengulangan yang sangat banyak.

### 1. Instrumen Uji *Functional Suitability*

Pengujian aspek kualitas *functional suitability* dilakukan menggunakan kuesioner yang berisi pencacahan tiap fungsi di sistem dan apakah masing-masing fungsi berjalan dengan benar menurut pengguna. Instrumen pengumpulan data berupa kuesioner yang berisi 52 item fungsi sistem dengan jawaban ya/tidak (skala Guttman) yang diujikan oleh ahli bidang.

### 2. Instrumen Uji *Performance Efficiency*

Pengujian aspek kualitas *performance efficiency* dilakukan menggunakan 5 *tools* yaitu PageSpeed Insights, Pingdom Website Speed Test, GTmetrix, Yellow Lab Tools, dan Geekflare Website Audit Tool. Hasil pengujian masing-masing *tool* dibagi menjadi 2 sub aspek, yaitu performa (segi waktu) dan struktur (segi teknis kode).

### 3. Instrumen Uji *Usability*

Pengujian aspek kualitas *usability* dilakukan menggunakan kuesioner yang berisi

pertanyaan seputar pengalaman penggunaan *website*. Kuesioner yang digunakan adalah kuesioner USE buatan Lund (2001) yang berisi 30 item kuesioner mengenai kegunaan sistem dan kepuasan pengguna, dengan 5 opsi jawaban (skala Likert). Kuesioner ini diujikan oleh mahasiswa.

#### 4. Instrumen Uji Reliability

Pengujian aspek kualitas *reliability* dilakukan menggunakan kuesioner yang berisi pertanyaan mengenai pengalaman penggunaan sistem, terutama seputar *error* yang ditemui dan apakah sistem *down*. Kuesioner berisi 12 item keandalan sistem dengan 5 opsi jawaban (skala Likert). Kuesioner ini diujikan oleh mahasiswa. Selain menggunakan kuesioner, aspek ini juga diuji menggunakan *tool* WAPT untuk melakukan *stress testing*.

#### 5. Instrumen Uji Security

Pengujian aspek *security* dilakukan menggunakan kuesioner yang berisi pertanyaan seputar keamanan sistem, di antaranya otentikasi, otorisasi, ketahanan terhadap serangan, dan *logging*. Instrumen pengumpulan data berupa kuesioner yang berisi 10 item keamanan sistem dengan jawaban ya/tidak (skala Guttman) yang diujikan oleh ahli bidang. Aspek ini juga diuji menggunakan 2 *tools*, yaitu Wapiti dan OWASP ZAP, untuk memeriksa kerentanan sistem terhadap *XSS* dan *SQL injection*.

#### 6. Instrumen Uji Maintainability

Pengujian aspek kualitas *maintainability* dilakukan menggunakan *tool* PhpMetrics, yang akan menghitung *maintainability codebase* dalam beberapa metrik, di antaranya *Halstead complexity*, *cyclomatic complexity number (CCN)*, *Kan's defects*, dan *Maintainability Index (MI)*.

### Teknik Analisis Data

#### 1. Kuesioner Functional Suitability dan Security

Kuesioner pengujian aspek *functional suitability* dan *security* menyediakan opsi jawaban

ya dan tidak (skala Guttman); jawaban ya diberi skor 1, dan jawaban tidak diberi skor 0. Untuk mendapatkan hasil skor akhir kualitas, total skor akan dibandingkan dengan skor maksimal, kemudian disuguhkan dalam bentuk persen.

$$X = \frac{S}{P \times R} \times 100\%$$

Dengan:

X : Kualitas

S : total skor jawaban semua responden

P : total jumlah pertanyaan

R : jumlah responden

Aspek *functional suitability* yang hasil kualitasnya 100% berarti sistem tersebut berhasil mengimplementasikan semua fitur dengan benar. Aspek *security* yang hasil kualitasnya 100% berarti sistem dapat dianggap aman digunakan dan aman dari serangan.

#### 2. Kuesioner Usability dan Reliability

Kuesioner pengujian aspek *usability* dan *reliability* menyediakan opsi jawaban dengan 5 skala (skala Likert); tiap opsi jawaban diberi skor 1 sampai 5. Untuk mendapatkan hasil skor akhir kualitas, total skor akan dihitung dan dibandingkan dengan skor maksimal/sem sempurna, kemudian disuguhkan dalam bentuk persen.

$$\text{Kualitas} = \frac{\text{Skor}_{\text{Total}}}{P \times R \times 5} \times 100\%$$

Dengan:

Skor<sub>Total</sub> : Skor total jawaban dari semua kuesioner responden

P : Jumlah pertanyaan dalam satu kuesioner

R : Jumlah responden

Setelah didapatkan skor persen, skor tersebut akan dikonversi menjadi nilai kualitatif sebagaimana diuraikan pada Tabel 1.

Tabel 1. Konversi Nilai Usability & Reliability ke Nilai Kualitatif

Nilai Kualitas	Interpretasi Nilai
0% - 20%	Sangat Tidak Layak
21% - 40%	Kurang Layak
41% - 60%	Cukup Layak
61% - 80%	Layak
81% - 100%	Sangat Layak

### 3. Hasil Data Tools Performance Efficiency

Hasil pengujian 5 tools pada aspek kualitas *performance efficiency* dibagi ke dua sub aspek, yaitu performa dan struktur. Nilai performa diambil dari rata-rata hasil performa semua (5) tools dalam bentuk persen. Nilai struktur diambil dari rata-rata hasil struktur 3 tools, yaitu PageSpeed, GTmetrix, dan Yellow Lab Tools, dalam bentuk persen.

### 4. Hasil Data Tools Reliability

Hasil pengujian tool WAPT pada aspek kualitas *reliability* adalah sebuah laporan yang detail dari *stress testing* yang dilakukan. Dari laporan tersebut, diambil data pada bagian *summary*, khususnya *successful hits* dan *failed hits*. Nilai hasil kualitas yang diambil adalah perbandingan antara *successful hits* dengan *total hits* (*successful hits* + *failed hits*) dalam bentuk persen. Nilai yang lebih dari 95% berarti bagus, karena hanya sedikit *request* yang ditolak atau tidak dapat ditangani server.

### 5. Hasil Data Tools Security

Hasil pengujian 2 tools pada aspek kualitas *security* adalah laporan yang detail mengenai keamanan *website*. Dari laporan tersebut, diambil 2 kategori penilaian, yaitu *XSS* dan *SQL injection*. Karena satu celah keamanan saja cukup untuk mengancam keamanan seluruh *website*, maka hasil penilaian berupa ya/tidak; *website* dinyatakan aman apabila terdapat nol celah keamanan, dan selain itu tidak aman. Untuk keperluan pengujian tools, fitur *captcha* dimatikan terlebih dahulu untuk memungkinkan dilakukan tes secara massal.

### 6. Hasil Data Tools Maintainability

Hasil pengujian tool PhpMetrics pada aspek kualitas *maintainability* adalah beberapa laporan yang cukup lengkap, dan laporan yang diperhatikan adalah laporan *complexity*. Beberapa skor penilaian diambil dari laporan tersebut. Skor *Cyclomatic complexity number* (CCN) memperkirakan seberapa kompleks *codebase* yang diuji, dan akan dikonversi menjadi nilai kualitatif sesuai dengan Tabel 2. Skor *Maintainability index*

(MI) memperkirakan kemudahan pemeliharaan *codebase*; semakin tinggi semakin baik, dan nilai MI di atas 80 sudah termasuk baik. Nilai Halstead memperkirakan rerata jumlah *bug*; semakin rendah semakin baik, dan Halstead yang kurang dari 5 masih bersifat baik. Nilai Kan's defects memperkirakan jumlah *defect*/kecacatan; semakin rendah semakin baik, dan Kan's defects yang kurang dari 5 masih bersifat baik.

Tabel 2. Rentang Kualitatif Nilai CCN

Rentang CCN	Deskripsi Nilai
1-10	Mudah dipahami dan mudah dites
10-20	Lebih kompleks, testing menjadi sedikit sulit karena banyak kemungkinan/percabangan
20-50	Sangat kompleks, pemeliharaan akan menjadi sangat sulit
50 ke atas	Terlalu kompleks, hampir mustahil dipelihara

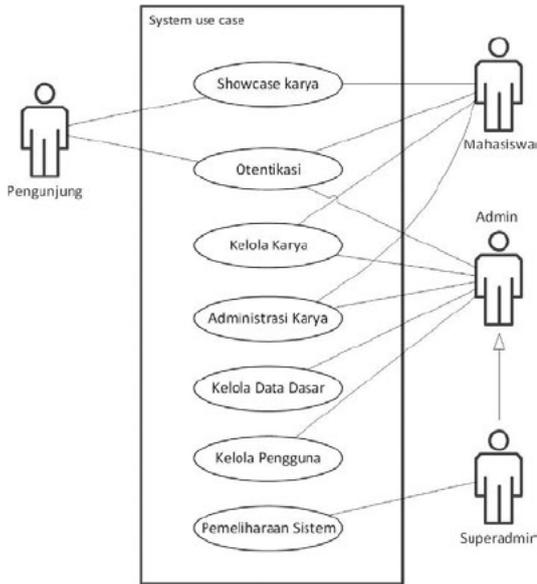
## HASIL PENELITIAN DAN PEMBAHASAN

### Pengembangan Produk

Sesuai dengan model pengembangan RUP, pengembangan produk dibagi menjadi empat tahap.

#### 1. Tahap Pengembangan Inception

Tahap *inception* meliputi *business modeling* dengan melakukan observasi pada sistem Karyaku untuk mendapatkan mengidentifikasi poin-poin kebutuhan yang penting. Observasi menghasilkan spesifikasi bahwa secara garis besar, sistem yang akan dibuat mencakup fitur yang ada pada Karyaku, yaitu utamanya pengelolaan karya mahasiswa. Sebagai tambahan, untuk dapat memenuhi karakteristik *security* dan menjadi aman, sistem harus dapat mencatat aktivitas perubahan data karya serta menyediakan proses verifikasi data oleh admin sebelum karya menjadi publik. Dari spesifikasi kebutuhan yang telah didapatkan, disusun *system use case* yang mencakup semua fungsi sistem dalam bentuk visual, terlihat pada Gambar 1.



Gambar 1. System Use Case

Pada tahap *inception* juga ditentukan teknologi apa saja yang akan digunakan untuk membuat sistem, yang diuraikan pada Tabel 3.

Tabel 3. Teknologi yang Digunakan Sistem

No	Komponen	Teknologi/Library
1	PHP framework	Laravel 6
2	Dependency manager	Composer 2 Node Package Manager (npm) 8
3	Website theme	Panel Admin – AdminLTE 3.0.1 Portal – Bootstrap 4.4.1
4	JS framework	jQuery 3.6.0
5	Icon pack	Font Awesome 5

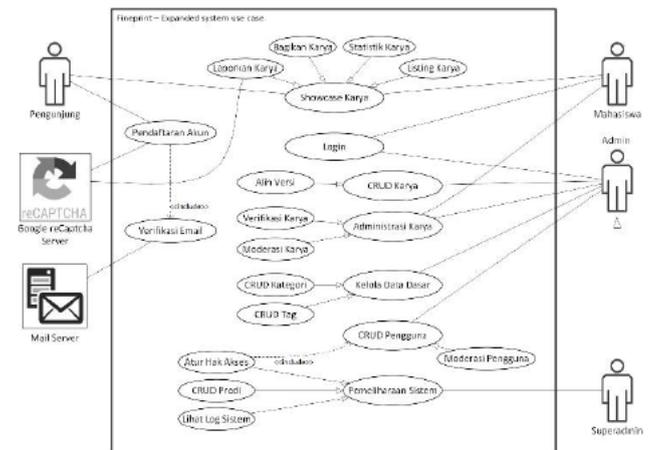
2. Tahap Pengembangan *Elaboration*

Tahap *elaboration* menjabarkan kebutuhan proyek dan mematangkan fondasi sistem sebelum memasuki proses implementasi yang lebih intensif. Arsitektur sistem digambarkan menggunakan berbagai jenis UML seperti *use case*, *activity diagram*, *sequence diagram*, dan *entity relationship diagram* (ERD). Pada tahap ini, ditentukan bahwa sistem yang dibuat dinamai *Fineprint*, untuk mewakili jejak perubahan data karya. Aktor eksternal yang berinteraksi dengan sistem dijabarkan pada Tabel 4.

Tabel 4. Aktor Sistem

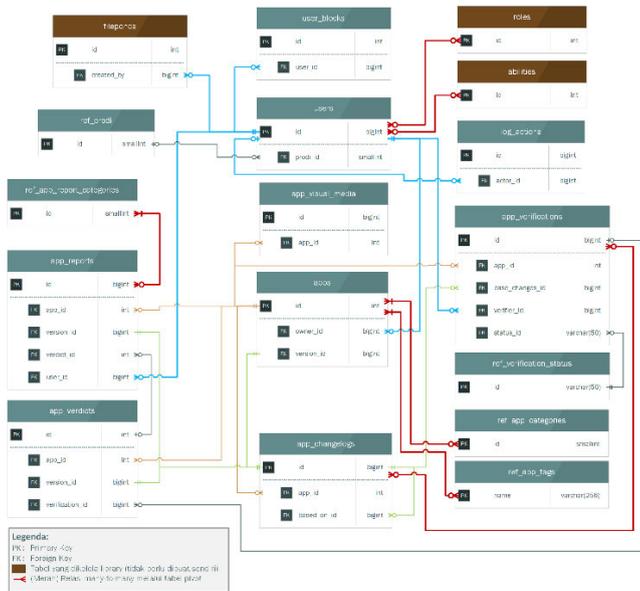
No	Aktor	Keterangan
1	Pengunjung	Pengunjung website yang melihat informasi publik seperti daftar karya dan statistik.
2	Mahasiswa	Mahasiswa dapat memasukkan data karya ke dalam sistem.
3	Admin	Admin adalah pengelola data level prodi. Admin bertugas verifikasi data dan meninjau karya terlapor.
4	Superadmin	Superadmin dapat menggunakan semua fitur yang ada. Aktor ini dimaksudkan untuk <i>maintenance</i> .
5	Layanan mail server	Sistem akan menggunakan layanan mail server eksternal untuk kebutuhan pengiriman email
6	Google reCaptcha Server	Layanan untuk melindungi form yang ada di halaman-halaman publik dari spam

*System use case* yang dibuat pada tahap sebelumnya juga dijabarkan menjadi lebih detail dengan sub-sub poin yang mewakili fungsi sistem dengan lebih terperinci. Penjabaran ini disebut *expanded use case*, dan terlihat pada Gambar 2.



Gambar 2. Expanded Use Case

Setelah semua kebutuhan dianalisis dan entitas-entitas dalam sistem ditentukan, dibuat *entity relationship diagram* (ERD) untuk menggambarkan hubungan antar entitas dalam sistem, terlihat pada Gambar 3. Diagram tersebut tidak mencantumkan semua atribut entitas supaya menjadi lebih *compact* dan hubungan antar entitas lebih mudah dipahami; spesifikasi atribut data tiap entitas dibuat pada diagram terpisah.



Gambar 3. ERD

### 3. Tahap Pengembangan *Construction*

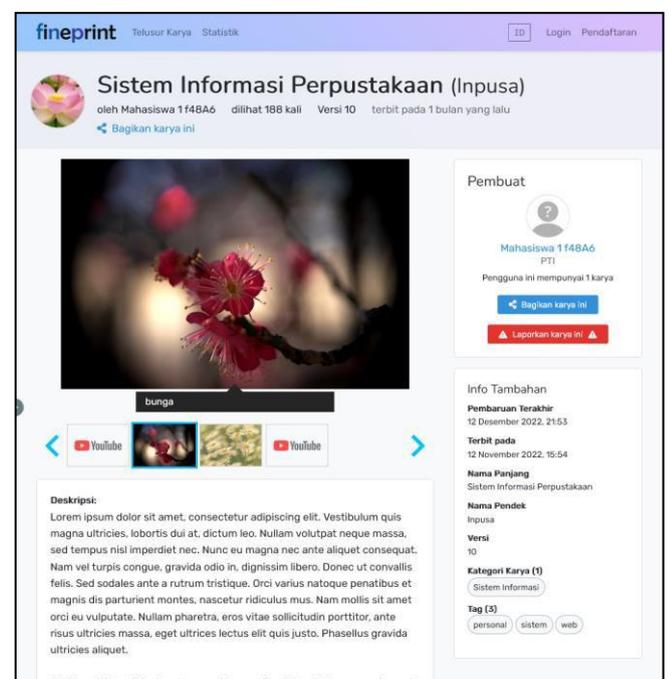
Tahap *construction* meliputi aktivitas *coding* sistem. Kode program dibuat dengan *framework* Laravel yang menganut pola MVC (*Model-View-Controller*). Singkatnya, Controller bertugas menerima *request* dari browser, Model adalah representasi entitas dari *database*, dan View adalah *file template* yang akan diisi konten.

Implementasi *data versioning* dilakukan pada tahap *construction*. Ketika terdapat data karya yang akan masuk ke *database*, baik menambah atau meng-edit karya, sistem akan mengumpulkan data yang masuk. Untuk karya baru, semua data yang masuk dianggap sebagai data baru; untuk edit data karya, sistem akan membandingkan data yang masuk dengan data karya, dan perbedaannya dianggap data baru. Data baru tersebut adalah *difference/diff* (perbedaan) data karya lama dengan data yang masuk. Perbedaan tersebut disebut *changelog*, dan akan diberi nomor versi kemudian disimpan.

Dalam *database*, data dalam tabel karya adalah data versi sekarang – data versi terakhir – yang ditampilkan dalam sistem sebagai data karya. Catatan mengenai tiap perubahan/versi karya disimpan dalam tabel khusus. Untuk menjaga keakuratan relasi data yang berhubungan dengan karya, maka verifikasi karya dan moderasi karya tidak menunjuk langsung ke data dalam tabel karya (yang berisi data paling mutakhir), melainkan menunjuk ke versi di waktu verifikasi

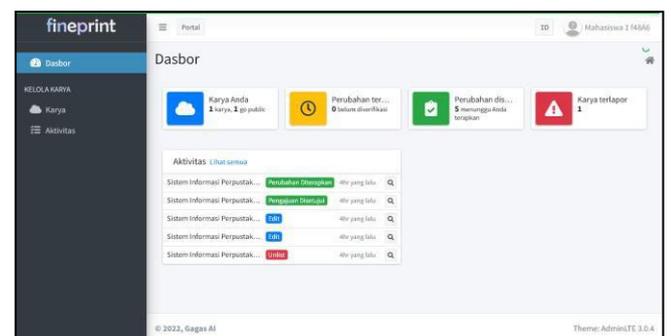
*Pengembangan Web Direktori.....(Gagas Akar Ilalang) 7* atau moderasi karya tersebut dilakukan. Dengan begitu, walaupun data sebuah karya telah mengalami perubahan dari waktu ke waktu, catatan verifikasi dan moderasi karya tetap dapat menampilkan data karya yang benar, yaitu data karya tersebut saat dulunya ditinjau.

Tampilan halaman karya publik, seperti yang ditampilkan Gambar 4, mengambil inspirasi dari halaman item pada ThemeForest. Nama karya dijadikan judul di atas, kemudian media visual serta deskripsi karya sebagai konten utama ditaruh di kiri, dan di bagian kanan ditampilkan informasi sekunder dan tersier mengenai karya.



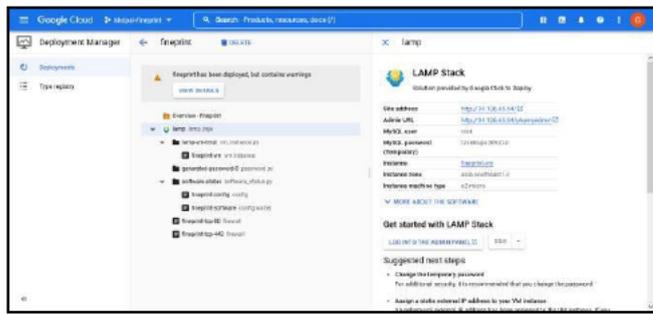
Gambar 4. Tampilan Halaman Karya Publik

Tampilan Panel Admin, seperti yang tampil pada Gambar 5, didesain menggunakan tampilan standar dari tema AdminLTE, yaitu dengan *navbar* di atas yang menampung navigasi, dan *sidebar* di kiri yang menampung menu.



Gambar 5. Tampilan Dasbor Panel Admin

Proses *deployment* juga dilakukan pada tahap ini, sebagai persiapan untuk digunakan pada tahap berikutnya. Untuk keperluan demonstrasi aplikasi, server yang digunakan untuk *hosting* adalah Google Cloud Platform (GCP) yang sebenarnya berbayar, namun setiap pengguna mendapatkan \$300 kredit gratis untuk *free trial*. Gambar 6 menampilkan panel administrasi di GCP. Jenis server yang digunakan adalah *LAMP stack* (Linux, Apache, MySQL, PHP) dengan satu IP publik gratis. Setelah server terinstal, Apache dikonfigurasi supaya dapat digunakan. Nama domain juga dibeli dari GCP, dan domain yang dipilih adalah [fineprint-ptiuny.xyz](http://fineprint-ptiuny.xyz).



Gambar 6. Administrasi LAMP Stack GCP

#### 4. Tahap Pengembangan *Transition*

Tahap *transition* menaruh fokus pada menyempurnakan *deployment* yang telah dilakukan di tahap sebelumnya. Konfigurasi Apache diperlengkap menurut *best practices* untuk menambah keamanan. Selain itu, protokol koneksi *website* di-*upgrade* menjadi HTTPS sebagai kriteria keamanan *website* paling minimal di jaman sekarang. Sertifikat HTTPS didapatkan gratis dari Let's Encrypt, dan *tool* yang digunakan untuk konfigurasi HTTPS adalah Certbot. Kemudian, konfigurasi Apache diatur supaya melarang membuka *website* menggunakan HTTP; koneksi yang menggunakan HTTP akan otomatis di-*redirect* ke HTTPS untuk mencegah serangan berbasis *man-in-the-middle* (MITM).

#### Hasil Uji Produk

##### 1. *Functional Suitability*

Aspek kualitas *functional suitability* diuji dengan kuesioner oleh 4 ahli bidang. Gambar 7 menampilkan ringkasan data yang didapatkan.

No.	Skor	No.	Skor	No.	Skor	No.	Skor
1	4	14	4	27	4	40	4
2	3	15	4	28	4	41	4
3	4	16	4	29	3	42	4
4	4	17	4	30	4	43	4
5	4	18	4	31	4	44	4
6	4	19	4	32	4	45	4
7	4	20	4	33	4	46	4
8	4	21	3	34	4	47	4
9	4	22	4	35	4	48	4
10	4	23	4	36	4	49	4
11	4	24	4	37	4	50	4
12	4	25	4	38	4	51	4
13	4	26	4	39	4	52	4

Jumlah Responden Ahli: 4  
 Jumlah Item Kuesioner: 52  
 Skala Jawaban Kuesioner: 0-1 (Guttman)  
 Total Skor: 205

Gambar 7. Skor Kuesioner *Functional Suitability*

$$X = \frac{205}{52 \times 4} \times 100\% = 98,55\%$$

Hasil X (kualitas) *functional suitability* bernilai 98% (dibulatkan ke bawah) yang berarti sebagian besar fitur yang dibuat sudah berjalan dengan benar. Utamanya, fitur yang dinilai tidak berjalan benar adalah fitur verifikasi email karena kecepatan pengiriman emailnya kadang terlalu lama dan membuat malas menunggu.

##### 2. *Performance Efficiency*

Aspek kualitas *performance efficiency* diuji dengan 5 *tools* dan dibagi ke dua sub aspek kualitas, yaitu performa dan struktur. Sebagai tolak ukur, *sample* halaman yang diuji adalah halaman yang akan memuat informasi/konten terbanyak, yaitu halaman item karya publik dari sisi Portal, dan halaman daftar karya dari sisi Panel Admin. Tabel 5 menampilkan hasil pengujian.

Tabel 5. Data *tools* uji *performance efficiency*

#	Halaman	Tool	Nilai	
			Performa	Struktur
1	Portal - halaman item karya publik	PageSpeed	86%	98%
		Pingdom	82%	-
		GTmetrix	88%	98%
		Yellow	94%	-
		Geekflare	81%	93%
2	Admin - daftar karya	GTmetrix	97%	95%
		Yellow	94%	-
Rata-rata*			88,86%	96%

\* ) Penilaian yang tidak tersedia tidak ikut dihitung

$$Kualitas = \frac{1478}{12 \times 30 \times 5} \times 100\% = 82,11\%$$

Rata-rata nilai performa adalah **88%** (dibulatkan ke bawah) yang berarti performa *website* tergolong baik; rata-rata nilai struktur adalah **96%** yang berarti struktur kode dan aksesibilitas *website* tergolong sangat baik.

### 3. Usability

Aspek kualitas *usability* diuji dengan kuesioner oleh 31 mahasiswa UNY, 21 di antaranya dari prodi Informatika (67%). Gambar 8 menampilkan ringkasan data yang didapatkan.

Ringkasan Skor Kuesioner Usability		
Jawaban	Skor	Jumlah Dijawab
STS	1	3
TS	2	20
RG	3	102
S	4	419
SS	5	386

Jumlah Responden Mahasiswa: 31  
 Jumlah Item Kuesioner: 30  
 Skala Jawaban Kuesioner: 1-5 (Likert)  
 Total Skor: 3955

Gambar 8. Skor Kuesioner *Usability*

$$Kualitas = \frac{3955}{30 \times 31 \times 5} \times 100\% = 85,05\%$$

Hasil kualitas *usability* bernilai **85%** (dibulatkan ke bawah), yang setelah dikonversi sesuai Tabel 1 artinya kualitas *usability website* termasuk sangat layak.

### 4. Reliability

Aspek kualitas *reliability* diuji dengan kuesioner dan *tool* WAPT. Kuesioner diisi oleh 30 mahasiswa UNY, 21 di antaranya dari prodi Informatika (70%). Gambar 8 menampilkan ringkasan data yang didapatkan.

Ringkasan Skor Kuesioner Reliability		
Jawaban	Skor	Jumlah Dijawab
STS	1	8
TS	2	12
RG	3	42
S	4	170
SS	5	128

Jumlah Responden Mahasiswa: 30  
 Jumlah Item Kuesioner: 12  
 Skala Jawaban Kuesioner: 1-5 (Likert)  
 Total Skor: 1478

Gambar 9. Skor Kuesioner *Reliability*

Hasil kualitas kuesioner *reliability* bernilai **82%** (dibulatkan ke bawah), yang setelah dikonversi sesuai Tabel 1 termasuk sangat layak, artinya kualitas *reliability website* menurut mahasiswa penguji adalah andal.

Tabel 6. Hasil uji WAPT

Kategori	Hasil
Successful hits	4624
Failed hits	50

$$Reliability = \left(1 - \frac{50}{4624}\right) \times 100\% = 98,92\%$$

Tabel 6 menampilkan hasil uji *tool* WAPT. Hasil perhitungan *tool reliability* bernilai **98%** (dibulatkan ke bawah), artinya menurut pengujian *tool* WAPT kualitas *reliability website* termasuk baik dan tahan tekanan (*stress resistant*).

### 5. Security

Aspek kualitas *security* diuji dengan kuesioner dan 2 *tools*. Kuesioner diisi oleh 4 ahli bidang. Gambar 10 menampilkan ringkasan data yang didapatkan.

Ringkasan Skor Kuesioner Security							
Kerahasiaan		Keutuhan		Akuntabilitas		Keaslian	
No.	Skor	No.	Skor	No.	Skor	No.	Skor
1	4	4	4	7	4	10	4
2	4	5	4	8	4		
3	4	6	4	9	4		

Jumlah Responden Ahli: 4  
 Jumlah Item Kuesioner: 10  
 Skala Jawaban Kuesioner: 0-1 (Guttman)  
 Total Skor: 40

Gambar 10. Ringkasan Skor Kuesioner *Security*

$$X = \frac{40}{10 \times 4} \times 100\% = 100\%$$

Hasil X (kualitas) *security* bernilai **100%** yang berarti menurut para ahli bidang, *website* sepenuhnya aman digunakan.

Tabel 7 dan Tabel 8 menampilkan hasil pengujian *tool* Wapiti dan OWASP ZAP.

Tabel 7. Hasil uji Wapiti

Kategori	Celah Ditemukan
SQL injection	
SQL injection	0
Blind SQL injection	0
Cross Site Scripting (XSS)	
Reflected Cross Site Scripting	0
Stored Cross Site Scripting	0

Tabel 8. Hasil uji OWASP ZAP

Kategori	Peringatan	False Positives
SQL injection	2	2
Cross Site Scripting (XSS)	0	0

Hasil pengujian kedua *tools security* adalah ditemukan nol celah keamanan, yang berarti *website* sepenuhnya aman dari serangan *SQL injection* dan *XSS*.

Dari hasil pengujian kuesioner dan *tools*, dapat disimpulkan *website* sangat aman digunakan dengan skor sempurna dalam hal keamanan.

## 6. Maintainability

Aspek kualitas *maintainability* diuji dengan menggunakan *tool* PhpMetrics, yang menghasilkan banyak laporan detail. Laporan yang diambil adalah laporan *complexity*, dan skor yang diambil dari laporan tersebut tersaji pada Tabel 9.

Tabel 9. Hasil uji *complexity* PhpMetrics

Metrik Penilaian	Skor
Average Maintainability Index (MI)	91,96 (max 220)
Average cyclomatic complexity number (CCN)	14,16
Average bugs (Halstead)	0,39
Average defects (Kan's defects)	0,86

Skor CCN bernilai 14,16 yang jika dikonversi sesuai Tabel 2 artinya *source code* sedikit kompleks. Skor MI bernilai 91,96 yang sudah termasuk baik. Skor Halstead adalah 0,39 yang termasuk rendah, artinya diperkirakan tidak banyak *bug* potensial. Skor Kan's defects adalah 0,86 yang termasuk rendah, artinya diperkirakan tidak banyak kecacatan. Dari semua skor tersebut dapat disimpulkan bahwa walaupun *source code*

*website* sedikit kompleks, namun masih relatif mudah untuk dipelihara.

## SIMPULAN DAN SARAN

### Simpulan

Berdasarkan hasil penelitian, dapat diambil kesimpulan yaitu: (1) Pengembangan menghasilkan *website* direktori karya bernama *Fineprint* yang mengatasi keterbatasan platform publikasi karya. Produk dibuat dengan *framework* Laravel 6 dan dikembangkan menggunakan model pengembangan RUP dengan empat tahap yaitu *inception*, *elaboration*, *construction*, dan *transition*. (2) *Fineprint* sudah mempunyai fitur *data versioning* sebagai pencatatan perubahan data karya sehingga membantu pelacakan perubahan data serta membantu memastikan keaslian data. (3) Sistem *Fineprint* telah memenuhi 6 dari 8 aspek kualitas *product quality model* ISO/IEC 25010:2011. Aspek *functional suitability* bernilai 98% yaitu sebagian besar fitur sudah diimplementasikan dengan benar. Pada aspek *performance efficiency*, sub-aspek performa bernilai 88% yaitu performa *website* baik dan prosesnya cepat, dan sub-aspek struktur bernilai 96% yaitu struktur kode *website* sangat baik. Aspek *usability* bernilai 85% yaitu sangat layak. Pada aspek *reliability*, keandalan bernilai 82% yaitu sangat layak, dan ketahanan tekanan bernilai 98% yaitu *website* tahan terhadap tekanan. Pada aspek *security*, hasil pengujian membuktikan bahwa *website* sangat aman digunakan dengan 0 celah keamanan. Aspek *maintainability* diuji dan hasilnya adalah *source code website* sedikit kompleks dengan skor CCN 14,16 namun masih relatif mudah untuk dipelihara dengan skor MI 91,96, Halstead 0,39, dan Kan's defects 0,86.

### Saran

Beberapa fitur dapat ditambahkan untuk membuat *website* direktori karya menjadi lebih baik. Salah satu fitur yang dapat ditambahkan adalah fitur sosial seperti *like/comment* untuk menghidupkan interaksi antar pengguna. Perlu untuk mencari layanan *mail server* yang prosesnya lebih cepat supaya pengguna yang mendaftarkan akun tidak terlalu lama menunggu sehingga malas menggunakan sistem.

## **DAFTAR PUSTAKA**

Lund, A. M. (2001). Measuring Usability with the USE Questionnaire. *Usability interface*, 8(2), 3–6.

Muhammad, B. (2018). *Pengembangan Aplikasi Web Karyaku Sebagai Direktori Karya Mahasiswa Universitas Negeri Yogyakarta*. Universitas Negeri Yogyakarta. Diambil dari <https://eprints.uny.ac.id/60044/>

Pressman, R. S., & Maxim, B. R. (2019).

*Software Engineering: A Practitioner's Approach*. McGraw-Hill Education.

Sommerville, I. (2016). *Software Engineering* (10th editi). Boston: Addison-Wesley.

Wazlawick, R. S. (2013). *Object-Oriented Analysis and Design for Information Systems: Modeling with UML, OCL, and IFML*. Morgan Kaufmann.