

RANCANG BANGUN PORTAL MAGANG BERBASIS WEBSITE MENGGUNAKAN NODE JS (STUDI KASUS: FASILKOM UNSIKA)

Wisnu Saputra, Kamal Prihandani

Informatika, Universitas Singaperbangsa Karawang

Jalan HS. Ronggowaluyo, Karawang, Indonesia

2010631170126@student.unsika.ac.id

ABSTRAK

Dalam lingkup Fakultas Ilmu Komputer Universitas Singaperbangsa Karawang, magang merupakan salah satu kegiatan wajib dan merupakan syarat bagi mahasiswa dalam melaksanakan tugas akhir. Namun, yang menjadi permasalahan saat ini adalah mahasiswa mengalami kesulitan ketika mencari tempat untuk melaksanakan kegiatan magang. Aplikasi portal magang berbasis website merupakan sebuah inovasi terkini yang dirancang untuk mempermudah mahasiswa dalam mencari tempat magang, mempermudah alur magang, dan membantu dalam menjembatani antara pihak mahasiswa dengan mitra perusahaan. Penelitian ini memiliki tujuan untuk melakukan rancang bangun portal magang berbasis website dengan menggunakan Node Js. Metode pengembangan yang diterapkan untuk membangun aplikasi ini merupakan metode *Extreme Programming* yang terdiri dari empat tahap yaitu *planning*, *design*, *coding*, dan *testing*. Untuk mempermudah pengembangan sistem, *Unified Modeling Language* (UML) digunakan dalam proses perancangan aplikasi yang akan dibangun. Sedangkan untuk implementasi sistem akan menggunakan bahasa pemrograman Javascript. Berdasarkan analisis masalah yang telah dilakukan menggunakan metode wawancara dan observasi, maka didapatkan beberapa fitur dan kebutuhan fungsional berdasarkan analisis kebutuhan pengguna yang sudah dibuat. Hasil pengujian dilakukan dengan menggunakan teknik kotak putih (white box) dan kotak hitam (black box) yang menyatakan bahwa aplikasi yang dibuat valid dan dapat berjalan dengan baik sesuai dengan kebutuhan yang diharapkan.

Kata kunci : Magang, Website, UML, Extreme Programming, Node Js

1. PENDAHULUAN

Tingkat pengangguran di Indonesia mengalami peningkatan sejak awal tahun 2020 akibat pandemi Covid-19. Kelompok yang paling terdampak adalah para *freshgraduate*. Menurut data Badan Pusat Statistik (BPS), tingkat pengangguran terbuka (TPT) pada usia 15-24 tahun mencapai tingkat tertinggi, yakni 20,46 persen dengan lulusan diploma dan sarjana menyumbang angka sebesar 12,33 persen [3]. Sulitnya mendapatkan pekerjaan bagi *freshgraduate* juga dikarenakan keterbatasan pengalaman kerja mereka. Rendahnya kemauan dan kesempatan para lulusan tersebut dalam mengikuti pelatihan kerja atau magang menjadi salah satu faktor utama yang menyulitkan mereka dalam mendapatkan pekerjaan.

Magang dianggap dapat menjadi sebuah solusi dalam menurunkan tingkat pengangguran di Indonesia. Hal ini dikarenakan program magang dapat memberikan pengalaman kerja praktis yang menjadikan seseorang mendapatkan pengalaman kerja secara langsung di dunia nyata.

Dalam lingkup Fakultas Ilmu Komputer Universitas Singaperbangsa Karawang, salah satu kewajiban untuk melaksanakan tugas akhir adalah setiap mahasiswa diwajibkan mengikuti kegiatan magang. Namun, yang menjadi permasalahan saat ini adalah mahasiswa mengalami kesulitan ketika mencari tempat untuk melaksanakan kegiatan magang.

Saat ini, Fakultas Ilmu Komputer Universitas Singaperbangsa Karawang belum menyediakan

website portal magang yang dapat memfasilitasi proses pencarian tempat magang bagi mahasiswa. Website yang ada saat ini hanya bersifat administratif dan tidak memadai untuk membantu mahasiswa dalam menemukan tempat magang yang sesuai dengan minat dan keahlian mereka.

Berdasarkan permasalahan tersebut, maka dibutuhkan sebuah aplikasi portal magang yang dapat membantu mahasiswa dalam pencarian tempat magang yang mudah untuk diakses melalui berbagai macam perangkat.

Node Js dipilih dalam pengembangan aplikasi portal magang berbasis website ini. Hal ini dikarenakan Node js dapat membangun sebuah aplikasi web yang responsif dan dinamis dengan menggunakan javascript di sisi *backend* dan *frontend* pada sistem yang dibangun. Node js sendiri merupakan suatu lingkungan server yang dapat diakses secara bebas dan dapat beroperasi pada berbagai platform.

2. TINJAUAN PUSTAKA

2.1. Portal

Portal merupakan suatu situs yang berperan sebagai tempat penyimpanan informasi di internet atau WWW. Sebuah portal berita di internet biasanya dirancang dengan tampilan yang interaktif dan desain yang beragam [1].

2.2. Website

Website merupakan sebuah media yang berfungsi untuk memperkenalkan, menampilkan, atau mencari informasi melalui internet [6]. Website terdiri dari beberapa halaman yang disatukan serta digunakan dalam menampilkan sebuah informasi dalam bentuk teks, suara, gambar atau *video*. Situs pada website modern terdiri dari tiga bagian fundamental yaitu HTML, CSS, dan Javascript [7].

2.3. Magang

Magang merupakan suatu bentuk pelatihan atau praktek yang bertujuan untuk mengembangkan keahlian khusus di bawah bimbingan dan pengawasan instruktur berpengalaman [2]. Program magang memegang peran penting dalam dunia pendidikan dan instansi/perusahaan. Khususnya bagi mahasiswa yang mengikuti magang, kegiatan ini memberikan manfaat berupa pengalaman langsung dalam atmosfer dunia kerja sebenarnya, meningkatkan kompetensi, dan mengasah keterampilan sesuai dengan kapasitas yang dimiliki [2].

2.4. Javascript

Javascript adalah sebuah bahasa pemrograman yang dapat digunakan untuk memanipulasi elemen HTML guna membuat halaman web menjadi interaktif. Javascript menambahkan aspek pemrograman ke dalam HTML dan CSS [7].

2.5. Node Js

Node.js merupakan suatu lingkungan server yang dapat diakses secara bebas dan dapat beroperasi pada berbagai platform, termasuk Linux, Windows, dan Mac. Lingkungan *run time* yang disediakan oleh Node.js memungkinkan eksekusi JavaScript di luar lingkungan peramban (*browser*) [5].

2.6. Software Development Life Cycle (SDLC)

SDLC merupakan kependekan dari *Software Development Life Cycle* yang merujuk pada proses pengubah atau pengembangan suatu sistem perangkat lunak. Dalam proses ini, digunakan metodologi dan model-model yang telah terbukti efektif, yang telah digunakan oleh para pengembang sistem perangkat lunak sebelumnya [8].

2.7. Extreme Programming

Extreme Programming (XP) merupakan suatu pendekatan dalam metodologi pengembangan perangkat lunak yang bersifat *agile* dengan menekankan pengkodean sebagai aktivitas utama pada setiap tahap siklus pengembangan perangkat lunak [4]. Metode XP menggunakan pendekatan iteratif yang dapat diulang sesuai kebutuhan proyek. *Extreme programming* menyediakan siklus pengembangan perangkat lunak yang singkat dan berulang untuk mencapai fokus yang ditetapkan. Metode XP sendiri terdiri dari beberapa tahapan seperti perencanaan, desain, pengkodean, dan pengujian [10].

2.8. Unified Modeling Language (UML)

Unified Modeling Language (UML) merupakan sebuah pemodelan standar dalam pengembangan perangkat lunak berdasarkan teknik *object oriented programming* (OOP). UML hadir untuk memenuhi kebutuhan visual dalam memahami, mendeskripsikan, membangun, serta mendokumentasikan sistem perangkat lunak [9].

2.9. Alpha Testing

Alpha testing adalah pengujian pada system atau aplikasi guna mengetahui aplikasi yang dibuat dapat berfungsi dengan baik berdasarkan kebutuhan yang diharapkan. *Alpha testing* berfokus terhadap kebutuhan fungsional perangkat lunak dan fungsi yang diuji merupakan tombol menu. Terdapat 2 pendekatan dalam pengujian menggunakan *alpha testing*, yaitu pengujian kotak putih dan pengujian kotak hitam [11].

3. METODE PENELITIAN

Metodologi penelitian yang digunakan pada penelitian ini merupakan metode *Extreme Programming*. Pengembangan model *Extreme Programming* melibatkan beberapa tahapan yang terstruktur seperti perencanaan, desain, pengkodean, dan pengujian.

3.1. Perencanaan (Planning)

Tahap perencanaan ini memiliki tujuan guna mengidentifikasi kebutuhan dalam pengembangan rancang bangun portal magang. Hasil dari analisis kebutuhan mengarah pada pemahaman tujuan utama pembuatan portal magang dan proses pemodelan sistem yang dibutuhkan.

3.2. Perancangan (Design)

Tahap perancangan ini bertujuan untuk membuat sebuah pemodelan dan alur kerja yang sesuai dengan kebutuhan fungsional. Dalam proses perancangan ini, peneliti menggunakan metode pemodelan UML dan menerapkannya melalui diagram urutan dan diagram kelas..

3.3. Pengkodean (Coding)

Pada tahap pengkodean ini, rancangan yang sudah di desain sebelumnya diubah kedalam sebuah program menggunakan bahasa pemrograman yang sesuai dengan kebutuhan sistem. Javascript dipilih sebagai bahasa pemrograman yang akan digunakan dalam membangun aplikasi ini. Selain javascript, terdapat beberapa teknologi pendukung dalam pengembangan sistem ini seperti node js, react js, express js, dan mysql sebagai database.

3.4. Pengujian (Testing)

Dalam penelitian ini, peneliti menerapkan metode pengujian *Alpha Testing* dengan menggunakan teknik pengujian *black box* (kotak hitam) dan *white box* (kotak putih).

4. HASIL DAN PEMBAHASAN

Hasil dari penelitian ini adalah suatu sistem portal magang berbasis website yang dapat membantu menjembatani antara mahasiswa dan mitra perusahaan dalam kegiatan magang, yang awalnya proses tersebut dilakukan secara manual, saat ini berinovasi dengan cara yang terkomputerisasi. Diharapkan melalui hasil penelitian ini dapat membantu dan bermanfaat bagi mahasiswa Fasilkom Unsika dalam mencari tempat untuk melaksanakan kegiatan magang.

4.1. Perencanaan (Planning)

Tahap perencanaan dilakukan dengan tujuan untuk mengetahui kebutuhan apa saja yang dibutuhkan oleh perangkat lunak dan pengguna. Tahap perencanaan dimulai dengan mengumpulkan data yang dibutuhkan untuk membangun aplikasi portal magang, berikut merupakan teknik pengumpulan data yang dilakukan:

a. Wawancara

Wawancara dilakukan langsung dengan salah satu tim magang Fasilkom Unsika yaitu Bapak Adhi Rizal, M.T. Adapun hasil dari wawancara tersebut adalah sebagai berikut

- Peneliti mendapatkan data-data yang terkait dengan sistem magang, pengelolaan magang, tata cara dan alur magang.
- Peneliti mendapatkan data-data mahasiswa Fasilkom yang sudah melaksanakan kegiatan magang dalam beberapa tahun terakhir sehingga dapat mengarahkan kepada suatu solusi untuk perbaikan atau pengembangan sistem.

b. Observasi

Observasi yang dilakukan pada penelitian ini yaitu peneliti melakukan observasi secara langsung terhadap website magang Fasilkom yang sudah berjalan, serta melakukan observasi mengenai alur magang dimulai dari tahap pendaftaran hingga kegiatan magang selesai dilaksanakan. Data yang diambil dari observasi ini adalah data umum mengenai magang Fasilkom Unsika dan data penunjang penelitian.

4.2. Analisis Kebutuhan User

Analisis kebutuhan user dilakukan dengan pengumpulan cerita awal dari calon pengguna yang dilakukan dengan cara wawancara secara terstruktur. Setelah analisis kebutuhan user dilakukan maka akan didapatkan data yang akan diolah ditahap selanjutnya pada penelitian ini. Berdasarkan hasil pengumpulan data maka didapatkan permasalahan yang dialami calon pengguna adalah sebagai berikut:

- Website magang yang berjalan saat ini hanya mencakup hal-hal yang bersifat administratif.
- Penyebaran informasi dan pendaftaran magang masih dilakukan secara manual melalui *group whatsapp* masing-masing program studi.

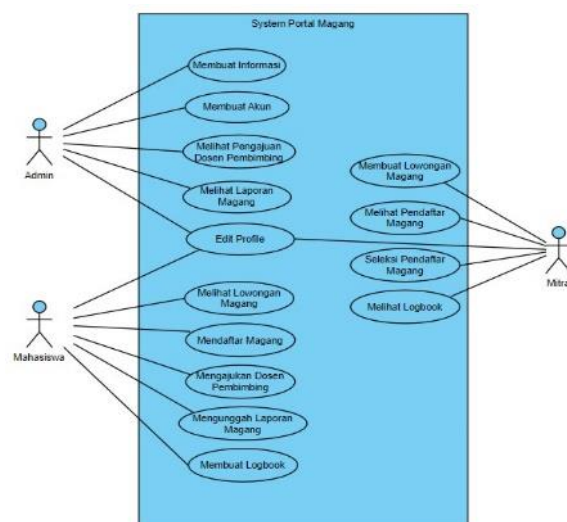
- Belum adanya sistem yang terintegrasi untuk menjembatani antara pihak mahasiswa dan mitra magang dalam proses pendaftaran magang.

4.3. Perancangan (Design)

Perancangan dilakukan guna mempermudah perancangan aplikasi yang akan dibangun. Desain aplikasi ini akan dibangun dengan menggunakan pemodelan UML. Terdapat beberapa diagram UML yang digunakan dalam tahap perancangan ini seperti diagram *use case*, diagram aktivitas, diagram *sequence*, dan diagram kelas.

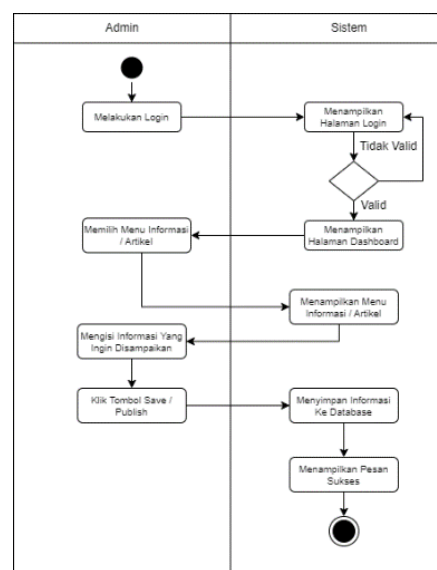
4.4. Use Case Diagram

Use Case Diagram mendeskripsikan interaksi antara aktor dengan sistem yang akan dibuat, seperti ditunjukkan pada Gambar 1 di bawah ini:



Gambar 1. Use Case Diagram Sistem Portal Magang

4.5. Activity Diagram Membuat Informasi

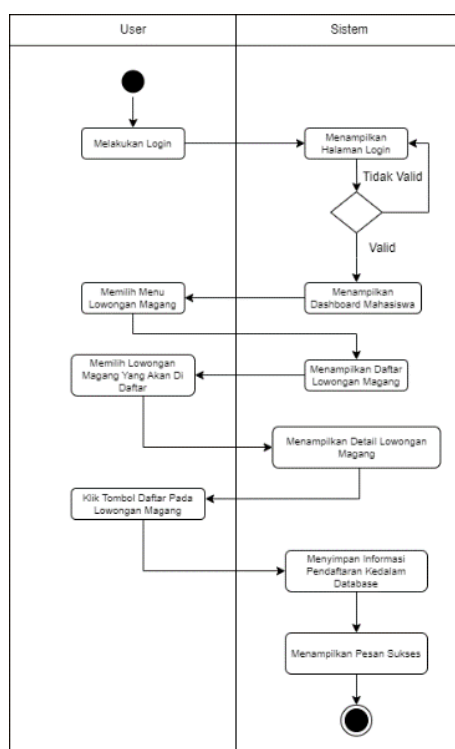


Gambar 2. Activity Diagram Membuat Informasi

Gambar 2 menunjukkan aktivitas untuk membuat informasi kegiatan magang. Admin akan memilih menu informasi, kemudian admin mengisi informasi yang akan disampaikan, setelah admin menekan tombol publish maka sistem akan secara otomatis menyimpan data kedalam database.

4.6. Activity Diagram Mendaftar Magang

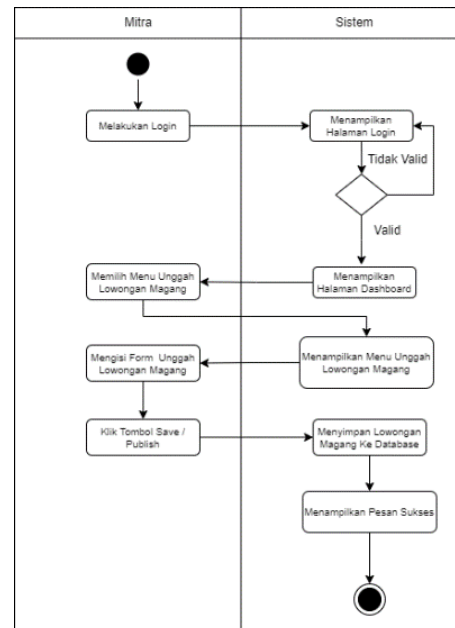
Gambar 4 menunjukkan aktivitas yang dilakukan oleh mahasiswa ketika mendaftar magang. Mahasiswa akan mengakses menu lowongan magang, kemudian mahasiswa akan menekan tombol detail untuk melihat detail magang. Pada halaman detail magang mahasiswa akan menekan tombol daftar, setelah itu sistem akan menyimpan pendaftaran tersebut kedalam database.



Gambar 4. Activity Diagram Mendaftar Magang

4.7. Activity Diagram Upload Lowongan Magang

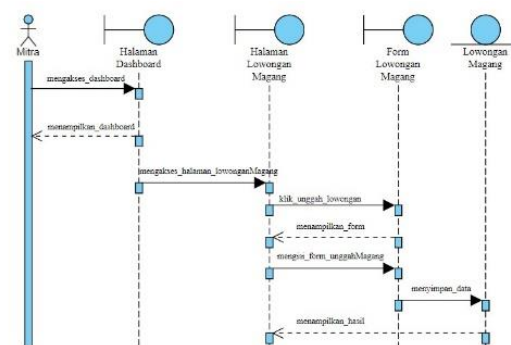
Gambar 5 menunjukkan aktivitas yang dilakukan oleh mitra dalam membuat lowongan magang. Setelah melakukan login, mitra akan mengakses halaman lowongan magang, setelah itu sistem akan menampilkan halaman lowongan magang. Mitra selanjutnya akan mengisi form upload lowongan magang dengan sesuai, setelah mitra menekan tombol create maka sistem akan menyimpan informasi kedalam database.



Gambar 5. Activity Diagram Upload Lowongan Magang

4.8. Sequence Diagram Membuat Informasi

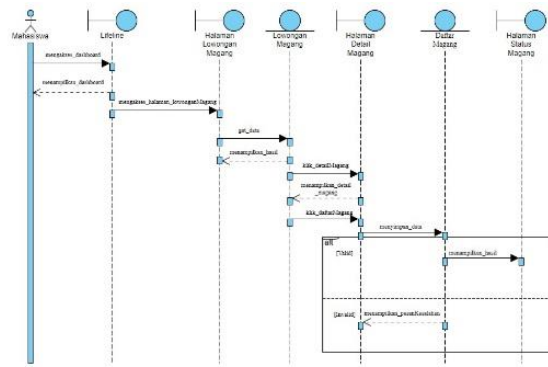
Gambar 6 menunjukkan alur pada saat admin akan membuat sebuah informasi, admin akan mengakses menu informasi. Setelah itu admin akan mengisi informasi yang akan disampaikan pada halaman tersebut, sistem akan menyimpan informasi yang dibuat oleh admin kedalam database, apabila berhasil maka sistem akan menampilkan pesan sukses.



Gambar 6. Sequence Diagram Membuat Informasi

4.9. Sequence Diagram Mendaftar Magang

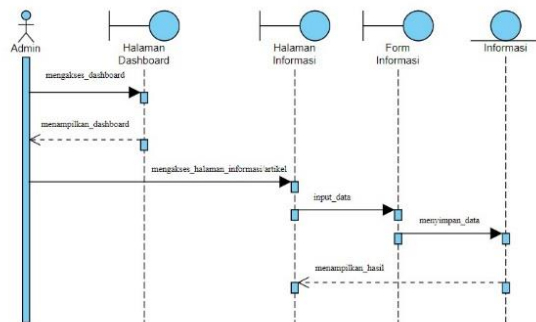
Gambar 7 menunjukkan alur pada saat mahasiswa akan mendaftar magang, maka mahasiswa akan mengakses menu lowongan magang, kemudian sistem akan menampilkan halaman tersebut. Setelah itu mahasiswa akan melihat detail magang dan melakukan pendaftaran pada lowongan magang tersebut, sistem akan menyimpan informasi pendaftaran yang dilakukan oleh mahasiswa kedalam database, apabila berhasil maka sistem akan menampilkan pesan sukses.



Gambar 7. Sequence Diagram Mendaftar Magang

4.10. Sequence Diagram Upload Lowongan Magang

Gambar 8 menunjukkan alur pada saat mitra melakukan *upload* lowongan magang, mitra akan mengakses menu lowongan magang, kemudian sistem akan menampilkan halaman tersebut. Mitra selanjutnya akan mengisi *form upload* lowongan magang, sistem akan menyimpan informasi lowongan magang tersebut kedalam database, apabila berhasil maka sistem akan menampilkan pesan sukses.



Gambar 8. Sequence Diagram Upload Lowongan Magang

4.11. Class Diagram



Gambar 9. Class Diagram Sistem Portal Magang

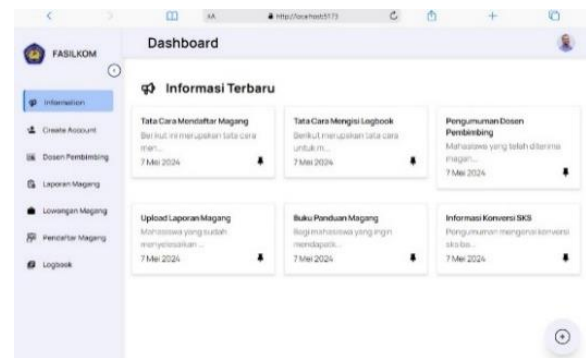
Perancangan database dilakukan setelah tahap perancangan sistem. Perancangan database ini meliputi *class diagram* dan struktur data, dan diimplementasikan menggunakan MySQL. Proses ini

mencakup pembentukan *class diagram* yang kemudian diterjemahkan menjadi tabel-tabel yang saling terkait. *Class diagram* yang telah dibuat dapat dilihat pada Gambar 9.

4.12. Pengkodean (Coding)

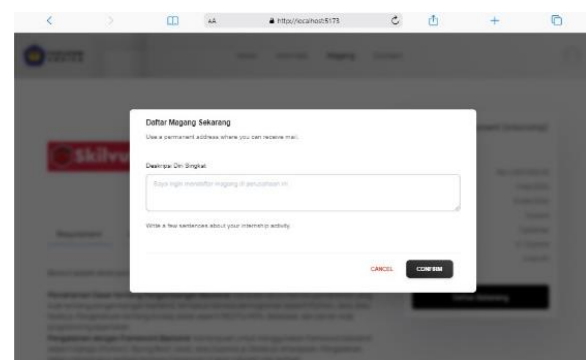
Berdasarkan desain yang telah dibuat, selanjutnya akan diimplementasikan kedalam suatu program perangkat lunak yang hasilnya merupakan sebuah website portal magang, program ini dibuat dengan menggunakan JavaScript, React Js, Express Js, dan MySQL sebagai databasenya. Berikut merupakan tampilan portal magang berbasis website yang telah dibuat:

- Halaman Membuat Informasi / Artikel
Gambar 10 merupakan halaman informasi atau artikel yang digunakan untuk membuat informasi mengenai segala kegiatan magang.



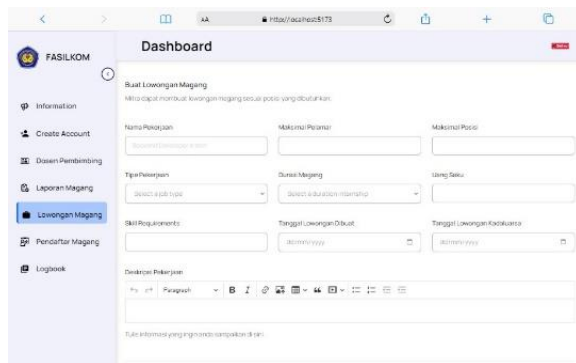
Gambar 10. Halaman Membuat Informasi / Artikel

- Halaman Mendaftar Magang
Gambar 11 merupakan halaman daftar magang yang digunakan untuk melihat detail lowongan magang dan melakukan pendaftaran pada lowongan magang tersebut.



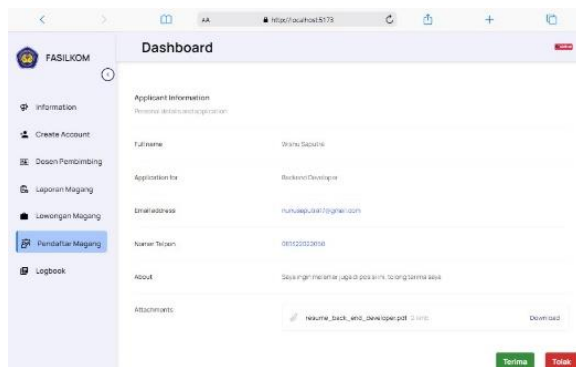
Gambar 11. Halaman Mendaftar Magang

- Halaman Upload Lowongan Magang
Gambar 12 merupakan halaman lowongan magang yang digunakan untuk membuat lowongan magang bagi mitra.



Gambar 12. Halaman Upload Lowongan Magang

- d. Halaman Seleksi Pendaftar Magang
Gambar 13 merupakan halaman seleksi pendaftar magang yang digunakan untuk melakukan seleksi terhadap pendaftar magang tersebut.



Gambar 13. Halaman Seleksi Pendaftar Magang

4.13. Pengujian (Testing)

Pengujian dilakukan menggunakan dua metode yaitu pengujian kotak putih dan pengujian kotak hitam. Pada pengujian kotak putih akan dilakukan dengan menguji logika program pada sistem yang sudah dibuat. Sedangkan pada pengujian kotak hitam akan dilakukan dengan menguji fungsionalitas pada sistem tersebut.

4.14. White Box Testing

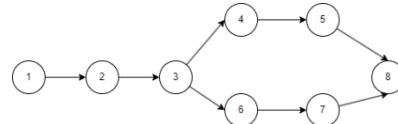
White Box Testing adalah metode pengujian dengan menganalisis dan memeriksa kode program untuk mendeteksi kesalahan. Metode ini menguji logika dalam program yang dibuat. Berikut ini adalah pengujian yang telah dilakukan:

- a. *White Box* Membuat Informasi

Tabel 1. *White Box* Membuat Informasi

1	<code>createInfo: async (req, res) => {</code>
2	<code>const { title, author, desc } = req.body;</code>
3	<code>try {</code>
4	<code>const data = await info.create({</code> <code>title, author, desc, });</code>
5	<code>res.status(201).json({</code> <code>message: "Information has successfully created",</code> <code>data: data,</code> <code>});</code>

6	<code>} catch (error) {</code>
7	<code>res.status(500).json({</code> <code>message: error.message,</code> <code>});</code>
8	<code>},</code>

Gambar 14. *Flowgraph* Membuat Informasi

$$\begin{aligned}
 V(G) &= E - N + 2P \\
 &= 8 - 8 + 2 * 1 \\
 &= 2
 \end{aligned}$$

Berdasarkan *flowgraph* pada gambar 4.62, maka diketahui:

Node (N) = 8 (Lingkaran)

Edge (E) = 8 (Tanda Panah)

Path (P) = 1 (Jumlah komponen terhubung)

Maka berdasarkan basis *path* yang telah dilakukan, didapatkan *cyclomatic complexity* dari kode tersebut adalah 2 *path* (jalur)

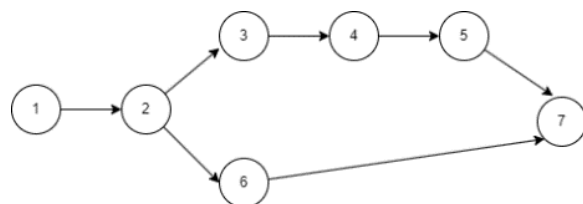
Path 1 = 1 - 2 - 3 - 4 - 5 - 8

Path 2 = 1 - 2 - 3 - 6 - 7 - 8

- b. *White Box* Upload Lowongan Magang

Tabel 2. *White Box* Upload Lowongan Magang

1	<code>addJobs: async (req, res) => {</code>
2	<code>try {</code>
3	<code>const {</code> <code>jobTitle, maxApplicants, maxPositions,</code> <code>jobType, salary, skillSet, duration, jobPost,</code> <code>deadline, desc, } = req.body;</code>
4	<code>const job = await Jobs.create({</code> <code>jobTitle, maxApplicants, maxPositions,</code> <code>jobType, salary, skillSet, duration, jobPost,</code> <code>deadline, desc, userId: req.userId, });</code>
5	<code>res.status(201).json({</code> <code>message: "Success create a new job",</code> <code>data: job,</code> <code>});</code>
6	<code>} catch (error) {</code> <code>res.status(500).json({</code> <code>message: "Internal server error",</code> <code>});</code>
7	<code>},</code>

Gambar 15. *Flowgraph* Upload Lowongan Magang

$$\begin{aligned}
 V(G) &= E - N + 2P \\
 &= 7 - 7 + 2 * 1 \\
 &= 2
 \end{aligned}$$

Berdasarkan *flowgraph* pada gambar 4.64, maka diketahui:

Node (N) = 7 (Lingkaran)
 Edge (E) = 7 (Tanda Panah)
 Path (P) = 1 (Jumlah komponen terhubung)

Maka berdasarkan basis *path* yang telah dilakukan, didapatkan *cyclomatic complexity* dari kode tersebut adalah 2 *path* (jalur)

Path 1 = 1 – 2 – 3 – 4 – 5 – 7

Path 2 = 1 – 2 – 6 – 7

Berdasarkan hasil pengujian menggunakan *white box testing* maka dapat ditarik kesimpulan bahwa

kode program telah diuji secara menyeluruh dari segi struktur dan logika, memastikan bahwa setiap jalur eksekusi telah diperiksa dan fungsi-fungsi yang terkait telah diuji dengan benar. Dari hasil pengujian ini memberikan keyakinan bahwa perangkat lunak telah diuji secara efektif dari sudut pandang pengembang kode, dan dapat meningkatkan kepercayaan terhadap keandalan dan kualitas sistem secara keseluruhan.

4.15. Black Box Testing

Black Box Testing adalah metode pengujian yang mengevaluasi fungsionalitas sistem yang telah dikembangkan. Berikut ini adalah tabel pengujian sistem yang dilakukan menggunakan *black box testing*.

Tabel 3. *Black Box Testing*

Halaman Yang Diuji	Skenario Pengujian	Test Case	Hasil Yang Diharapkan	Hasil Pengujian
Halaman Informasi / Artikel	Mengisi seluruh <i>form</i> informasi / artikel dengan benar	<i>title</i> : Informasi Konversi SKS <i>author</i> : Admin Fasilkom <i>desc</i> : Pengumuman tata cara konversi SKS	Sistem berhasil menyimpan data informasi yang dibuat kedalam database, dan memberikan pesan sukses pada halaman informasi / artikel	Sesuai Harapan
Halaman Lowongan Magang	Menekan tombol <i>create</i> lowongan magang dan mengisi seluruh <i>form</i> lowongan magang dengan benar, kemudian menekan tombol <i>create</i>	<i>job title</i> : Backend Developer <i>job type</i> : Internship <i>salary</i> : Rp. 2.000.000 <i>max applicants</i> : 10 <i>max positions</i> : 8 <i>duration</i> : 3 bulan <i>job description</i> : lowongan <i>backend</i>	Data berhasil disimpan kedalam database dan sistem akan menampilkan pesan sukses pada halaman lowongan magang	Sesuai Harapan
Halaman Pendaftar Magang	Melakukan seleksi pendaftar magang	Mengakses halaman pendaftar magang dan menekan tombol detail pada salah satu pendaftar, kemudian klik <i>button accept</i> pada halaman pendaftar magang	Sistem akan menyimpan status perubahan pendaftar magang kedalam database dan menampilkan pesan sukses pada halaman pendaftar magang	Sesuai Harapan
Halaman Daftar Magang	Mendaftar pada salah satu lowongan magang	Menekan tombol detail pada salah satu lowongan magang, kemudian menekan tombol daftar magang dan mengisikan <i>form</i> yang tersedia pada halaman detail magang	Data berhasil disimpan kedalam database dan sistem akan menampilkan pesan sukses pada halaman lowongan magang	Sesuai Harapan

Berdasarkan hasil pengujian menggunakan *black box testing*, maka dapat disimpulkan bahwa sistem telah diuji secara menyeluruh dari perspektif fungsionalitas pengguna, memastikan bahwa setiap fitur dan fungsionalitas telah berjalan sesuai dengan yang diharapkan. Kesimpulan ini memberikan keyakinan bahwa sistem telah diuji secara efektif dari sudut pandang pengguna akhir, dan dapat meningkatkan tingkat kepercayaan terhadap kinerja dan pengalaman pengguna sistem secara keseluruhan.

5. KESIMPULAN DAN SARAN

Dari hasil penelitian yang dilakukan, dapat ditarik kesimpulan bahwa perancangan portal magang berbasis website ini dapat menampilkan hasil yang

akurat bagi pengguna, yaitu sistem dapat membantu mahasiswa dalam pencarian tempat magang secara lebih mudah dan efektif. Sistem yang dibangun ini menggunakan metode *Extreme Programming* dan memiliki fungsionalitas yang baik berdasarkan hasil pengujian menggunakan *black box testing*, sehingga sistem ini dapat berjalan dengan baik saat digunakan oleh pengguna akhir. Saran dari penulis untuk meningkatkan kualitas aplikasi portal magang berbasis website ini adalah dengan menambahkan fitur notifikasi untuk informasi terbaru, lowongan magang terbaru, dan pengumuman kelulusan magang, serta fitur *helpdesk* dan lupa password untuk memudahkan pengguna dalam mengatasi masalah akses.

DAFTAR PUSTAKA

- [1] Andria, R. and Pamungkas, "Evaluasi Kualitas Web Portal Fakultas Teknik Unipma Dengan Metode Mccall," *Association for Information Systems – Indonesia chapter (AISINDO)*., 3(2), 2018.
- [2] Azwar, E, "Program Pengalaman Lapangan (Magang) Terhadap Kepercayaan Diri Mahasiswa Pendidikan Jasmani dan Rekreasi," *Jurnal Penjaskesrek.*, Vol. 6, No. 2, 211-221, 2019.
- [3] BPS, "Berita Resmi Statistik: Keadaan Ketenagakerjaan Indonesia Agustus," Jun. 17, 2023. Available: <https://www.bps.go.id/pressrelease/2020/11/05/1673/agustus-2020--tingkatpengangguran-terbuka--tpt--sebesar-7-07-persen.html/>
- [4] Gumelar, T. Astuti, R, and Sunarni, A. T, "Sistem Penjualan Online Dengan Metode Extreme Programming," *Jurnal TELEMATIKA MKOM.*, 2017.
- [5] Node, "About Node.js," Jun. 17, 2023. Available: <https://nodejs.org/en/about/>
- [6] Nurkhozin, M. Basir, A. and Aznar A.M, "Sistem Informasi Sekolah Berbasis Web Sebagai Media Promosi di Smk Muhammadiyah 2 Paguyangan," *Jurnal Teknik Informatika Dan Sistem Informasi (JURTISI)*., 2(2), 96-106, 2022.
- [7] Pratama, A, *Javascript Uncover – Panduan Javascript Untuk Pemula*. Jakarta: Duniaikom, 2017.
- [8] Rosa, A. S, *Analisis dan Desain Perangkat Lunak: Rekayasa Perangkat Lunak Untuk Pemrograman Terstruktur, Berorientasi Objek, dan Agile*, Bandung: Informatika, 2022.
- [9] Rosa, A. S. and Shalahudin, M, *Rekayasa Perangkat Lunak (Terstruktur dan Berorientasi Objek)*. Bandung: Informatika, 2018.
- [10] Suryantara, I. G, *Merancang Aplikasi dengan Metodologi Extreme Programming*. Jakarta: PT. Elex Media Komputindo, 2017.
- [11] Tresnawati, D. and Milah, A. S, "Pengembangan Aplikasi Komik Hadis Berbasis Android," *Jurnal Algoritma*, 14(2), 263-271, 2017, doi: 10.33364/algoritma/v.14-2.263