

**Analisis Perbandingan Kinerja Front End Framework pada
Pengembangan Proyek Talent Pool Platform
(Studi Kasus: PT Javan Cipta Solusi)**



Disusun Oleh:

N a m a : Syuhda Fakhrunnisa

NIM : 20523238

**PROGRAM STUDI INFORMATIKA – PROGRAM SARJANA
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS ISLAM INDONESIA**

2024

HALAMAN PENGESAHAN DOSEN PEMBIMBING

**Analisis Perbandingan Kinerja Front End Framework pada
Pengembangan Proyek Talent Pool Platform
(Studi Kasus: PT Javan Cipta Solusi)**

TUGAS AKHIR JALUR MAGANG



الجمعة الإسلامية الأندونيسية

Yogyakarta, 4 November 2024

Pembimbing,

(Rahadian Kurhiawan, S.Kom., M.Kom.)

HALAMAN PENGESAHAN DOSEN PENGUJI

**Analisis Perbandingan Kinerja Front End Framework pada
Pengembangan Proyek Talent Pool Platform
(Studi Kasus: PT Javan Cipta Solusi)**

TUGAS AKHIR JALUR MAGANG

Telah dipertahankan di depan sidang penguji sebagai salah satu syarat untuk memperoleh gelar Sarjana Komputer dari Program Studi Informatika – Program Sarjana di Fakultas Teknologi Industri Universitas Islam Indonesia

Yogyakarta, 4 November 2024

Tim Penguji

Rahadian Kurniawan, S.Kom., M.Kom.

Anggota 1

Chandra Kusuma Dewa, S.Kom., M.Cs., Ph.D.

Anggota 2

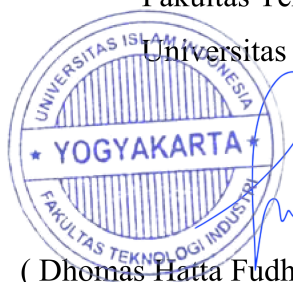
Elyza Gustri Wahyuni, S.T., M.Cs.

Mengetahui,

Ketua Program Studi Informatika – Program Sarjana

Fakultas Teknologi Industri

Universitas Islam Indonesia


(Dhomas Hatta Fudholi, S.T., M.Eng., Ph.D.)

HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR

Yang bertanda tangan di bawah ini:

Nama : Syuhda Fakhrunnisa

NIM : 20523238

Tugas akhir dengan judul:

Analisis Perbandingan Kinerja Front End Framework pada Pengembangan Proyek Talent Pool Platform (Studi Kasus: PT Javan Cipta Solusi)

Menyatakan bahwa seluruh komponen dan isi dalam tugas akhir ini adalah hasil karya saya sendiri. Apabila di kemudian hari terbukti ada beberapa bagian dari karya ini adalah bukan hasil karya sendiri, tugas akhir yang diajukan sebagai hasil karya sendiri ini siap ditarik kembali dan siap menanggung risiko dan konsekuensi apa pun.

Demikian surat pernyataan ini dibuat, semoga dapat dipergunakan sebagaimana mestinya.

Yogyakarta, 29 Oktober 2024



(Syuhda Fakhrunnisa)

HALAMAN PERSEMBAHAN

Dengan penuh rasa syukur kepada Allah Swt., penulis persembahkan tugas akhir ini kepada kedua orang tua, bapak dan ibu yang selalu memberikan dukungan baik moral maupun material, serta doa yang tak pernah putus untuk kesuksesan penulis. Terima kasih atas segala pengorbanan dan kesabaran dalam mendidik hingga saat ini. Kepada ketiga kakak yang telah memberikan semangat dan motivasi selama proses perkuliahan hingga pengerjaan tugas akhir ini, terima kasih telah menjadi panutan dan teladan yang baik dalam keluarga. Ucapan terima kasih juga penulis sampaikan kepada dosen pembimbing yang telah meluangkan waktu dan tenaga untuk membimbing dengan penuh kesabaran dan ketelitian, serta kepada teman-teman seperjuangan yang telah banyak membantu dan memberikan dukungan selama masa perkuliahan. Semoga ilmu yang didapatkan dapat bermanfaat bagi diri sendiri dan orang lain. Akhir kata, penulis persembahkan tugas akhir ini untuk diri sendiri, sebagai bukti bahwa dengan kerja keras dan tekad yang kuat, setiap tujuan dapat tercapai.

HALAMAN MOTO

“Allah tidak membebani seseorang melainkan sesuai dengan kesanggupannya.”

– Q.S Al Baqarah: 286

*“If you want something you've never had. You must be willing to do something
you've never done.”* – Thomas Jefferson

“Gapailah ambisi setinggi langit! Bermimpilah setinggi langit! Ketika kamu jatuh,
kamu akan jatuh di antara bintang-bintang.” – Ir. Soekarno

“Anda mungkin bisa menunda, tapi waktu tidak akan menunggu.” – Benjamin Franklin

KATA PENGANTAR

Assalamu 'alaikum Warahmatullahi Wabarakatuh.

Alhamdulillah, puji dan syukur penulis panjatkan ke hadirat Allah Swt. atas limpahan rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan tugas laporan akhir jalur magang yang berjudul “Analisis Perbandingan Kinerja Front End Framework pada Pengembangan Proyek Talent Pool Platform (Studi Kasus: PT Javan Cipta Solusi)”. Laporan akhir ini disusun sebagai salah satu persyaratan kelulusan dari penjaluran magang di Program Studi Informatika – Program Sarjana, Fakultas Teknologi Industri, Universitas Islam Indonesia. Dalam proses penyusunan laporan akhir ini, penulis telah menerima banyak bantuan, bimbingan, dan dukungan dari berbagai pihak. Oleh karena itu, penulis ingin menyampaikan ucapan terima kasih yang tulus dan sebesar-besarnya kepada:

1. Allah Swt. yang senantiasa memberikan rahmat, pertolongan, dan kekuatan kepada penulis selama masa perkuliahan dan pengerjaan laporan akhir ini.
2. Kedua orang tua tercinta, Bapak Julianto dan Ibu Dr. Salminawati, S.S., M.A., yang tak henti-hentinya memberikan doa, perhatian, dan dukungan baik secara moral maupun materiil.
3. Saudara kandung penulis, Khairun Nadiyah, S.T., M.Eng (Mgt)., Anjah Hayati Fajrin, S.Pd., dan Muhammad Aulia Urrahman, Lc., yang selalu mendoakan dan memberikan motivasi selama masa studi.
4. Prof. Fathul Wahid, S.T., M.Sc., Ph.D., selaku Rektor Universitas Islam Indonesia.
5. Dr. Raden Teduh Dirgahayu, S.T., M.Sc., selaku Ketua Jurusan Informatika dan Bapak DThomas Hatta Fudholi, S.T., M.Eng., Ph.D., selaku Ketua Program Studi Informatika – Program Sarjana.
6. Bapak Rahadian Kurniawan, S.Kom., M.Kom., selaku dosen pembimbing, yang telah meluangkan waktu untuk memberikan arahan dan bimbingan dengan penuh kesabaran dan ketelitian.
7. Seluruh dosen Program Studi Informatika yang sangat berjasa memberikan ilmu yang bermanfaat selama masa perkuliahan.
8. Jajaran manajemen PT Javan Cipta Solusi, khususnya Bapak Wisnu Manupraba dan Ibu Novi Setiani, yang telah memberikan penulis kesempatan magang yang sangat berharga.
9. Bapak Dicky Puja Pratama (*Operational Excellence Lead*), Mas Evanta Yudhistira (*Tribe Lead*), dan Mba Onny Octaviani (Wakil *Tribe Lead*), yang telah

membimbing dan memantau pekerjaan penulis serta memberikan arahan yang bermanfaat selama pelaksanaan magang.

10. Tim teknis, Mas Zukron (*Tech Lead*) dan terutama Mas Imam selaku mentor, yang telah memberikan bantuan dan bimbingan teknis secara detail dan sangat baik, dalam pengerjaan tugas-tugas selama magang.
11. Para *product owner*, terutama Mba Novia Endah Larasati yang telah memberikan dukungan dan perhatian kepada penulis dan tim proyek.
12. Rekan-rekan kerja di PT Javan Cipta Solusi, Aisha Erfiandra, Ervalsa Dwi, Mba Ristri Krisnugraheni, Mba Nanda Nurisya, Mba Nurlaela, serta seluruh karyawan yang telah menciptakan lingkungan kerja yang nyaman dan suportif.
13. Sahabat-sahabat penulis, Fatimah Azzahra Kusuma Dewi, Intan Nabila Azmi, Kartika Salma, Mayla Ayyuni Sonya, Nadira Adiva Wibowo, dan Sonya Ainurohmah, yang senantiasa dengan tulus memberikan dukungan, motivasi, dan menjadi tempat berbagi suka duka.
14. Rekan-rekan *student staff*, terutama Anjeli Puteri Anggreini, yang telah berbagi pengalaman dan memberikan dukungan berarti sepanjang masa perkuliahan.
15. Adik-adik tingkat, khususnya angkatan 2021, yang memberikan warna dan pengalaman berharga selama beraktivitas dan berorganisasi di masa perkuliahan.
16. Semua pihak yang tidak dapat penulis sebutkan satu per satu yang telah memberikan dukungan secara langsung maupun tidak langsung dalam penyelesaian laporan akhir ini.

Penulis berharap semoga Allah Swt. senantiasa melimpahkan rahmat dan membalas segala kebaikan yang telah diberikan oleh semua pihak dengan pahala yang berlipat ganda. Penulis juga menyadari bahwa dalam penyusunan laporan akhir ini masih terdapat banyak kekurangan. Oleh karena itu, penulis sangat mengharapkan kritik dan saran yang membangun untuk penyempurnaan laporan ini. Akhir kata, penulis berharap laporan akhir ini dapat memberikan manfaat bagi para pembaca dan perkembangan ilmu pengetahuan.

Wassalamu'alaikum Warahmatullahi Wabarakatuh.

Yogyakarta, 29 Oktober 2024



(Syuhda Fakhrunnisa)

SARI

Perkembangan teknologi web telah menghadirkan berbagai *framework front-end* dengan karakteristik yang beragam untuk mendukung performa dan pengalaman pengguna yang optimal. Laporan ini menganalisis performa tiga *framework* populer, yaitu ReactJS, VueJS, dan Blade *Templating Engine*, dalam konteks pengembangan platform *talent pool* di PT Javan Cipta Solusi. Pemilihan *framework* ini didasarkan pada pendekatan berbeda dalam *rendering* konten web, di mana ReactJS dan VueJS mewakili *framework* JavaScript modern yang mendukung interaksi dinamis, sedangkan Blade, yang terintegrasi dengan Laravel, berfokus pada *server-side rendering*. Selain analisis performa, laporan ini juga membahas dokumentasi dan peran pengembang dalam proses pengembangan, termasuk pentingnya penyusunan dokumentasi teknis, manajemen proyek, dan koordinasi dalam tim.

Kajian ini bertujuan untuk memberikan pemahaman mendalam mengenai performa dan keandalan masing-masing teknologi dalam menghadirkan aplikasi web yang cepat dan responsif. Pengujian dilakukan menggunakan Google Lighthouse dan Chrome DevTools *Performance tab* untuk mengevaluasi aspek kinerja, aksesibilitas, praktik terbaik, dan optimisasi SEO. Google Lighthouse memberikan gambaran umum terkait performa aplikasi secara keseluruhan, sementara Chrome DevTools *Performance tab* menyediakan data *real-time* pada beberapa metrik kinerja penting (*Core Web Vitals*), seperti *Largest Contentful Paint* (LCP), *Cumulative Layout Shift* (CLS), dan *Interaction to Next Paint* (INP), yang relevan untuk memahami tingkat responsivitas aplikasi web.

Hasil pengujian menunjukkan bahwa setiap *framework* memiliki keunggulan dan kekurangan spesifik. ReactJS menunjukkan keunggulan dalam aspek praktik terbaik dan optimisasi SEO serta stabilitas tata letak, sedangkan Blade lebih unggul dalam waktu pemuatan awal konten, sementara VueJS, dengan integrasi yang fleksibel dalam Laravel, menawarkan fleksibilitas serta performa responsif pada interaksi pengguna, meskipun memerlukan optimalisasi lebih lanjut pada beberapa metrik kinerja spesifik. Rekomendasi pemilihan *framework* ini diharapkan dapat mendukung pengembangan aplikasi web yang sesuai dengan kebutuhan dan karakteristik proyek.

Kata kunci: *web performance*, *framework*, ReactJS, VueJS, Blade *Templating Engine*, Google Lighthouse, Chrome DevTools.

GLOSARIUM

Blade <i>Templating Engine</i>	sistem <i>templating</i> yang digunakan dalam <i>framework</i> Laravel untuk memudahkan pembuatan antarmuka pengguna. Blade mendukung <i>layout</i> dan <i>template</i> yang dapat digunakan kembali, serta dikompilasi menjadi kode PHP biasa yang disimpan dalam <i>cache</i> .
Chrome DevTools <i>Performance Tab</i>	alat analisis kinerja yang ada di Chrome DevTools, membantu mengidentifikasi waktu pemuatan, eksekusi JavaScript, <i>rendering</i> , dan interaksi pengguna untuk memahami efisiensi situs web.
<i>Cumulative Layout Shift (CLS)</i>	metrik yang mengukur kestabilan tata letak halaman saat dimuat, mencerminkan seberapa besar elemen-elemen halaman bergeser hingga mencapai keadaan stabil.
<i>First Contentful Paint (FCP)</i>	waktu yang diperlukan <i>website</i> untuk menampilkan konten pertama seperti gambar atau teks. FCP penting untuk menunjukkan kecepatan <i>website</i> dalam merespons pengguna.
<i>Framework</i>	kerangka kerja yang menyediakan struktur dasar untuk pengembangan aplikasi, membantu dalam menjaga konsistensi dan mempermudah kolaborasi dalam tim pengembangan.
Google Lighthouse	alat <i>open-source</i> untuk mengaudit kualitas halaman web dalam hal kinerja, aksesibilitas, SEO, dan praktik terbaik lainnya. Google Lighthouse memberikan rekomendasi perbaikan berdasarkan standar industri.
<i>Hierarchy</i>	struktur organisasi tradisional yang disusun berdasarkan tingkatan atau jabatan, di mana kewenangan dan tanggung jawab mengalir dari posisi atas ke bawah. Dalam hierarki, setiap level memiliki otoritas yang lebih tinggi atas level di bawahnya, dengan alur komunikasi dan pengambilan keputusan yang terpusat.
<i>Holacracy</i>	struktur manajemen organisasi yang tidak menggunakan hierarki tradisional, melainkan memfokuskan pada <i>self-management</i> dan peran yang dikelompokkan dalam ' <i>circle</i> '. Tiap <i>circle</i> memiliki tanggung jawab dan otonomi untuk mencapai tujuan tertentu.

<i>Interaction to Next Paint</i> (INP)	metrik yang mengukur waktu respons aplikasi dalam menampilkan elemen setelah interaksi pengguna seperti klik atau <i>input</i> . INP penting untuk memahami pengalaman pengguna terkait responsivitas aplikasi.
<i>Largest Contentful Paint</i> (LCP)	metrik yang mengukur waktu yang diperlukan untuk menampilkan konten terbesar pada halaman, menandakan seberapa cepat elemen penting halaman muncul di layar pengguna.
ReactJS	<i>library</i> JavaScript yang dikembangkan oleh Facebook untuk membangun antarmuka pengguna berbasis komponen, memungkinkan pengembangan aplikasi yang efisien dan dapat digunakan kembali.
Scrum	metode manajemen proyek dalam pengembangan perangkat lunak yang bersifat <i>agile</i> , membagi proyek menjadi sprint pendek untuk memaksimalkan efisiensi dan adaptabilitas terhadap perubahan kebutuhan proyek.
<i>Server-Side Rendering</i> (SSR)	proses <i>rendering</i> konten halaman web di server sebelum dikirimkan ke klien, memungkinkan waktu muat awal yang lebih cepat dan optimal untuk SEO.
<i>Single Page Application</i> (SPA)	pendekatan pengembangan aplikasi web yang memungkinkan pengguna berpindah halaman tanpa perlu memuat ulang, membuat aplikasi lebih cepat dan interaktif.
<i>Speed Index</i> (SI)	metrik yang menunjukkan seberapa cepat konten utama dari sebuah halaman muncul secara visual selama pemuatan halaman. SI memberikan gambaran kecepatan aplikasi dari perspektif pengguna.
<i>Total Blocking Time</i> (TBT)	total waktu selama proses pemuatan halaman di mana respons terhadap <i>input</i> pengguna diblokir. TBT menunjukkan tingkat interaktivitas situs selama <i>loading</i> .
VueJS	<i>framework</i> JavaScript yang digunakan untuk membangun aplikasi web yang interaktif. VueJS menawarkan pendekatan <i>Single Page Application</i> (SPA) dengan konsep reaktif, memiliki fitur data <i>binding</i> , <i>directives</i> , dan komponen yang dapat digunakan kembali.

DAFTAR ISI

HALAMAN JUDUL	i
HALAMAN PENGESAHAN DOSEN PEMBIMBING.....	ii
HALAMAN PENGESAHAN DOSEN PENGUJI.....	iii
HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR	iv
HALAMAN PERSEMBAHAN	v
HALAMAN MOTO	vi
KATA PENGANTAR	vii
SARI	ix
GLOSARIUM.....	x
DAFTAR ISI.....	xii
DAFTAR TABEL.....	xv
DAFTAR GAMBAR	xvi
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Ruang Lingkup	4
1.3 Tujuan.....	6
1.4 Manfaat.....	6
1.5 Sistematika Penulisan.....	6
BAB II LANDASAN TEORI DAN TINJAUAN PUSTAKA	8
2.1 <i>Website</i>	8
2.2 <i>Front-end</i>	8
2.3 <i>Framework</i>	8
2.3.1 <i>Framework</i> ReactJS.....	9
2.3.2 <i>Framework</i> VueJS	9
2.3.3 <i>Framework</i> Laravel dan Blade Templating Engine.....	10
2.4 Metode <i>Scrum</i>	11
2.5 Lighthouse	12
2.6 Chrome DevTools Performance tab	13
2.7 Tinjauan Pustaka	15
BAB III PELAKSANAAN MAGANG.....	19
3.1 Aktivitas Magang	19
3.1.1 <i>Onboarding</i>	20

3.1.2	<i>Governance Meeting</i>	25
3.1.3	Pelatihan BPMN	26
3.1.4	Membantu Mengembangkan Proyek A	29
3.1.5	Membantu Mengembangkan Proyek B	29
3.1.6	Membantu Mengembangkan Proyek C	30
3.1.7	<i>Tactical meeting</i>	30
3.1.8	Mengerjakan Dokumen <i>Source Code</i> Proyek A, C, D, E, dan F.....	31
3.1.9	Membantu Mengembangkan Proyek G	33
3.1.10	Membantu Mengembangkan Proyek H.....	33
3.2	Manajemen Proyek.....	33
3.3	Pengembangan Proyek A	37
3.3.1	Pendefinisian Proyek	37
3.3.2	Inisialisasi Proyek.....	37
3.3.3	Perencanaan dan Pelaksanaan Proyek	40
3.3.4	Pemantauan dan Pengendalian Proyek	59
3.3.5	Penutupan Proyek	60
3.4	Pengembangan Proyek B.....	60
3.4.1	Pendefinisian Proyek	60
3.4.2	Inisialisasi Proyek.....	61
3.4.3	Perencanaan dan Pelaksanaan Proyek	63
3.4.4	Pemantauan dan Pengendalian Proyek	97
3.4.5	Penutupan Proyek	98
3.5	Pembuatan Dokumen <i>Source Code</i>	98
BAB IV REFLEKSI PELAKSANAAN MAGANG		100
4.1	Evaluasi Proses Magang.....	100
4.1.1	Pemilihan Teknologi pada Setiap Proyek.....	100
4.1.2	Dokumentasi proyek.....	121
4.1.3	Metode <i>Scrum</i>	123
4.1.4	Tim Pengembangan Sistem	124
4.1.5	Evaluasi Supervisor	125
4.2	Manfaat Magang Bagi Mahasiswa	128
4.2.1	Magang di Perusahaan <i>Software House</i>	128
4.2.2	Pengenalan Lingkungan Perusahaan	129
4.2.3	Mempelajari dasar-dasar BPMN	130

4.2.4 Mengembangkan Proyek	130
4.2.5 <i>Meetings</i>	131
4.2.6 Hari Berkualitas	132
BAB V PENUTUP	134
5.1 Kesimpulan.....	134
5.2 Saran	134
DAFTAR PUSTAKA	136
LAMPIRAN.....	141

DAFTAR TABEL

Tabel 2.1 Tinjauan Pustaka Penelitian Terdahulu	15
Tabel 3.1 Aktivitas Magang.....	19
Tabel 3.2 Aktivitas penyusunan dokumen <i>source code</i>	99
Tabel 4.1 Kriteria metrik kinerja pada Lighthouse.....	114
Tabel 4.2 Hasil Pengujian Keseluruhan Metrik Lighthouse.....	115
Tabel 4.3 Hasil Pengujian Metrik Kinerja pada Lighthouse.....	115
Tabel 4.4 Hasil Pengujian Chrome DevTools <i>Performance tab</i>	115
Tabel 4.5 Hasil Evaluasi Magang oleh <i>Operational Lead</i>	126
Tabel 4.6 Hasil Evaluasi Magang oleh <i>Tribe Lead</i>	127

DAFTAR GAMBAR

Gambar 1.1 Kantor J100 PT Javan Cipta Solusi.....	2
Gambar 1.2 Kantor B23 PT Javan Cipta Solusi	2
Gambar 1.3 Lokasi Kantor PT Javan Cipta Solusi	2
Gambar 2.1 Tahapan aktivitas dalam metode <i>Scrum</i>	11
Gambar 2.2 Contoh skor sempurna Google Lighthouse terhadap sebuah <i>website</i>	12
Gambar 2.3 Metrik Chrome DevTools <i>Performance Tab</i> dengan kode warna berdasarkan ambang performa	14
Gambar 3.1 <i>Timeline</i> aktivitas magang	20
Gambar 3.2 <i>Onboarding</i> magang	21
Gambar 3.3 Pelatihan yang diadakan pada hari berkualitas untuk para <i>programmers</i>	21
Gambar 3.4 Kegiatan <i>town hall</i> Januari 2024.....	21
Gambar 3.5 <i>Task onboarding</i> magang.....	22
Gambar 3.6 Perkenalan anggota tim baru di PT Javan Cipta Solusi	23
Gambar 3.7 Suasana kerja di PT Javan Cipta Solusi	24
Gambar 3.8 Kunjungan dosen Informatika UII ke PT Javan Cipta Solusi.....	24
Gambar 3.9 Struktur organisasi PT Javan Cipta Solusi.....	25
Gambar 3.10 <i>Governance meeting</i>	26
Gambar 3.11 <i>Task</i> pelatihan BPMN	27
Gambar 3.12 Detail pengerjaan <i>task</i> pelatihan BPMN.....	28
Gambar 3.13 <i>Feedback</i> penilaian hasil pelatihan BPMN di ActiveCollab	28
Gambar 3.14 Integrasi antar modul	30
Gambar 3.15 <i>Tactical meeting tribe lead</i> Evan (13 Oktober 2023)	31
Gambar 3.16 <i>Tactical meeting tribe lead</i> Evan (25 Oktober 2023)	31
Gambar 3.17 Contoh <i>template</i> dokumen <i>source code</i>	32
Gambar 3.18 Tahapan aktivitas dalam <i>scrum framework</i>	34
Gambar 3.19 <i>Daily scrum</i>	35
Gambar 3.20 <i>Tools</i> dan teknologi yang digunakan di PT Javan Cipta Solusi.....	35
Gambar 3.21 <i>Dashboard</i> Taiga.....	36
Gambar 3.22 <i>Task life cycle</i>	36
Gambar 3.23 <i>Roles</i> yang terlibat dalam proyek A.....	37
Gambar 3.24 Diagram arsitektur sistem Proyek A	39
Gambar 3.25 <i>Task</i> dan detail instruksi <i>setup</i> proyek A di Taiga.....	40

Gambar 3.26 Dokumentasi hasil <i>setup</i> proyek A di <i>local environment</i>	41
Gambar 3.27 Detail <i>task</i> implementasi validasi data duplikat di Taiga	41
Gambar 3.28 Hasil implementasi validasi data duplikat pada <i>form</i>	42
Gambar 3.29 Detail <i>task</i> implementasi validasi batasan karakter di Taiga	43
Gambar 3.30 Hasil implementasi pesan validasi untuk pembatasan karakter pada formulir ..	44
Gambar 3.31 Detail <i>task</i> implementasi penghitung karakter di Taiga.....	45
Gambar 3.32 Hasil implementasi penghitung karakter pada kolom deskripsi pekerjaan.....	46
Gambar 3.33 Detail <i>task</i> integrasi fitur validasi dan penghitung karakter pada formulir.....	47
Gambar 3.34 Hasil implementasi sistem validasi dan penghitung karakter pada formulir	48
Gambar 3.35 Detail <i>task</i> pengurutan data terurut <i>default</i> dari yang terbaru di Taiga	49
Gambar 3.36 Hasil implementasi pengurutan data terurut <i>default</i> dari yang terbaru.....	50
Gambar 3.37 <i>Task</i> penyesuaian tampilan <i>disable</i> tombol <i>edit</i>	51
Gambar 3.38 Hasil implementasi penyesuaian tampilan <i>disable</i> tombol <i>edit</i>	52
Gambar 3.39 <i>Task</i> perbaikan <i>wording</i> status.....	53
Gambar 3.40 Hasil implementasi perbaikan <i>wording</i> status	54
Gambar 3.41 <i>Task</i> redesain ukuran <i>font</i> deskripsi dan modal konfirmasi.....	55
Gambar 3.42 Hasil implementasi redesain ukuran <i>font</i> deskripsi dan modal konfirmasi	56
Gambar 3.43 <i>Task</i> perubahan ikon dan tata letak <i>list</i> soal.....	57
Gambar 3.44 Hasil implementasi perubahan ikon dan tata letak <i>list</i> soal	58
Gambar 3.45 Diagram arsitektur sistem Proyek B	63
Gambar 3.46 Detail <i>task setup</i> proyek B di Taiga.....	64
Gambar 3.47 <i>Task</i> penyesuaian tampilan <i>header</i> tabel	65
Gambar 3.48 Hasil implementasi penyesuaian tampilan <i>header</i> tabel	66
Gambar 3.49 <i>Task</i> perbaikan indikator <i>loading</i> pada tombol finalisasi	67
Gambar 3.50 Hasil implementasi perbaikan indikator <i>loading</i> pada tombol finalisasi.....	68
Gambar 3.51 <i>Task</i> penambahan <i>default value</i> pada kolom deskripsi dan rincian.....	69
Gambar 3.52 Hasil implementasi penambahan <i>default value</i> kolom deskripsi dan rincian	70
Gambar 3.53 <i>Task</i> penyesuaian tampilan <i>field</i> ‘Nama Supplier’	71
Gambar 3.54 Hasil implementasi penyesuaian tampilan <i>field</i> ‘Nama Supplier’	72
Gambar 3.55 <i>Task</i> penyesuaian tombol pada daftar persetujuan.....	73
Gambar 3.56 Hasil implementasi penyesuaian tombol pada daftar persetujuan	74
Gambar 3.57 <i>Task</i> perbaikan <i>field</i> tanggal pada <i>form</i>	75
Gambar 3.58 Hasil implementasi perbaikan <i>field</i> tanggal pada <i>form</i>	76
Gambar 3.59 <i>Task</i> penambahan tombol cetak pada kolom aksi tabel	77

Gambar 3.60 Hasil implementasi penambahan tombol cetak pada kolom aksi tabel.....	78
Gambar 3.61 <i>Task</i> validasi <i>input</i> angka pada <i>form</i>	79
Gambar 3.62 Hasil implementasi validasi <i>input</i> angka pada <i>form</i>	80
Gambar 3.63 <i>Task</i> validasi kelengkapan <i>form</i> nota dinas.....	81
Gambar 3.64 Hasil implementasi validasi kelengkapan <i>form</i> nota dinas	82
Gambar 3.65 <i>Task</i> validasi kolom <i>mandatory</i> pada <i>form</i> honor narasumber	83
Gambar 3.66 Hasil implementasi validasi kolom <i>mandatory</i> pada <i>form</i> honor narasumber...	84
Gambar 3.67 <i>Task</i> implementasi <i>searching</i> data instansi.....	85
Gambar 3.68 Hasil implementasi <i>searching</i> data instansi	86
Gambar 3.69 <i>Task</i> <i>sorting</i> data Dokumen PBJ.....	87
Gambar 3.70 Hasil implementasi <i>sorting</i> data Dokumen PBJ	88
Gambar 3.71 <i>Task</i> implementasi <i>filtering</i> berdasarkan proses dokumen	89
Gambar 3.72 Hasil implementasi <i>filtering</i> berdasarkan proses dokumen	90
Gambar 3.73 <i>Task</i> implementasi <i>filtering</i> data pegawai.....	91
Gambar 3.74 Hasil implementasi <i>filtering</i> data pegawai.....	92
Gambar 3.75 <i>Task</i> <i>save draft</i> finalisasi Surat Perjalanan Dinas	93
Gambar 3.76 Hasil implementasi <i>save draft</i> finalisasi Surat Perjalanan Dinas.....	94
Gambar 3.77 <i>Task</i> <i>save draft</i> dan <i>redirect</i> halaman	95
Gambar 3.78 Hasil implementasi <i>save draft</i> dan <i>redirect</i> halaman.....	96
Gambar 3.79 Contoh dokumen <i>source code</i>	99
Gambar 4.1 Bobot metrik kinerja spesifik Lighthouse versi 10	101
Gambar 4.2 Lighthouse <i>scoring calculator</i>	104
Gambar 4.3 Bobot untuk setiap audit aksesibilitas Lighthouse (1)	105
Gambar 4.4 Bobot untuk setiap audit aksesibilitas Lighthouse (2)	106
Gambar 4.5 Bobot untuk setiap audit aksesibilitas Lighthouse (3)	107
Gambar 4.6 Bobot untuk setiap audit <i>best practices</i> Lighthouse	108
Gambar 4.7 Bobot untuk setiap audit SEO Lighthouse	109
Gambar 4.8 Hasil pengujian empat metrik Lighthouse dan lima metrik kinerja (<i>performance</i>) pada proyek B (ReactJS).....	109
Gambar 4.9 Hasil pengujian empat metrik Lighthouse dan lima metrik kinerja (<i>performance</i>) pada proyek G (VueJS).....	110
Gambar 4.10 Hasil pengujian empat metrik Lighthouse dan lima metrik kinerja (<i>performance</i>) pada proyek A (Blade)	111

Gambar 4.11 Hasil pengujian empat metrik Lighthouse dan lima metrik kinerja (<i>performance</i>) pada proyek G (VueJS).....	111
Gambar 4.12 Hasil pengujian kinerja (<i>performance</i>) proyek A (Blade), B (ReactJS), dan G (VueJS) menggunakan Chrome DevTools <i>Performance tab</i>	113
Gambar 4.13 Tahapan aktivitas dalam <i>scrum framework</i>	123
Gambar 4.14 Kegiatan-kegiatan yang diadakan pada ‘Hari Berkualitas’	132
Gambar 4.15 Pelatihan BPMN yang diadakan pada ‘Hari Berkualitas’	133

BAB I

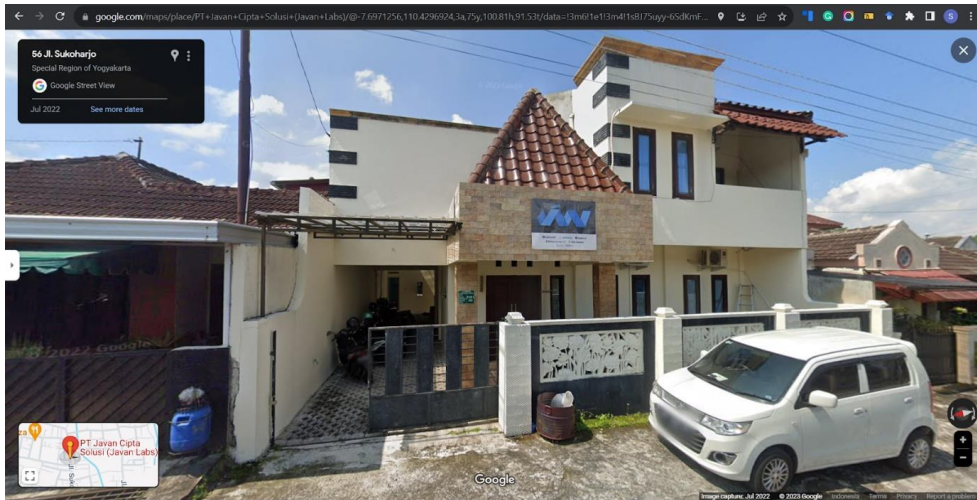
PENDAHULUAN

1.1 Latar Belakang

Kemajuan pesat teknologi web di era digital menghadirkan berbagai *framework* dan *library* yang semakin memudahkan pengembangan aplikasi. Di antara banyaknya pilihan yang tersedia, ReactJS, VueJS, dan Blade *Templating Engine*—yang digunakan dalam *framework* Laravel—telah menjadi tiga alat yang paling populer dalam membangun aplikasi web. Ketiga teknologi ini menawarkan pendekatan yang berbeda dalam pengembangan *front-end*. ReactJS dan VueJS, dengan pendekatannya sebagai *Single Page Application* (SPA), memungkinkan pengembangan aplikasi yang sangat interaktif dan responsif di sisi klien (CoreUI, 2022). Sementara itu, Blade dengan pendekatan *Server-Side Rendering* (SSR) menawarkan keunggulan dalam hal *rendering* awal yang cepat dan dukungan optimal terhadap SEO (Erolin, 2023). Meskipun ketiga teknologi ini memiliki tujuan yang sama, yaitu memfasilitasi pengembangan *front-end* yang baik, mereka memiliki karakteristik yang berbeda. ReactJS menggunakan konsep komponen dan virtual DOM (CoreUI, 2022; Long, 2023), VueJS menggabungkan reaktivitas dengan sistem *template* yang intuitif (Kurniawan et al., 2023), sementara Blade menggunakan sistem *template* yang lebih konvensional. Ketiga teknologi ini pun populer di kalangan *software house*, termasuk PT Javan Cipta Solusi.

PT Javan Cipta Solusi merupakan perusahaan *software house* yang bergerak dalam pengembangan berbagai aplikasi web dan perangkat lunak. Sebagai perusahaan yang bergerak di bidang teknologi dan pengembangan perangkat lunak, PT Javan Cipta Solusi berkomitmen untuk membantu mitra serta klien dalam mengoptimalkan proses bisnis melalui solusi berbasis teknologi. Didirikan pada Desember 2008, perusahaan ini secara resmi beroperasi dengan nama legal CV Javan IT Services pada bulan Februari 2009, dan pada tahun 2012, badan hukum perusahaan berubah menjadi PT Javan Cipta Solusi. Pada awal pendiriannya, PT Javan Cipta Solusi diinisiasi oleh tiga individu, yaitu Wisnu Manupraba, Indra Sakti Wijayanto, dan Bayu Hendra Winata. Saat ini, perusahaan memiliki kantor pusat di Bandung, Jawa Barat, serta cabang di Yogyakarta. Kantor utama di Yogyakarta berlokasi di Perum Sukoharjo Indah J100, Sleman, Yogyakarta, dengan tampak depan bangunan kantor utama dapat dilihat pada Gambar 1.1 dan tampilan dari Google Maps pada Gambar 1.3. Selain di Perumahan J100, perusahaan ini juga memiliki kantor tambahan yang terletak tidak jauh dari kantor utama, sebagaimana ditunjukkan pada Gambar 1.2. Kantor utama biasanya menampung tim operasional, sedangkan

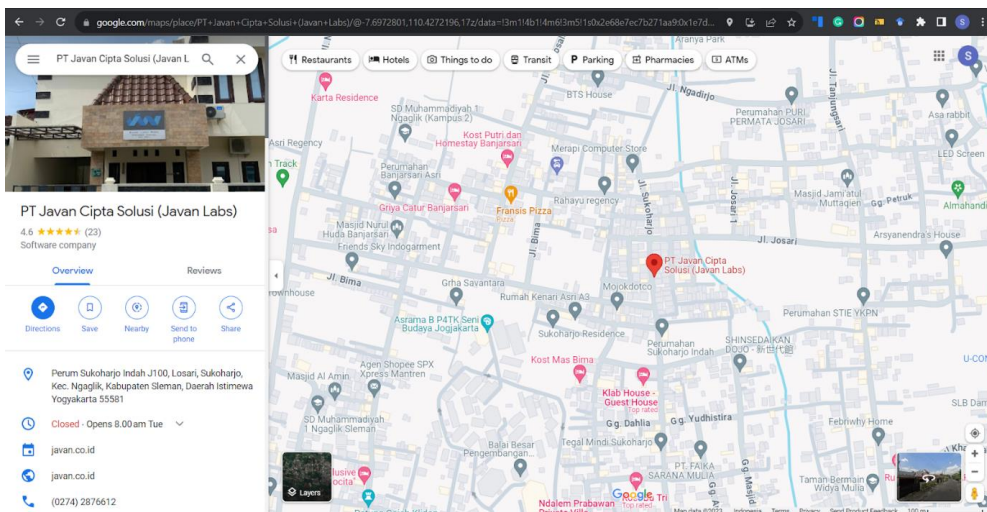
kantor tambahan digunakan oleh tim lainnya, seperti *quality assurance*, *finance*, *people growth*, dan berbagai tim lainnya.



Gambar 1.1 Kantor J100 PT Javan Cipta Solusi



Gambar 1.2 Kantor B23 PT Javan Cipta Solusi



Gambar 1.3 Lokasi Kantor PT Javan Cipta Solusi

Sebagai perusahaan yang terus berkembang, PT Javan Cipta Solusi menangani berbagai proyek untuk klien dari sektor swasta, lembaga pemerintah, hingga organisasi non-profit. Dengan pengalaman yang luas dalam bidang teknologi, perusahaan ini mengerjakan proyek-proyek yang mencakup pengembangan aplikasi web, *mobile*, sistem *enterprise*, hingga analitik data. Selama menjalani program magang di perusahaan tersebut, penulis berperan sebagai *Front-end (React) Programmer* dan terlibat dalam delapan proyek yang menggunakan berbagai teknologi *front-end*. Kontribusi yang diberikan penulis bervariasi, mencakup pengembangan, penyelesaian masalah kode, serta pembuatan dokumentasi *source code* untuk kebutuhan teknis.

Proyek-proyek yang dikerjakan oleh PT Javan Cipta Solusi tersebut mencakup **Proyek A** yaitu **Sistem Informasi Jobseeker**, **Proyek B** yang merupakan **Modul Administrasi Keuangan**, **Proyek C** sebagai **Layanan API dan Widget**, **Proyek D** untuk **Aplikasi Presensi Peserta Ujian**, **Proyek E** sebagai **Showcase Portfolio Peserta**, **Proyek F** untuk **Web Crawling Data Sosial Media**, **Proyek G** sebagai **Platform Ujian – Pelatihan Coding**, dan **Proyek H** yang merupakan **Sistem Informasi Pengendalian Kasus**. Dalam proyek A dan C, penulis berperan aktif dalam pengembangan kode sekaligus penyusunan dokumentasi *source code*. Sementara itu, pada proyek B, G, dan H, penulis lebih banyak berfokus pada pengembangan dan penyelesaian masalah kode tanpa terlibat dalam proses dokumentasi. Adapun dalam tiga proyek lainnya, yaitu proyek D, E, dan F, peran penulis lebih difokuskan pada pembuatan dokumentasi *source code* sebagai bagian dari kebutuhan teknis proyek.

Dari delapan proyek tersebut, penulis akan berfokus pada *Talent Pool Platform*, sebuah ekosistem terintegrasi yang dirancang untuk menghubungkan talenta digital Indonesia dengan berbagai peluang karier di industri teknologi. *Platform* ini mendukung program pengembangan talenta dengan tujuan meningkatkan keterampilan dan daya saing SDM bidang teknologi informasi dan komunikasi. Dalam pengembangannya, *platform* ini terdiri dari beberapa modul yang masing-masing dibangun menggunakan teknologi *front-end* yang berbeda. Tiga modul utama yang menjadi fokus pembahasan adalah Proyek A, B, dan G yang menggunakan teknologi *front-end* berbeda namun terintegrasi dalam satu ekosistem. Proyek A merupakan Sistem Informasi *Jobseeker* yang dikembangkan menggunakan Blade sebagai *templating engine* yang terintegrasi dengan Laravel. Sistem ini dirancang untuk menjembatani kesenjangan antara pencari kerja muda Indonesia dan perusahaan yang membutuhkan tenaga kerja potensial. Sistem ini mengintegrasikan profil dan portofolio pengguna, memungkinkan

masyarakat muda menonjolkan keterampilan mereka, serta mempermudah perusahaan dalam proses rekrutmen yang lebih selektif. Proyek G adalah Platform Ujian – Pelatihan Coding yang dikembangkan dengan VueJS dan Laravel sebagai *back-end*, memberikan fasilitas tambahan dalam pelaksanaan pelatihan *coding* dan menyediakan media praktik *coding* dasar tanpa memerlukan instalasi di perangkat pengguna. Sementara itu, Proyek B yang menggunakan kombinasi ReactJS dan Laravel, berfokus pada pengelolaan administrasi keuangan termasuk data kegiatan, perjalanan dinas, dan kwitansi.

Ketiga modul yang dibangun dengan arsitektur *microservice* ini memberikan penulis pengalaman langsung dalam mengembangkan aplikasi menggunakan teknologi *front-end* yang berbeda. Pengalaman ini membuka wawasan tentang karakteristik, tantangan, dan kinerja masing-masing *framework* dalam konteks pengembangan yang nyata dan terintegrasi. Dalam era pengembangan aplikasi web yang semakin kompleks, kinerja menjadi faktor penting yang harus dipertimbangkan. Pengembang sering kali dihadapkan pada dilema dalam memilih teknologi yang tepat, terutama dalam proyek-proyek yang membutuhkan kinerja tinggi dan pengalaman pengguna yang optimal. Melalui analisis perbandingan ini, diharapkan hasilnya dapat memberikan gambaran komprehensif tentang kinerja ketiga teknologi *front-end* tersebut, khususnya dalam konteks pengembangan *platform talent pool*. Hasil analisis ini diharapkan dapat membantu organisasi dan pengembang dalam membuat keputusan yang lebih informatif mengenai pemilihan teknologi *front-end* serta mendorong inovasi lebih lanjut dalam pengembangan aplikasi web yang berkinerja tinggi. Selain itu, penulis juga akan mengkaji berbagai aspek manajemen proyek di PT Javan Cipta Solusi, mencakup implementasi sprint dalam metode *Scrum*, dokumentasi proyek, serta distribusi peran dan tanggung jawab dalam tim pengembangan, yang berperan penting dalam keberhasilan pengembangan proyek.

1.2 Ruang Lingkup

Pelaksanaan magang sebagai *frontend programmer* di PT Javan Cipta Solusi berlangsung selama kurang lebih enam bulan, mulai dari September 2023 hingga April 2024. Selama periode magang, penulis berkontribusi dalam pengembangan delapan proyek berbasis web, di mana tiga proyek di antaranya melibatkan penulis secara khusus dalam pembuatan dokumentasi *source code* untuk kebutuhan teknis. Proyek-proyek tersebut dikembangkan dengan metode *Scrum* yang melibatkan berbagai peran, termasuk *product owner*, *system analyst*, *back-end programmer*, dan *front-end programmer*. Setiap proyek mengimplementasikan beragam teknologi pada sisi *back-end* dan *front-end*, serta didukung

dengan dokumentasi yang komprehensif untuk memudahkan pengembangan dan pemeliharaan sistem. Dari delapan proyek yang dikerjakan, tiga proyek yang menjadi fokus utama analisis adalah sebagai berikut:

- a. Proyek A (Sistem Informasi *Jobseeker*), menggunakan Blade sebagai *templating engine* yang terintegrasi dengan Laravel.
- b. Proyek B (Modul Administrasi Keuangan), menggunakan ReactJS sebagai teknologi utama dalam pengembangan *front-end*, dengan Laravel berfungsi sebagai *back-end*.
- c. Proyek G (Platform Ujian – Pelatihan *Coding*), menggunakan VueJS dalam pengembangan *front-end*, dengan Laravel berfungsi sebagai *back-end*.

Pemilihan ketiga proyek tersebut dari total delapan proyek didasarkan pada keterlibatan penulis dalam pengembangannya. Dari delapan proyek, lima proyek melibatkan penulis dalam pengembangan langsung, sementara tiga proyek lainnya berfokus pada pembuatan dokumentasi *source code* untuk kebutuhan teknis. Dari lima proyek yang dikembangkan, tiga proyek menggunakan ReactJS sebagai teknologi utama dalam pengembangan *front-end*, sesuai dengan peran penulis sebagai *front-end (React) programmer*. Sementara itu, Proyek A yang menggunakan Blade dan Proyek G yang menggunakan VueJS menjadi pilihan analisis karena penggunaan *framework* yang berbeda. Pemilihan ketiga proyek ini juga didasari oleh kesamaan *back-end* yang menggunakan Laravel dan keterkaitan dalam satu ekosistem *talent pool platform* yang terintegrasi, memungkinkan perbandingan yang lebih objektif dalam hal kinerja dan implementasi.

Ketiga proyek terpilih merupakan bagian dari ekosistem *talent pool platform* yang terintegrasi, namun dikembangkan dengan pendekatan *microservice* menggunakan teknologi *front-end* yang berbeda. Dalam konteks pengembangan aplikasi web modern, pemilihan teknologi yang sesuai menjadi sangat penting untuk memastikan kinerja yang optimal. Setiap teknologi *framework* memiliki karakteristik, keunggulan, dan tantangan tersendiri yang dapat mempengaruhi hasil akhir proyek. PT Javan Cipta Solusi, sebagai perusahaan yang menerapkan pendekatan selektif dalam pemilihan teknologi, pasti membutuhkan pemahaman mendalam tentang kinerja masing-masing *framework* untuk optimasi pengembangan di masa depan. Oleh karena itu, penulis merasa perlu melakukan analisis perbandingan kinerja antara Proyek A (Blade *Templating Engine*), Proyek B (ReactJS), dan Proyek G (VueJS) dalam konteks pengembangan *talent pool platform* yang terintegrasi, guna memahami keunggulan teknis masing-masing teknologi dalam mendukung pengembangan aplikasi web yang optimal.

1.3 Tujuan

Tujuan dari analisis perbandingan kinerja *front-end framework* pada pengembangan proyek *talent pool platform* adalah sebagai berikut:

- a. Menganalisis perbedaan kinerja antara ReactJS, VueJS, dan Blade *Templating Engine* dalam pengembangan *front-end* aplikasi web.
- b. Memberikan rekomendasi pemilihan teknologi *front-end* yang sesuai berdasarkan karakteristik dan kebutuhan proyek tertentu.

1.4 Manfaat

Manfaat dari analisis perbandingan kinerja *front-end framework* pada pengembangan proyek *talent pool platform* adalah sebagai berikut:

- a. Menyediakan wawasan mendalam tentang karakteristik, kelebihan, dan tantangan ReactJS, VueJS, dan Blade *Templating Engine*, yang dapat membantu dalam pengambilan keputusan teknologi *front-end* berdasarkan kebutuhan spesifik dan karakteristik masing-masing.
- b. Menyediakan analisis komprehensif yang dapat digunakan sebagai referensi dalam mengevaluasi teknologi *front-end* untuk proyek-proyek di masa depan.

1.5 Sistematika Penulisan

Sistematika penulisan pada laporan ini menjelaskan mengenai gambaran umum yang terdapat pada laporan. Adapun susunan sistematika yang digunakan sebagai berikut:

- a. BAB I: Pendahuluan
Bab ini membahas mengenai latar belakang, ruang lingkup magang, tujuan dan manfaat, serta sistematika penulisan.
- b. BAB II: Landasan Teori dan Tinjauan Pustaka
Bab ini membahas mengenai teori-teori yang mendukung topik laporan akhir, berupa penjelasan mengenai *website*, *front-end*, *framework* ReactJS, *framework* VueJS, *framework* Laravel dan Blade *Templating Engine*, metode pengembangan *Scrum*, Lighthouse dan Chrome DevTools *Performance Tab* sebagai alat pengujian, serta tinjauan pustaka.

c. BAB III: Pelaksanaan Magang

Bab ini membahas mengenai pelaksanaan magang pada proyek-proyek yang menjadi topik laporan akhir, mencakup dokumentasi dari kegiatan dan tugas yang dilaksanakan selama menjalani proses magang.

d. BAB IV: Refleksi Pelaksanaan Magang

Bab ini membahas mengenai pembelajaran yang didapatkan selama menjalani proses magang.

e. BAB V: Penutup

Bab ini membahas mengenai kesimpulan yang didapatkan dari pembahasan yang telah dilakukan pada bab-bab sebelumnya dan saran yang dapat diberikan untuk perbaikan ke depannya.

BAB II

LANDASAN TEORI DAN TINJAUAN PUSTAKA

2.1 *Website*

Website adalah sekumpulan halaman web yang di-*hosting* di-*server* dan dapat diakses melalui internet menggunakan *browser*. *Website* berada dalam domain atau subdomain yang sering disebut dengan WWW atau *World Wide Web* (Endra et al., 2021). *Website* berfungsi sebagai *platform* untuk menyampaikan informasi, berinteraksi dengan pengguna, dan menyediakan layanan (Sulistiyanto et al., 2024). *Website* dapat dibedakan menjadi beberapa jenis, termasuk *website* statis dan dinamis, tergantung pada cara konten dikelola dan ditampilkan kepada pengguna. *Website* statis biasanya tidak berubah setelah dibuat, sedangkan *website* dinamis dapat memperbarui kontennya secara otomatis berdasarkan data dari server atau interaksi pengguna (Sa'ad, 2020).

Pengembangan *website* melibatkan berbagai teknologi dan alat yang memungkinkan pengembang untuk menciptakan pengalaman pengguna yang menarik. Dalam konteks bisnis, *website* berfungsi sebagai alat pemasaran yang kuat, memungkinkan perusahaan untuk menjangkau audiens yang lebih luas dan meningkatkan visibilitas merek mereka. *Website* yang dirancang dengan baik dapat meningkatkan kepercayaan pengguna dan mendorong konversi (Adhitya et al., 2024), sehingga penting bagi perusahaan untuk menginvestasikan waktu dan sumber daya dalam pengembangan *website* yang efektif.

2.2 *Front-end*

Front-end adalah bagian dari aplikasi web yang berinteraksi langsung dengan pengguna akhir. Ini mencakup semua elemen visual yang terlihat di layar pengguna, seperti *layout*, desain grafis, dan navigasi. Pengembangan *front-end* melibatkan penggunaan HTML untuk struktur konten, CSS untuk *styling* visual, dan JavaScript untuk interaktivitas dinamis pada halaman web (Pardosi et al., 2024). Pengembang *front-end* bertanggung jawab untuk memastikan bahwa antarmuka pengguna tidak hanya menarik tetapi juga responsif terhadap berbagai perangkat dan ukuran layar.

2.3 *Framework*

Framework adalah kerangka kerja yang menyediakan struktur dasar untuk pengembangan aplikasi web (Pardosi et al., 2024). Sebuah kerangka kerja pemrograman dapat

menyederhanakan proses menyusun kode fungsi program dengan mengurangi kode operasi-operasi yang bersifat *repetitive*. Para pengembang menggunakan *framework* untuk memudahkan pengembangan perangkat lunak. *Framework* secara signifikan mengurangi waktu, sumber daya, dan usaha yang dibutuhkan untuk mengembangkan dan mengelola aplikasi web (Pasaribu, 2017). *Framework* juga membantu dalam menjaga konsistensi kode dan memfasilitasi kolaborasi antar anggota tim, karena semua orang bekerja dengan pendekatan dan alat yang sama. Selain itu, banyak *framework* dilengkapi dengan komunitas aktif yang menyediakan dukungan dan pembaruan berkala, sehingga pengembang dapat tetap mengikuti perkembangan teknologi terbaru (Anggraeni et al., 2024). Penggunaan *framework* yang tepat dapat sangat mempengaruhi kualitas akhir dari produk perangkat lunak serta kecepatan dalam pengembangan.

2.3.1 *Framework ReactJS*

ReactJS adalah *library* JavaScript yang dikembangkan oleh Facebook untuk membangun antarmuka pengguna yang dinamis dan interaktif dengan pendekatan berbasis komponen (*component-based architecture*). Dengan menggunakan ReactJS, pengembang dapat membangun aplikasi web yang kompleks dengan lebih efisien karena komponen dapat digunakan kembali di berbagai bagian aplikasi tanpa perlu menulis kode baru (Wali & Ahmad, 2018). Selain itu, ReactJS menggunakan Virtual DOM untuk meminimalkan pembaruan langsung ke DOM nyata, sehingga meningkatkan performa aplikasi secara keseluruhan (Uriawan et al., 2024). Keunggulan lain dari ReactJS adalah kemampuannya dalam mengelola *state* aplikasi secara efisien melalui fitur-fitur unggulan seperti Hooks, Redux, dan Context API (Yusran, 2023).

2.3.2 *Framework VueJS*

VueJS adalah *framework* JavaScript yang digunakan untuk membangun aplikasi web atau tampilan antarmuka situs web agar lebih interaktif. *Framework* ini cocok digunakan dalam pengembangan aplikasi berbasis antarmuka pengguna, baik untuk halaman web, aplikasi *mobile*, maupun aplikasi desktop (Rosliani et al., 2023). *Framework* ini mengadopsi konsep *Single Page Application* (SPA), yang memungkinkan pengguna berpindah halaman tanpa perlu memuat ulang, sehingga meningkatkan pengalaman pengguna (Antoro et al., 2022). VueJS dikembangkan pada tahun 2013 oleh Evan You, yang juga berperan dalam pengembangan Angular (Prasetyo et al., 2022). Sama seperti ReactJS, VueJS menggunakan Virtual DOM,

yang membuatnya memiliki performa yang cepat dan stabil, bahkan ketika menangani lalu lintas yang besar.

Beberapa fitur utama dalam VueJS antara lain adalah *data binding*, yang memungkinkan perubahan antarmuka sesuai dengan data yang diberikan secara otomatis, serta *components*, yaitu *template* HTML yang dapat digunakan kembali. Selain itu, VueJS juga memiliki *directives*, yang merupakan atribut HTML untuk memanipulasi elemen DOM, dan *filters*, yang memproses data sebelum ditampilkan di antarmuka. VueJS memiliki banyak kelebihan, termasuk dokumentasi resmi yang lengkap dan mudah dipelajari oleh pemula, serta ukurannya yang ringan, sekitar 33 KB, sehingga memungkinkan pengembangan aplikasi dengan lebih cepat. Selain itu, VueJS mendukung interaksi dua arah berkat konsep MVVM (*Model-View-View-Model*). Namun, *framework* ini juga memiliki beberapa kekurangan, seperti komponen yang belum sepenuhnya stabil dan komunitas yang masih relatif kecil dibandingkan *framework* lain, sehingga dukungan dan forum diskusi masih terbatas (Prasetyo et al., 2022).

2.3.3 *Framework* Laravel dan Blade Templating Engine

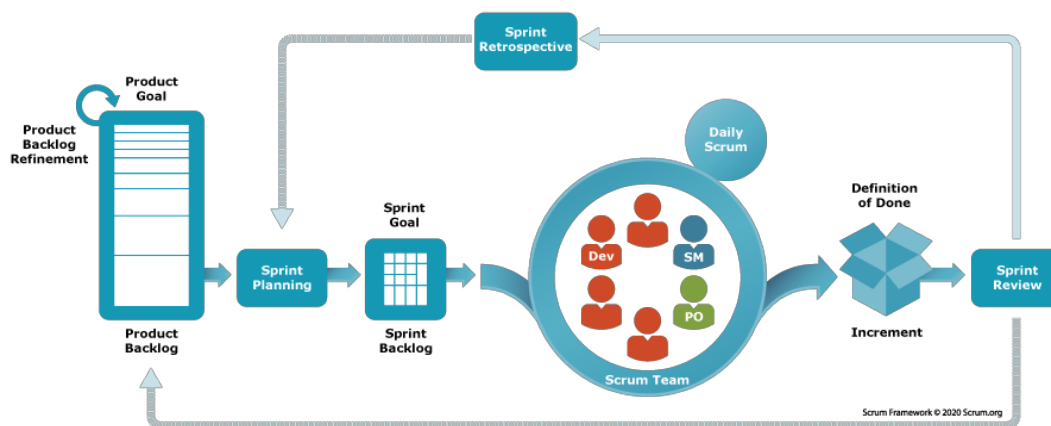
Laravel adalah *framework open source* PHP berbasis web gratis dibuat oleh Taylor Otwell dan ditujukan untuk pengembangan aplikasi web mengikuti *model-view-controller* (MVC) atau pola arsitektur (Luhukay & Mailoa, 2024). Laravel menyediakan berbagai fitur yang memudahkan pengembangan dan pengelolaan aplikasi web, termasuk kemudahan pengelolaan dan keamanan yang baik. Laravel juga dilengkapi dengan Eloquent ORM, yang memudahkan interaksi dengan *database* menggunakan model PHP, dan Blade Templating Engine, yang memudahkan pembuatan dan pengelolaan tampilan dengan *template* yang dinamis (Sinlae et al., 2024).

Blade *templating engine* merupakan salah satu fitur unggulan Laravel yang digunakan untuk pembuatan antarmuka pengguna. Blade menawarkan sintaks yang sederhana dan bersih, memungkinkan pengembang untuk membuat *template* HTML yang dinamis dan mudah dipelihara (Sinlae et al., 2024). Blade mendukung penggunaan *layout* dan *template*, yang memudahkan pengelolaan tampilan yang konsisten di seluruh aplikasi. Selain itu, Blade juga memiliki keunggulan dalam performa karena semua *template* dikompilasi menjadi kode PHP biasa dan disimpan dalam *cache* hingga ada perubahan, sehingga menambah *overhead* yang sangat kecil pada aplikasi (Wijayanto & Firmansyah, 2022). *File template* Blade menggunakan ekstensi *.blade.php* dan biasanya disimpan di direktori *resources/views*. Blade juga menyediakan fitur-fitur seperti *template inheritance* dan *sections* yang memungkinkan

pengembang untuk mengorganisir kode tampilan dengan lebih baik, mengurangi redundansi, dan meningkatkan *maintainability* kode (Sinlae et al., 2024). Selain itu, Blade memungkinkan pengembang untuk menambahkan logika PHP langsung ke dalam *template* dengan cara yang aman dan efisien, sehingga mempercepat proses pengembangan antarmuka pengguna.

2.4 Metode *Scrum*

Scrum adalah metode manajemen proyek *agile* yang digunakan secara luas dalam pengembangan perangkat lunak untuk meningkatkan kolaborasi tim dan respons terhadap perubahan kebutuhan proyek. Menurut Schwaber & Sutherland (2020), *scrum* menggunakan pendekatan inkremental dan iteratif untuk memaksimalkan kemungkinan keberhasilan sekaligus mengendalikan risiko yang ada. Metode ini membagi proyek menjadi *sprint-sprint* pendek dengan durasi satu hingga empat minggu, di mana tim bekerja pada sejumlah tugas tertentu dari *backlog* produk. *Sprint*, yang merupakan inti dari *scrum*, di mana ide diubah menjadi nilai, adalah acara berdurasi tetap selama satu bulan atau kurang agar tercipta konsistensi. Setelah satu *sprint* berakhir, *sprint* baru segera dimulai. Seluruh pekerjaan yang dibutuhkan untuk mencapai *product goal*, termasuk *sprint planning*, *daily scrum*, *sprint review*, dan *sprint retrospective*, terjadi dalam *sprint* seperti yang terlihat pada Gambar 2.1. Setiap *sprint* diakhiri dengan evaluasi hasil kerja melalui pertemuan *sprint review* dan *retrospective* untuk meningkatkan proses kerja tim di *sprint* berikutnya. Keuntungan utama dari metode *Scrum* adalah fleksibilitasnya dalam menghadapi perubahan kebutuhan selama siklus hidup proyek. Tim dapat beradaptasi dengan cepat terhadap umpan balik dari pemangku kepentingan tanpa harus memulai ulang seluruh proyek atau membuat perubahan besar pada rencana awal.



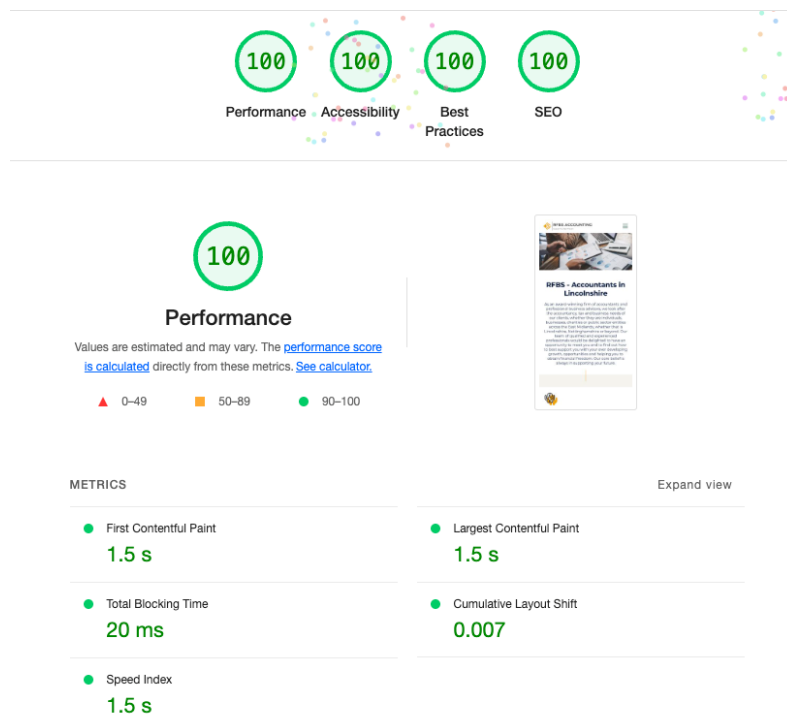
Gambar 2.1 Tahapan aktivitas dalam metode *Scrum*

Sumber: (Scrum.org, 2020)

2.5 Lighthouse

Lighthouse adalah alat otomatis *open-source* yang digunakan untuk mengaudit kualitas halaman web dalam hal performa, aksesibilitas, *progressive web app*, *search engine optimization* (SEO) serta praktik terbaik lainnya (Siahaan & Vianto, 2022). Alat ini memberikan laporan terperinci mengenai berbagai aspek *website* serta rekomendasi perbaikan berdasarkan standar industri terkini. Lighthouse sangat berguna bagi para tim pengembang web untuk memastikan bahwa aplikasi mereka memenuhi standar kualitas tinggi sebelum diluncurkan ke publik.

Kinerja berkaitan dengan pengukuran seberapa cepat *browser* dapat menyusun halaman web. Lighthouse menggunakan *browser* bernama Chromium untuk *me-render* halaman dan menjalankan pengujian saat halaman tersebut dibuat. Alat ini bersifat *open-source* dan terdapat empat metrik yang diukur oleh Lighthouse, yaitu kinerja (*performance*), aksesibilitas (*accessibility*), praktik terbaik (*best practices*), dan SEO (Muna et al., 2022). Siahaan & Vianto (2022) menguraikan enam kriteria spesifik untuk mengukur kinerja *website*, namun versi terbaru dari Lighthouse menyederhanakannya menjadi lima kriteria, termasuk *First Contentful Paint* (FCP), *Largest Contentful Paint* (LCP), *Total Blocking Time* (TBT), *Cumulative Layout Shift* (CLS), dan *Speed Index* (SI), dengan tingkat kinerja yang diukur berdasarkan waktu atau pergeseran *layout*.



Gambar 2.2 Contoh skor sempurna Google Lighthouse terhadap sebuah *website*

Sumber: (TJS, 2023)

Gambar 2.2 merupakan contoh skor sempurna yang diberikan oleh Google Lighthouse (TJS, 2023), menunjukkan bahwa situs web tersebut berhasil memenuhi kriteria tertinggi untuk kinerja, aksesibilitas, praktik terbaik, dan SEO. Skor sempurna ini mencerminkan bahwa semua aspek yang dinilai oleh Lighthouse telah dioptimalkan dengan sangat baik. Alat ini sangat berguna bagi pengembang web dalam mengukur kualitas aplikasi berdasarkan kriteria evaluasi yang jelas dan standar yang terukur.

2.6 Chrome DevTools Performance tab

Menurut Kaleev (2024), tab *Performance* di Chrome DevTools adalah serangkaian alat dan metrik yang membantu menganalisis seberapa baik situs web atau aplikasi dimuat dan berjalan. Tab ini melacak berbagai aspek seperti waktu pemuatan halaman, eksekusi JavaScript, *rendering*, dan interaksi pengguna, serta memberikan wawasan terperinci tentang penggunaan sumber daya dan hambatan yang memengaruhi kecepatan dan responsivitas situs. Meskipun lebih kompleks dibandingkan alat pengembang lainnya, Chrome DevTools menawarkan berbagai keunggulan, meliputi:

a. Data *Real-time*

Sebagian besar alat kecepatan halaman, seperti PageSpeed Insights, hanya menyediakan laporan statis. Namun, Tab *Performance* di DevTools memungkinkan pengguna melihat respons situs secara *real-time* ketika mereka berinteraksi, seperti saat menggulir, mengklik tombol, atau menavigasi halaman. Hal ini memberikan gambaran yang lebih akurat tentang pengalaman pengguna nyata.

b. Analisis *Core Web Vitals* Langsung

Core Web Vitals adalah metrik utama yang digunakan Google untuk menilai performa situs, seperti seberapa cepat situs dimuat (LCP), seberapa stabil *layout* halaman (CLS), dan seberapa cepat responsnya terhadap interaksi pengguna (INP). Dengan DevTools, metrik ini ditampilkan langsung selama interaksi, sehingga dapat membantu mengidentifikasi masalah performa yang bisa mengganggu pengalaman pengguna dengan cepat.

c. Kustomisasi Pengujian

DevTools memungkinkan untuk simulasi bagaimana situs akan dimuat di perangkat berbeda atau dalam kondisi internet lambat. Ini sangat berguna untuk

menguji kinerja situs di berbagai skenario, seperti di perangkat dengan spesifikasi lebih rendah atau koneksi internet yang lambat.

Dengan metrik *Core Web Vitals* yang kini dikumpulkan secara otomatis di latar belakang, pengguna dapat memeriksa performa kapan saja tanpa harus memulai perekaman. Dengan kata lain, setiap kali merasa situs web lambat, cukup mengakses panel *Performance* dan mengetahui alasannya. Hasil metrik ini mewakili metrik performa yang dikumpulkan dari sesi penjelajahan lokal, termasuk kondisi jaringan dan perangkat, dengan pembaruan *real-time* saat memuat atau berinteraksi dengan halaman. Setiap interaksi di situs akan menghasilkan data *real-time* terkait *Largest Contentful Paint* (LCP), *Cumulative Layout Shift* (CLS), dan *Interaction to Next Paint* (INP) seperti yang terlihat pada Gambar 2.3, di mana semua metrik ini diberi kode warna (hijau, oranye, dan merah) berdasarkan ambang performa untuk memudahkan identifikasi masalah yang terjadi (Kaleev, 2024).



Gambar 2.3 Metrik Chrome DevTools *Performance Tab* dengan kode warna berdasarkan ambang performa

Sumber: (Kaleev, 2024)

2.7 Tinjauan Pustaka

Pada subbab ini, penulis membahas beberapa hasil penelitian sebelumnya yang memiliki kesamaan atau relevansi dengan topik utama dalam laporan akhir ini. Pembahasan ini bertujuan untuk mengidentifikasi persamaan dan perbedaan dari setiap penelitian terkait. Hasil penelitian tersebut disajikan pada Tabel 2.1.

Tabel 2.1 Tinjauan Pustaka Penelitian Terdahulu

No.	Judul	Tujuan	Teknologi	Hasil
1.	Perbandingan Pengembangan <i>Front End</i> Menggunakan Blade Template dan Vue Js	Untuk mengetahui <i>framework</i> yang cocok untuk digunakan pada saat membangun sebuah <i>website</i> (Chastro & Darmawan, 2020).	Blade Templating Engine, VueJS	Dalam penelitian ini, untuk membangun aplikasi sederhana sebaiknya menggunakan Blade <i>Template</i> , akan tetapi jika aplikasi tersebut kompleks dapat menggunakan Vue Js (Chastro & Darmawan, 2020).
2.	Tokopedia <i>and</i> Shopee <i>Marketplace Performance Analysis Using Metrix</i> Google Lighthouse	Untuk menilai kinerja, aksesibilitas, praktik terbaik, dan SEO kedua <i>platform</i> , sehingga dapat memberikan wawasan untuk perbaikan di masa depan dan menjadi panduan dalam upaya optimasi kedua <i>marketplace</i> (Muna et al., 2022).	ReactJS	Tokopedia lebih unggul dibandingkan Shopee dalam sebagian besar metrik Lighthouse. Peneliti merekomendasikan agar Shopee memprioritaskan peningkatan kinerja dan aksesibilitas untuk meningkatkan pengalaman pengguna secara keseluruhan (Muna et al., 2022).

3.	Analisis Perbandingan <i>Framework Front-End</i> Javascript React dan Vue Pada Pengembangan <i>Website</i>	Untuk memberikan rekomendasi yang berguna bagi pengembang <i>website</i> dan membantu membuat keputusan yang lebih informatif tentang pemilihan <i>framework front-end</i> untuk pengembangan <i>website</i> (Sofi'ie & Qoiriah, 2023).	Vue JS, React JS	VueJS memiliki performa dan efisiensi yang lebih baik dibandingkan dengan <i>framework</i> ReactJS (Sofi'ie & Qoiriah, 2023).
4.	Analisis Perbandingan <i>Framework Front-End</i> Javascript Solid JS Dan Vue JS pada Pengembangan <i>Website Interaktif</i>	Untuk mengetahui apakah <i>framework</i> baru yang kurang populer dapat bersaing dengan <i>framework</i> yang sudah mapan dan memberikan rekomendasi yang lebih informatif dalam pemilihan <i>framework front-end</i> JavaScript untuk pengembangan <i>website</i> interaktif (Khoirurrizal et al., 2024).	Solid JS, Vue JS	SolidJS menunjukkan keunggulan dalam performa, efisiensi, dan penggunaan memori, meskipun sedikit kurang stabil dibandingkan VueJS yang memiliki tingkat stabilitas lebih baik. Meskipun SolidJS kurang dikenal, hasil ini menegaskan bahwa ia merupakan pesaing yang layak dalam pengembangan <i>website</i> interaktif (Khoirurrizal et al., 2024).
5.	<i>Comparative Analysis Study of Front-End JavaScript Frameworks Performance Using Lighthouse Tool</i>	Untuk memberikan pemahaman tentang kinerja web melalui analisis perbandingan beberapa <i>framework</i> JavaScript, seperti React, Angular, Vue, Svelte, dan SolidJS, dengan Lighthouse, yang diharapkan dapat memberikan masukan bagi para pengembang <i>website</i> dalam	React, Angular, Vue, Svelte, dan SolidJS	Perbandingan kinerja <i>Weather App</i> dengan berbagai <i>framework front-end</i> menunjukkan bahwa Vue memberikan kinerja terbaik, walaupun <i>framework</i> lainnya juga mampu menghasilkan kinerja

		mempertimbangkan dan mempelajari <i>framework</i> yang relevan (Siahaan & Vianto, 2022).		yang baik meskipun tidak secepat Vue (Siahaan & Vianto, 2022).
6.	Top 5 <i>E-Commerce Performance Analysis Using Google Lighthouse Matrix</i>	Membantu pengembang meningkatkan kualitas <i>website</i> dengan memberikan wawasan nyata berdasarkan metrik yang ditetapkan untuk perbaikan di masa depan (Adli et al., 2023).	ReactJS, Perl, Ruby on Rail, VueJS	Berdasarkan hasil audit pada lima <i>e-commerce</i> teratas menggunakan Google Lighthouse, Tokopedia mencetak skor tertinggi dalam kinerja tetapi perlu mengoptimalkan <i>total blocking time</i> . Sementara itu, Bukalapak unggul dalam aksesibilitas dengan skor 97 (indeks hijau), Tokopedia unggul dalam <i>best practices</i> dengan skor sempurna 100 (indeks hijau), dan keduanya memiliki skor SEO 93 (indeks hijau), menunjukkan bahwa mereka telah memenuhi standar dasar SEO. Namun, di sisi lain, masih banyak aspek yang perlu dioptimalkan, terutama dalam hal kinerja, untuk mencapai hasil yang lebih baik (Adli et al., 2023).
7.	<i>Comparison of JavaScript Frontend Frameworks –</i>	Membandingkan tiga <i>framework frontend</i> JavaScript yang populer— Angular, React, dan Vue—berdasarkan kinerja dan popularitasnya,	Angular, React, dan VueJS	Vue menawarkan kinerja terbaik untuk aplikasi kecil hingga menengah, sementara React tetap menjadi pilihan yang paling

	Angular, React, and Vue	untuk memberikan wawasan kepada pengembang dan pengambil keputusan agar dapat memilih <i>framework front-end</i> yang tepat untuk proyek kecil hingga menengah (Karić & Durmić, 2024).	populer di kalangan pengembang. Angular lebih cocok untuk aplikasi skala besar di mana kecepatan kinerja bukan menjadi prioritas utama. Pemilihan <i>framework</i> yang tepat bergantung pada kebutuhan spesifik proyek dan fitur-fitur yang dimiliki setiap <i>framework</i> (Karić & Durmić, 2024).
--	-------------------------	--	---

Berdasarkan kajian pustaka yang telah dijabarkan pada Tabel 2.1, dapat dilihat bahwa sebagian besar penelitian terdahulu berfokus pada perbandingan antara berbagai *framework front-end* JavaScript seperti Vue, React, Angular, dan SolidJS. Sebagian besar studi tersebut menggunakan Google Lighthouse sebagai alat ukur untuk mengevaluasi kinerja *framework* dalam pengembangan web serta memberikan wawasan tentang kelebihan dan kekurangan masing-masing *framework* dalam konteks pengembangan web. Hasil-hasil penelitian menunjukkan bahwa VueJS sering mengungguli *framework* lainnya dalam hal performa untuk aplikasi skala kecil hingga menengah (Siahaan & Vianto, 2022; Sofi'ie & Qoiriah, 2023), sementara ReactJS lebih populer dan cocok untuk pengembangan aplikasi yang lebih kompleks (Karić & Durmić, 2024). Di sisi lain, Blade *Template* dinilai lebih sesuai untuk pengembangan aplikasi sederhana (Chastro & Darmawan, 2020). Meskipun telah ada beberapa studi yang membandingkan *framework-framework* tersebut, belum ada penelitian yang secara khusus membandingkan kinerja ReactJS, VueJS, dan Blade *Templating Engine*. Oleh karena itu, laporan ini diharapkan dapat memberikan kontribusi baru dengan menganalisis perbandingan kinerja ketiga *framework* tersebut dalam pengembangan aplikasi web.

BAB III

PELAKSANAAN MAGANG

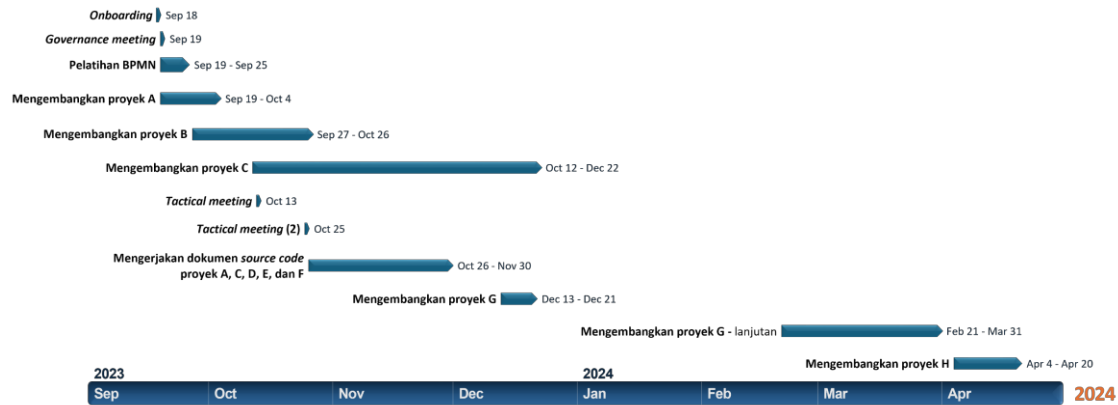
3.1 Aktivitas Magang

Kegiatan magang sebagai *frontend (React) programmer* di PT Javan Cipta Solusi berlangsung selama kurang lebih enam bulan, mulai dari 18 September 2023 hingga 20 April 2024. Selama periode tersebut, penulis berkontribusi dalam pengembangan delapan proyek berbasis web, di mana tiga di antaranya melibatkan penulis secara khusus dalam pembuatan dokumentasi *source code* untuk kebutuhan teknis. Aktivitas-aktivitas magang lainnya dapat dilihat pada Tabel 3.1, dan penjelasan lebih detail terkait setiap aktivitas akan dibahas pada subbab berikutnya.

Tabel 3.1 Aktivitas Magang

No.	Deskripsi Kegiatan	Waktu	Durasi
1.	<i>Onboarding</i>	18 September 2023	1 Hari
2.	<i>Governance meeting</i>	19 September 2023	1 Hari
3.	Pelatihan BPMN	19-25 September 2023	1 Minggu
4.	Membantu mengembangkan proyek A (Sistem Informasi <i>Jobseeker</i>)	19 September - 4 Oktober 2023	2 Minggu
5.	Membantu mengembangkan proyek B (Modul Administrasi Keuangan)	27 September - 26 Oktober 2023	1 Bulan
6.	Membantu mengembangkan proyek C (Layanan API dan <i>Widget</i>)	12 Oktober - 22 Desember 2023	2 Bulan
7.	<i>Tactical meeting</i>	13 Oktober 2023, 25 Oktober 2023	1 Hari (sesuai arahan <i>tribe lead</i>)
8.	Mengerjakan dokumen <i>source code</i> proyek A, C, D (Aplikasi Presensi Peserta Ujian), E (Showcase Portfolio Peserta), dan F (Web <i>Crawling</i> Data Sosial Media)	26 Oktober - 30 November 2023	1 Bulan
9.	Membantu mengembangkan proyek G (<i>Platform</i> Ujian – Pelatihan <i>Coding</i>)	13-21 Desember 2023, 21 Februari - 31 Maret 2024	1 Bulan 10 Hari
10.	Membantu mengembangkan proyek H (Sistem Informasi Pengendalian Kasus)	4-20 April 2024	2 Minggu

Sebagai pelengkap dari Tabel 3.1, Gambar 3.1 berikut menunjukkan *timeline* kegiatan magang. Gambar ini memberikan gambaran visual mengenai alur dan durasi setiap aktivitas selama periode magang, sehingga memudahkan pembaca untuk memahami rentang waktu serta keterlibatan penulis dalam berbagai proyek yang dijalankan.

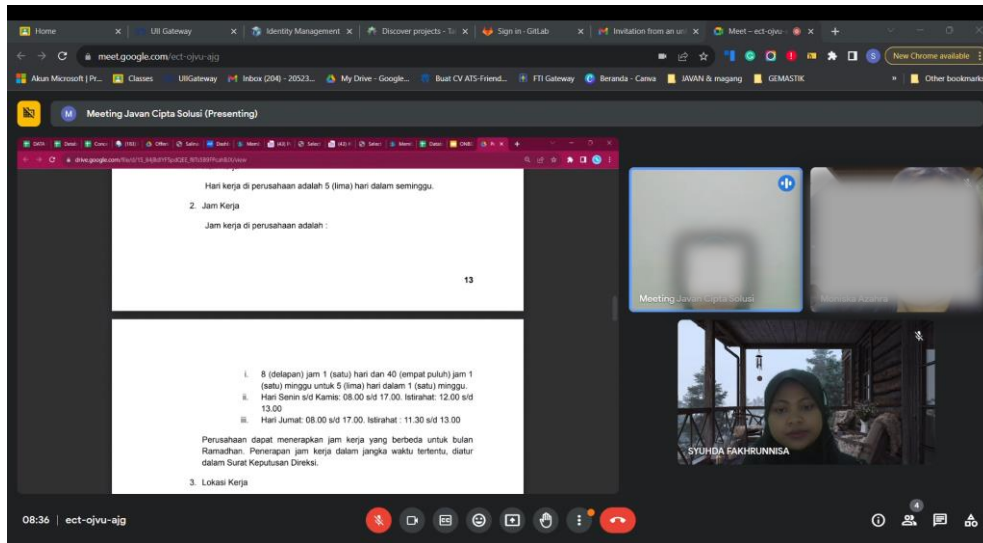


Gambar 3.1 *Timeline* aktivitas magang

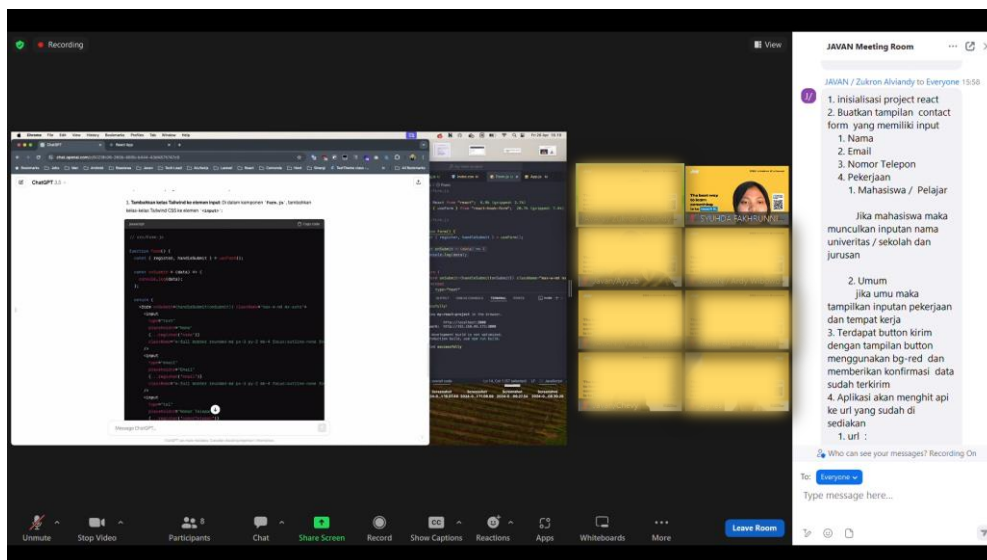
3.1.1 Onboarding

Kegiatan *onboarding* merupakan tahap awal yang wajib diikuti oleh setiap karyawan baru di PT Javan Cipta Solusi. Pelaksanaan *onboarding* kali ini dilakukan secara *online* dan dipimpin oleh perwakilan tim *people growth*, mengingat adanya peserta dari dua lokasi berbeda, yaitu karyawan baru di Cimahi dan penulis yang berada di kantor Yogyakarta. Seperti yang ditunjukkan pada Gambar 3.2, kegiatan *onboarding* dilaksanakan melalui *platform video conference* yang memperlihatkan interaksi antara tim *people growth* dengan para karyawan baru dalam sesi orientasi virtual. Dalam sesi ini, tim *people growth* memberikan pemaparan komprehensif mengenai perusahaan, mencakup sejarah, peraturan, dan struktur organisasi. Informasi penting lainnya yang disampaikan adalah jadwal kerja perusahaan yang berlangsung dari pukul 08.00 hingga 17.00 WIB dengan waktu istirahat satu jam pada pukul 12.00 WIB. Khusus untuk hari Jumat, waktu istirahat lebih panjang yaitu 1,5 jam.

Selain itu, PT Javan Cipta Solusi juga memiliki tradisi mingguan yang disebut ‘Hari Berkualitas’ yang diselenggarakan setiap Jumat. Kegiatan ini diisi dengan berbagai agenda seperti *meeting*, *town hall*, *workshop*, atau pelatihan, yang bertujuan untuk meningkatkan kompetensi karyawan. Gambar 3.3 merupakan salah satu pelatihan yang diadakan pada hari berkualitas untuk para *programmers* yang bertemakan *coding with AI*, dan Gambar 3.4 menunjukkan suasana ketika dilaksanakan kegiatan *town hall* yang wajib dihadiri oleh seluruh karyawan, dengan pihak manajemen yang sedang menyampaikan *updates* penting mengenai perkembangan dan pencapaian perusahaan, serta rencana-rencana strategis ke depan.



Gambar 3.2 Onboarding magang

Gambar 3.3 Pelatihan yang diadakan pada hari berkualitas untuk para *programmers*Gambar 3.4 Kegiatan *town hall* Januari 2024

Selama proses *onboarding*, tim *people growth* memberikan beberapa tugas kepada penulis melalui platform manajemen proyek ActiveCollab. Seperti yang ditunjukkan pada Gambar 3.5, platform ini menampilkan *task* yang berisi daftar hal-hal yang harus diselesaikan sebelum memulai pekerjaan sebagai Magang *Frontend (React) Programmer*. Tugas-tugas tersebut mencakup verifikasi akses ke berbagai *tools* yang akan digunakan selama di Javan, pengerjaan soal evaluasi, pelaksanaan *typing test*, dan terdapat beberapa *subtasks* lainnya. *Task* ini harus diselesaikan pada hari yang sama, dan setiap *subtask* memiliki persyaratan khusus di mana penulis harus melampirkan bukti penyelesaian berupa *screenshot* pada kolom komentar sebelum dapat menandai *subtask* tersebut sebagai selesai.

The screenshot displays the ActiveCollab interface for a task titled "Berhasil menyelesaikan onboarding (magang) - Syuhda Fakhrunnisa (Magang Frontend (React) Programmer)". The task is assigned to Syuhda Fakhrunnisa and is due on Sep 18, 2023. The task list includes various subtasks such as "Dapat memahami SOP Target", "Goal Evaluasi Tim Teknis-Magang", and "Berhasil Download Telegram Desktop". The discussion section shows a comment from Syuhda F. on Sep 18, 2023, at 11:31, stating "Hasil kecepatan mengetik: 71 - 74 - 70". Below the comment are three screenshots of typing test results: "fastfingers1.png" (71 WPM), "fastfingers2.png" (74 WPM), and "fastfingers3.png" (70 WPM). The sidebar on the right shows the task list, time tracking (0:00), expense tracking (0 IDR), and subscribers (Agnes Nabela Pratiwi, Anita Nur Hidayati, Dicky Puja Pratama, Noviani Widyalisti, Rahmi Rahayu Fatihan, Syuhda Fakhrunnisa).

Gambar 3.5 *Task onboarding* magang

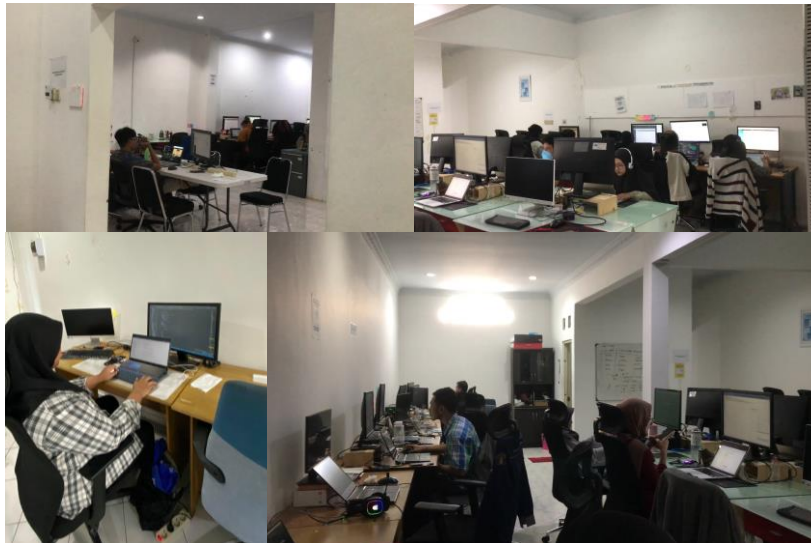
Suasana Kerja di PT Javan Cipta Solusi

Sekitar satu minggu setelah *onboarding*, PT Javan Cipta Solusi mengadakan sesi perkenalan untuk setiap anggota tim baru kepada seluruh karyawan. Acara perkenalan ini biasanya dilaksanakan sesuai jam makan siang, di mana seluruh karyawan yang bekerja dari kantor (WFO) diminta berkumpul di ruangan yang telah ditentukan oleh tim *people growth*. Mereka kemudian membentuk lingkaran besar, menciptakan suasana yang mirip dengan yang terlihat pada Gambar 3.6. Acara yang dipandu oleh tim *people growth* ini dimulai dengan perkenalan dari anggota tim baru, dilanjutkan dengan perkenalan singkat dari seluruh karyawan yang hadir dengan menyebutkan nama dan peran masing-masing di Javan. Sesi perkenalan ini bertujuan membantu anggota baru mengenali rekan-rekan kerjanya meskipun hanya secara garis besar, mengingat jumlah tim yang cukup besar. Rangkaian acara kemudian dilengkapi dengan sambutan dari CEO, *operational lead*, atau *tribe lead*, sebelum akhirnya ditutup kembali oleh tim *people growth*.



Gambar 3.6 Perkenalan anggota tim baru di PT Javan Cipta Solusi

Setelah acara perkenalan selesai, seluruh karyawan dipersilakan untuk melanjutkan aktivitas masing-masing. Namun, jika waktu istirahat telah habis, mereka akan kembali ke meja masing-masing dan melanjutkan pekerjaannya. Seperti yang ditunjukkan pada Gambar 3.7, suasana kerja di PT Javan Cipta Solusi terlihat profesional dan kondusif, dengan para karyawan yang fokus bekerja di meja masing-masing menggunakan perangkat kerja yang telah disediakan perusahaan.



Gambar 3.7 Suasana kerja di PT Javan Cipta Solusi

Kunjungan Dosen Informatika ke PT Javan Cipta Solusi

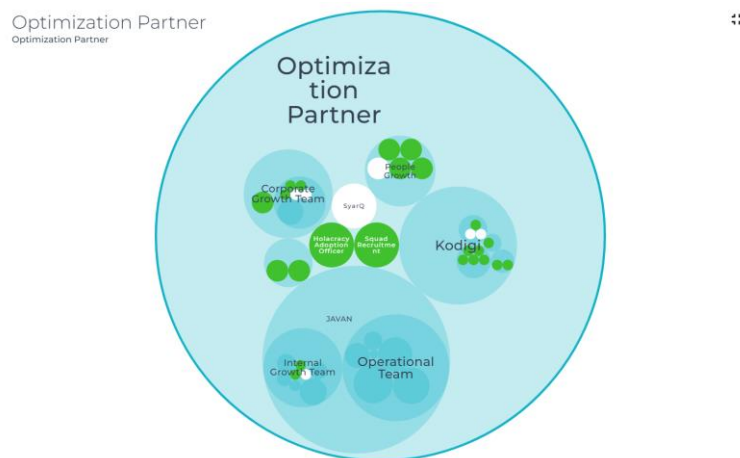
Pada 11 Oktober 2023, sekitar satu bulan setelah *onboarding*, PT Javan Cipta Solusi menerima kunjungan dari Bapak Hanson Prihantoro Putro, dosen Informatika UII. Sebagai dosen dan pembina mahasiswa jalur magang, beliau datang untuk memantau mahasiswa yang sedang menjalani magang, termasuk penulis. Kunjungan tersebut disambut oleh tim *people growth* dan dilanjutkan dengan sesi obrolan bersama *operational lead*, *tribe lead*, dan penulis. Diskusi yang berlangsung sekitar satu jam ini mencakup berbagai topik, seperti proyek-proyek yang ditangani penulis, pengalaman selama magang di perusahaan, dan hal-hal terkait lainnya. Seperti yang terlihat pada Gambar 3.8, suasana kunjungan diabadikan ketika Pak Hanson memberikan tanda mata kepada *operational lead* sebelum melanjutkan kunjungan ke perusahaan tempat mahasiswa jalur magang lainnya.



Gambar 3.8 Kunjungan dosen Informatika UII ke PT Javan Cipta Solusi

3.1.2 Governance Meeting

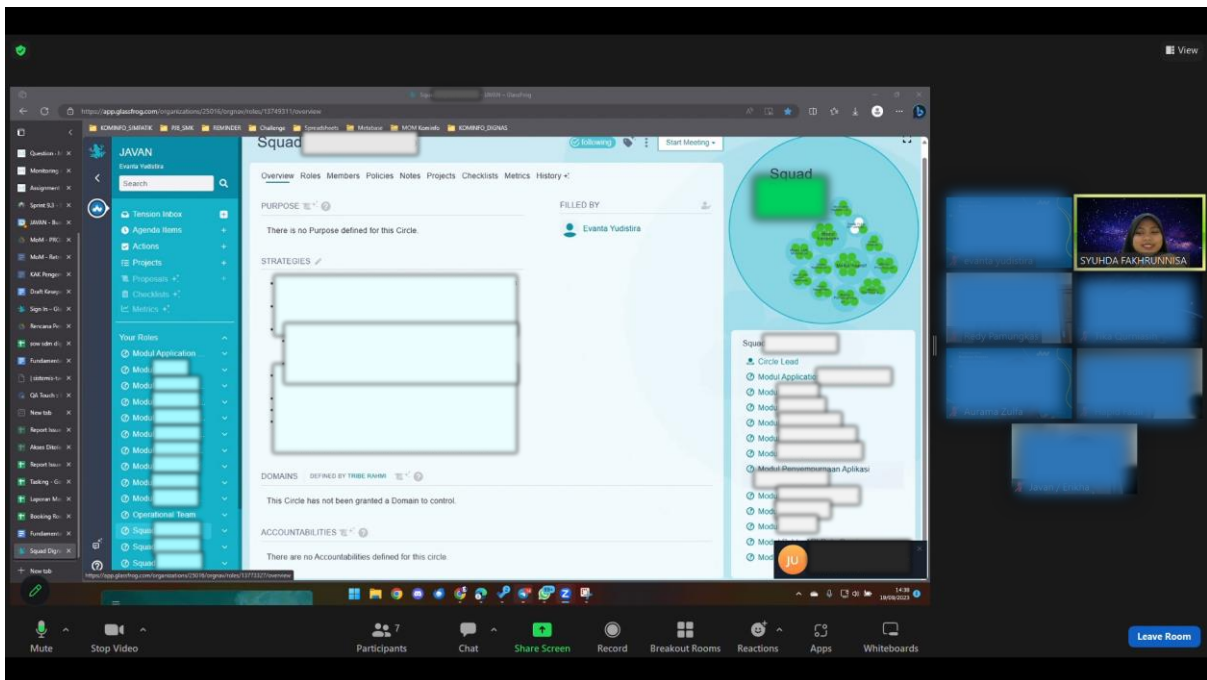
Gambar 3.9 menampilkan bagan struktur organisasi PT Javan Cipta Solusi yang menerapkan manajemen sistem *holacracy*, suatu manajemen sistem yang menggunakan model struktur *self-management organization*. Tidak seperti manajemen sistem *hierarchy* yang berbentuk piramida, manajemen sistem *holacracy* menjelaskan struktur organisasi dengan bentuk lingkaran besar/*circle*. Dalam bagan tersebut, tampak jelas bagaimana struktur organisasi disusun dalam bentuk *circle* utama yang terhubung dengan *sub-circle*, sesuai dengan prinsip *self-management*. Setiap *circle* dalam organisasi ini memiliki kewenangan untuk mengatur diri sendiri dan menentukan anggota serta peran yang dibutuhkan untuk mencapai tujuan bersama. Untuk mendukung implementasi sistem *holacracy* secara efektif, PT Javan Cipta Solusi menggunakan aplikasi bernama GlassFrog. Aplikasi ini dikembangkan oleh HolacracyOne, sebuah perusahaan konsultan yang memelopori pengembangan metode *holacracy*. Dengan GlassFrog, semua yang terlibat di dalam perusahaan dapat melihat *responsibilities*, *accountabilities*, serta *action* berdasarkan *role* dan *circle* mereka. Selain itu GlassFrog juga memfasilitasi dan menunjukkan bagaimana cara berkomunikasi antar *circle* di dalam manajemen sistem *holacracy*, yaitu dengan adanya fitur *meeting* yang terbagi menjadi dua jenis, yakni *tactical meeting* dan *governance meeting* (Kamal, 2022).



Gambar 3.9 Struktur organisasi PT Javan Cipta Solusi

Governance meeting diadakan dengan frekuensi yang lebih jarang dibanding *tactical meeting*. Sesi ini digunakan untuk membahas hal-hal yang struktural dalam *circle*. Sebagai contoh, *governance meeting* menghasilkan perubahan *accountabilities* dan penambahan anggota pada *role* tertentu (Kamal, 2022). Oleh karena itu, ketika penulis bergabung dengan tim proyek, diadakanlah *governance meeting* untuk memperkenalkan anggota baru yang bergabung ke dalam *circle* proyek. Gambar 3.10 menampilkan suasana *governance meeting*

yang dilaksanakan secara virtual melalui Zoom, di mana penulis berkesempatan untuk memperkenalkan diri sebagai anggota magang baru dalam tim. Pertemuan ini dihadiri sebagian besar oleh para *product owner*. Selama sesi ini, penulis diberitahu tentang proyek yang akan ditangani dan siapa *product owner* yang terlibat. Selain itu, *product owner* juga memperkenalkan struktur tim proyeknya saat ini dan menjelaskan adanya penambahan *role* dalam proyek tersebut, di mana penulis akan berperan sebagai *front-end programmer*.



Gambar 3.10 Governance meeting

3.1.3 Pelatihan BPMN

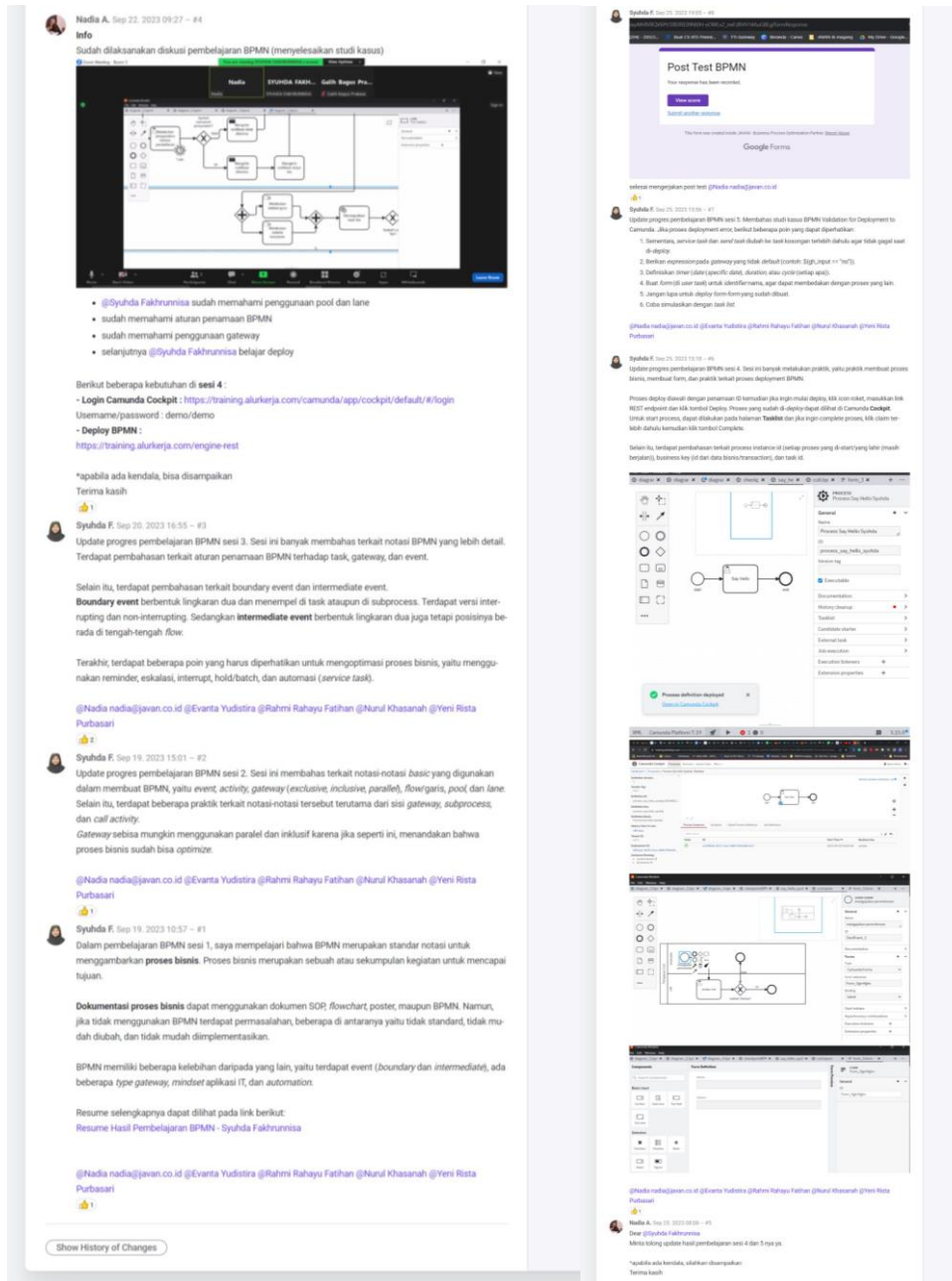
Sebagai karyawan magang baru di PT Javan Cipta Solusi, pelatihan BPMN merupakan salah satu hal yang wajib diikuti dan dikerjakan dengan baik. Pada hari pertama setelah *onboarding*, penulis dihubungi oleh seorang *competency development officer* Javan dan diberikan informasi mengenai pembelajaran BPMN. Beliau menjelaskan bahwa, sebagai anggota tim baru, penulis harus menyelesaikan pembelajaran BPMN, yang merupakan standar kompetensi di Javan, mengingat PT Javan Cipta Solusi adalah *A Business Process Optimization Partner* yang menggunakan BPMN dan Camunda sebagai teknologi utama untuk membantu mengoptimalkan proses bisnis.

Pembelajaran BPMN dilakukan secara asinkron di ActiveCollab, seperti yang terlihat pada Gambar 3.11. Penulis diharuskan untuk menonton video pembelajaran yang terlampir di deskripsi *task*, kemudian membuat resume dari hasil yang dipelajari per sesi, dengan total

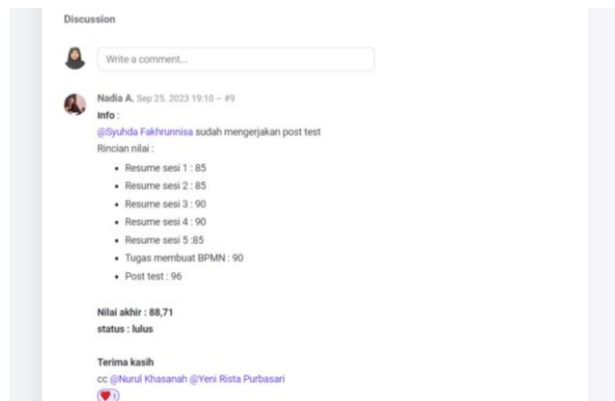
terdapat lima sesi. Setelah mempelajari setiap sesi, peserta harus memperbarui hasilnya di kolom komentar ActiveCollab, seperti yang terlihat pada Gambar 3.12, dengan mencantumkan informasi mengenai apa yang dipelajari dari sesi tersebut, bukti percobaan dalam bentuk *screenshot*, serta kendala yang dihadapi jika ada. Waktu yang diberikan untuk *coaching* ini adalah lima hari, di mana setelah sesi tiga, akan diadakan cek poin, untuk mendiskusikan kendala yang dihadapi penulis dan membahas studi kasus bersama perwakilan *competency development officer*. Jika seluruh sesi telah dikerjakan, peserta akan menghadapi *post-test* di akhir sesi. Setelah hasil *post-test* keluar, rincian nilai akan diberikan di kolom komentar, seperti yang terlihat pada Gambar 3.13. Dalam pengumuman tersebut, dinyatakan apakah penulis lulus atau tidak, dan penulis berhasil meraih nilai akhir yang cukup memuaskan, yaitu 88.71, dengan status lulus.

The screenshot shows a task page in ActiveCollab. The task title is "Tersedia-update-hasil-pembelajaran-BPMN" assigned to "Syuhda.F". The task description asks for an update on the BPMN competency test results, including a resume, screenshots, and any issues. It provides a list of learning materials (BPMN sessions 1-5) and a list of subtasks. The subtasks are: "Ade nilai akhir peserta", "Berhasil mengerjakan post-test", and "Ade hasil pembelajaran sesi 5". The task also includes a "Time Tracking" section with a timer set to 0:00 and an "Expense Tracking" section with a total of 0 IDR.

Gambar 3.11 *Task* pelatihan BPMN



Gambar 3.12 Detail pengerjaan *task* pelatihan BPMN



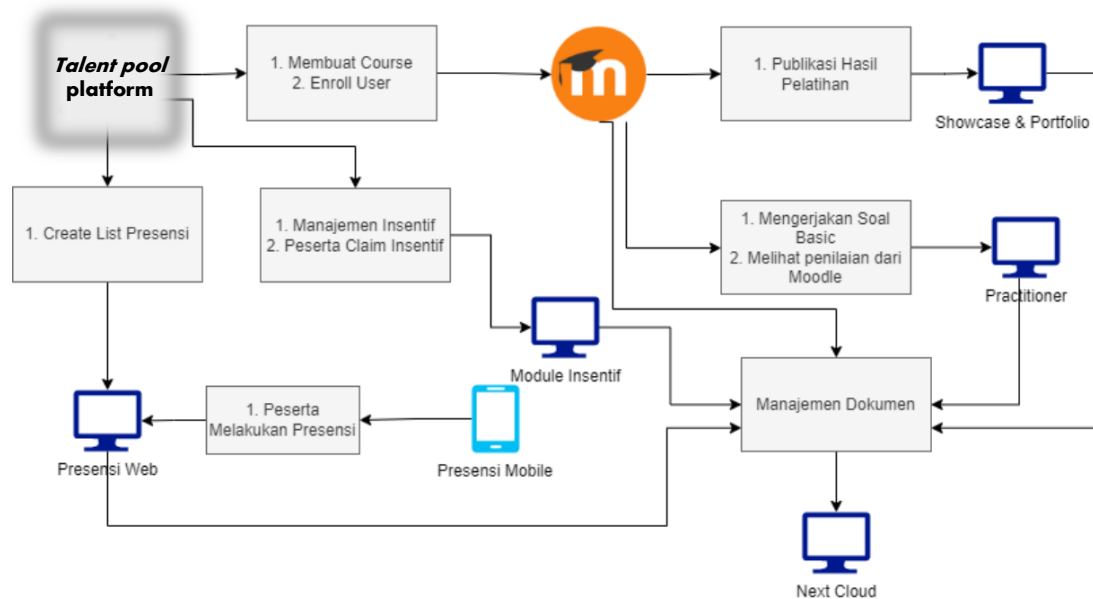
Gambar 3.13 *Feedback* penilaian hasil pelatihan BPMN di ActiveCollab

3.1.4 Membantu Mengembangkan Proyek A

Setelah melaksanakan *onboarding*, mengerjakan latihan BPMN, dan menghadiri *governance meeting*, penulis resmi bergabung dalam salah satu proyek, yaitu Proyek A. Proyek A adalah pengembangan aplikasi berbasis web yang bertujuan menjembatani kesenjangan antara masyarakat muda Indonesia yang mencari pekerjaan sesuai keterampilan mereka, dengan perusahaan yang mencari karyawan potensial. Sistem informasi ini mendukung pertumbuhan ekonomi dan kemajuan karier bagi generasi muda Indonesia. Selama magang, penulis tergabung dalam tim *programmer* sebagai *mentee programmer*. Proses pengembangan proyek A melibatkan peran lain dalam tim, termasuk *product owner*, satu *analyst*, dan empat *backend programmers*. Dalam aktivitas pengembangan proyek, komunikasi dilakukan melalui Telegram dan Taiga. Taiga juga digunakan sebagai *tools* manajemen proyek, di mana *analyst* menggunakan *tools* ini untuk mendistribusikan *tasks* dalam satu *sprint* kepada setiap *programmer* yang terlibat.

3.1.5 Membantu Mengembangkan Proyek B

Seiring berjalannya waktu, penulis dirilis dari proyek A karena jumlah *programmer* sudah mencukupi dan sebagian besar *issue* lebih berfokus pada sisi *backend*. Selain itu, proyek A menggunakan *framework* Laravel dan Blade yang kurang sesuai dengan peran penulis sebagai *frontend (React) programmer*. Penulis kemudian dipindahkan ke proyek B yang menggunakan React sebagai *framework* di sisi *frontend*. Proyek B merupakan modul administrasi keuangan yang digunakan untuk mengelola data kegiatan (realisasi), perjalanan dinas, dan kuitansi. Seperti yang terlihat pada Gambar 3.14, proyek ini terintegrasi dengan beberapa modul lainnya yaitu proyek D (aplikasi presensi peserta), E (*showcase portfolio* peserta), dan G (platform ujian – aplikasi pelatihan *coding*). Dalam proyek B, penulis berperan sebagai (*mentee*) *frontend programmer*, dan dalam proses pengembangannya, proyek ini juga melibatkan *product owner*, seorang *analyst*, serta dua *backend programmers*.



Gambar 3.14 Integrasi antar modul

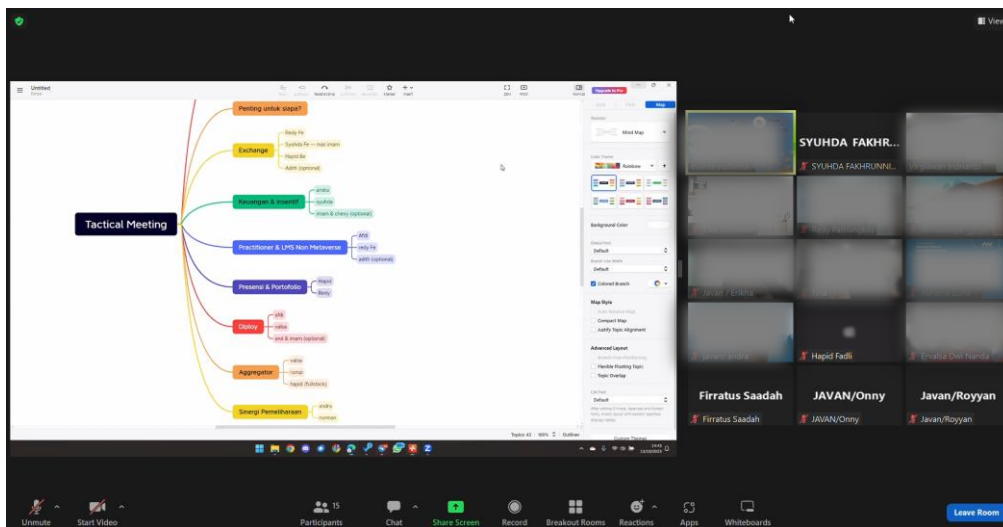
3.1.6 Membantu Mengembangkan Proyek C

Sekitar dua minggu setelah tergabung dalam proyek B, penulis dimasukkan ke dalam proyek C dan diminta untuk membantu mengerjakan beberapa *issue* serta meredesain beberapa komponen halaman. Proyek C merupakan proyek yang bertujuan untuk membantu salah satu instansi pemerintah dalam mendistribusikan informasi kepada seluruh masyarakat Indonesia. Informasi tersebut disampaikan melalui media *widget* yang dapat disematkan (*embed*) di *website*, serta API yang dapat diakses secara publik oleh peserta atau mitra yang bekerja sama dengan instansi tersebut. Proyek ini menggunakan React Typescript dalam pengembangannya. Dalam proyek C, penulis berperan sebagai *mentee (frontend) programmer*, dan proses pengembangannya melibatkan *product owner*, seorang *analyst*, satu *frontend programmer* lain, serta satu *backend programmer*.

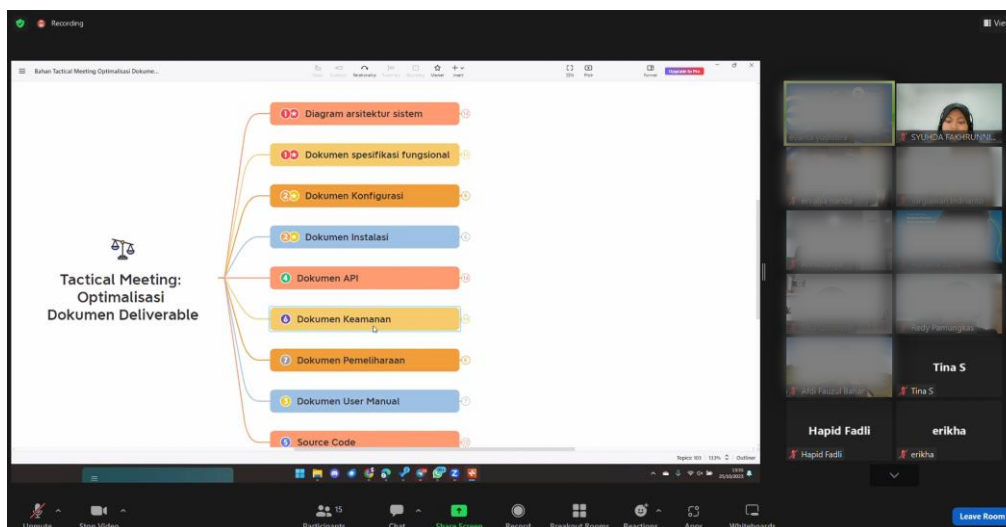
3.1.7 Tactical meeting

Selama pengembangan beberapa proyek, *tribe lead* secara rutin mengadakan *tactical meeting*. Berdasarkan (Kamal, 2022), *tactical meeting* ini berfungsi untuk membahas hal-hal praktikal, seperti perencanaan, kendala, serta tindakan yang perlu diambil, di mana setiap rencana dan kendala dapat menghasilkan tugas jangka pendek (*task*) maupun proyek jangka panjang. Rapat ini biasanya diadakan seminggu sekali dengan melibatkan semua anggota tim dalam lingkaran. Prosesnya cepat dan fokus, di mana setiap anggota melaporkan kemajuan, tantangan, serta hambatan yang dihadapi dalam pekerjaannya. Anggota juga bisa meminta

bantuan atau saran dari rekan lainnya. Tujuan utama rapat ini adalah untuk meningkatkan koordinasi, kolaborasi, dan efektivitas tim dalam bekerja. Seperti yang terlihat pada Gambar 3.15, *tactical meeting* tanggal 13 Oktober berfokus pada pemberdayaan anggota tim dalam satu *tribe* agar dapat saling membantu antar proyek sambil tetap menjaga keseimbangan jumlah *task* agar tidak *overload*. Selanjutnya, *tactical meeting* diadakan kembali pada tanggal 25 Oktober, seperti yang terlihat pada Gambar 3.16, untuk membahas optimalisasi dokumen-dokumen *deliverable* mengingat *timeline* yang sudah sangat ketat.



Gambar 3.15 *Tactical meeting* tribe lead Evan (13 Oktober 2023)

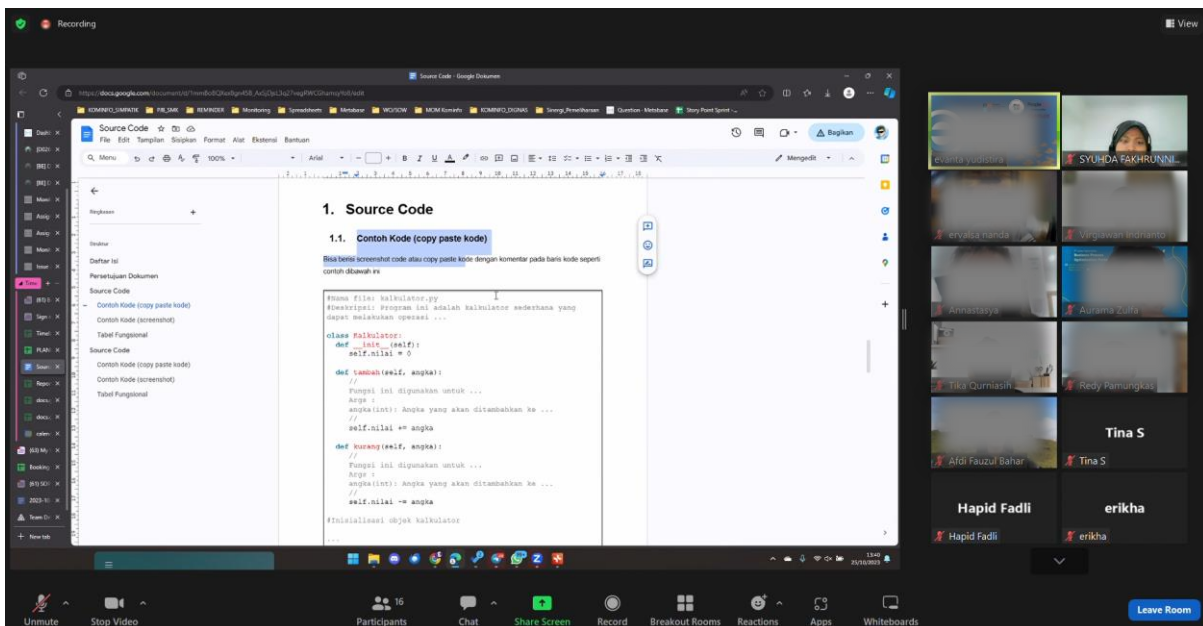


Gambar 3.16 *Tactical meeting* tribe lead Evan (25 Oktober 2023)

3.1.8 Mengerjakan Dokumen *Source Code* Proyek A, C, D, E, dan F

Setelah pelaksanaan *tactical meeting* mengenai dokumen *deliverable*, sejumlah rencana perubahan diharapkan dapat diimplementasikan untuk meningkatkan struktur pengerjaan

proyek dan memenuhi *timeline* yang ditentukan. Dalam pertemuan tersebut, penulis ditugaskan untuk mendokumentasikan sekitar sepuluh proyek yang berada di bawah naungan *circle*. Seperti yang terlihat pada Gambar 3.17, telah dipaparkan juga *template* dokumen *source code* yang harus dikerjakan. Keputusan ini diambil untuk mempercepat penyelesaian proyek, terutama karena saat itu, berdasarkan *dashboard monitoring*, jumlah *task* penulis terlihat paling sedikit dibandingkan *programmer* lainnya. Dengan demikian, penulis difokuskan pada dokumentasi *source code* dan dirilis dari proyek-proyek sebelumnya.



Gambar 3.17 Contoh *template* dokumen *source code*

Segera setelah *tactical meeting* dilaksanakan, salah satu *product owner* yang mengelola proyek D dan E meminta penulis untuk melakukan *setup* proyek di lokal dan mulai mengerjakan dokumen *source code*. Proyek D merupakan aplikasi presensi dan *monitoring* kehadiran peserta pelatihan *online*, sementara proyek E adalah aplikasi yang memungkinkan peserta pelatihan untuk mengunggah dan memamerkan hasil karya mereka, serta berinteraksi dengan peserta lain terkait karya tersebut. Pada minggu pertama, penulis fokus pada penyusunan dokumen *source code* untuk kedua proyek ini. Tak lama setelah itu, *product owner* lain meminta penulis mengerjakan dokumen *source code* proyek F, sebuah *tools* untuk mengetahui keanekaragaman berita, *posting-an*, atau komentar dari masyarakat terkait topik tertentu melalui *crawling*. Seiring berjalannya waktu, penulis juga diminta untuk mengerjakan dokumen *source code* dari proyek yang pernah ditangani sebelumnya, yaitu proyek A dan C.

3.1.9 Membantu Mengembangkan Proyek G

Selama kurang lebih satu bulan mengerjakan dokumen *source code*, penulis mengalami *overload tasks* karena banyaknya dokumen yang harus diselesaikan, sehingga beberapa *programmer* lain diminta untuk membantu. Mengingat tugas dokumentasi kurang sesuai dengan peran penulis sebagai *frontend programmer* yang seharusnya fokus pada *solving coding*, akhirnya penulis dipindahkan ke proyek G. Dalam proyek ini, penulis membantu menyelesaikan isu, memperbaiki *code smells* Sonar, mendesain ulang beberapa komponen, serta mengerjakan fitur hak akses. Proyek G yang menggunakan React dan Vue JS sebagai *framework frontend* ini bertujuan memberikan fasilitas tambahan dalam pelaksanaan pelatihan yang bertema *coding*, sekaligus menyediakan media bagi peserta untuk berlatih *coding* dasar tanpa perlu melakukan instalasi di laptop mereka. Tim pengembangan proyek G terdiri dari penulis sebagai *mentee (frontend) programmer*, seorang *product owner*, seorang *analyst*, satu *frontend programmer* lain, serta dua *backend programmer*.

3.1.10 Membantu Mengembangkan Proyek H

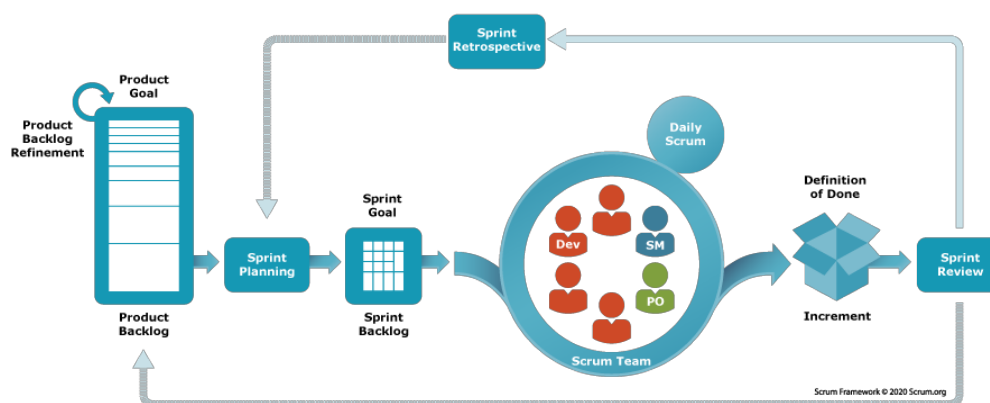
Seiring berkurangnya jumlah *tasks* penulis dan banyaknya *issue* di proyek G yang lebih berfokus pada sisi *backend*, penulis yang berada di bawah *circle lead* Evan ditugaskan untuk membantu proyek H. Meskipun masih berada di bawah *circle lead* Evan, penulis diperbantukan ke proyek H yang berada di bawah naungan *circle lead* Laras karena proyek tersebut membutuhkan tambahan *programmer* untuk membantu proses pengembangan. Proyek yang menggunakan React TypeScript sebagai *framework frontend* ini dinamakan Sistem Informasi Pengendalian Kasus. Di dalamnya terdapat modul-modul terkait pelaksanaan pelaporan kasus, investigasi, interogasi, dan pendakwaan yang dikelola oleh Bidang Data dan Informasi dari Lembaga XYZ dalam penindakan suatu kasus. Tim pengembangan proyek H terdiri dari penulis sebagai *mentee (frontend) programmer*, seorang *product owner*, empat *analyst*, dua *frontend programmer* lain, satu *fullstack developer*, serta enam *backend programmer*.

3.2 Manajemen Proyek

Kegiatan magang di PT Javan Cipta Solusi dilaksanakan dari hari Senin sampai Jumat, pukul 08.00 WIB hingga 17.00 WIB, dengan jeda istirahat siang selama satu jam pada pukul 12.00 WIB, kecuali hari Jumat yang memiliki jadwal istirahat lebih lama yaitu 1,5 jam. Kegiatan magang dilaksanakan secara *on site/work from office* (WFO). PT Javan Cipta Solusi menggunakan metode *Agile* dengan menerapkan *framework* Scrum dalam pengembangan

software. Setiap tim menerapkan metode *sprint* dalam kegiatan sehari-hari, yang dimulai dengan *sprint planning*. Dalam *sprint planning*, item-item dari *product backlog*, yang merupakan daftar lengkap semua fitur dan tugas yang harus dikerjakan, dipilih untuk dimasukkan ke dalam *sprint backlog*, yaitu daftar pekerjaan yang akan dikerjakan selama *sprint* berlangsung. Biasanya, satu *sprint* ini berlangsung selama dua minggu.

Setelah *sprint* berlangsung, tim mengadakan *daily scrum* selama 10 hingga 15 menit setiap hari untuk menyampaikan perkembangan pekerjaan serta kendala yang dihadapi. Setelah *sprint* selesai, dilakukan *sprint review* untuk mengevaluasi hasil pekerjaan, dilanjutkan dengan *sprint retrospective* yang berfokus pada peningkatan proses kerja tim ke depannya. Dalam retrospektif, tim merefleksikan apa yang sudah berjalan baik dan apa yang bisa diperbaiki agar *sprint* berikutnya lebih efisien. Dengan siklus ini, setiap *task* dan lama pengerjaannya direncanakan dengan baik sejak awal, sehingga seluruh tim dapat fokus dalam mengejar target yang telah ditetapkan. Untuk detail alur lengkap terkait tahapan aktivitas dalam *scrum* dapat dilihat pada Gambar 3.18.

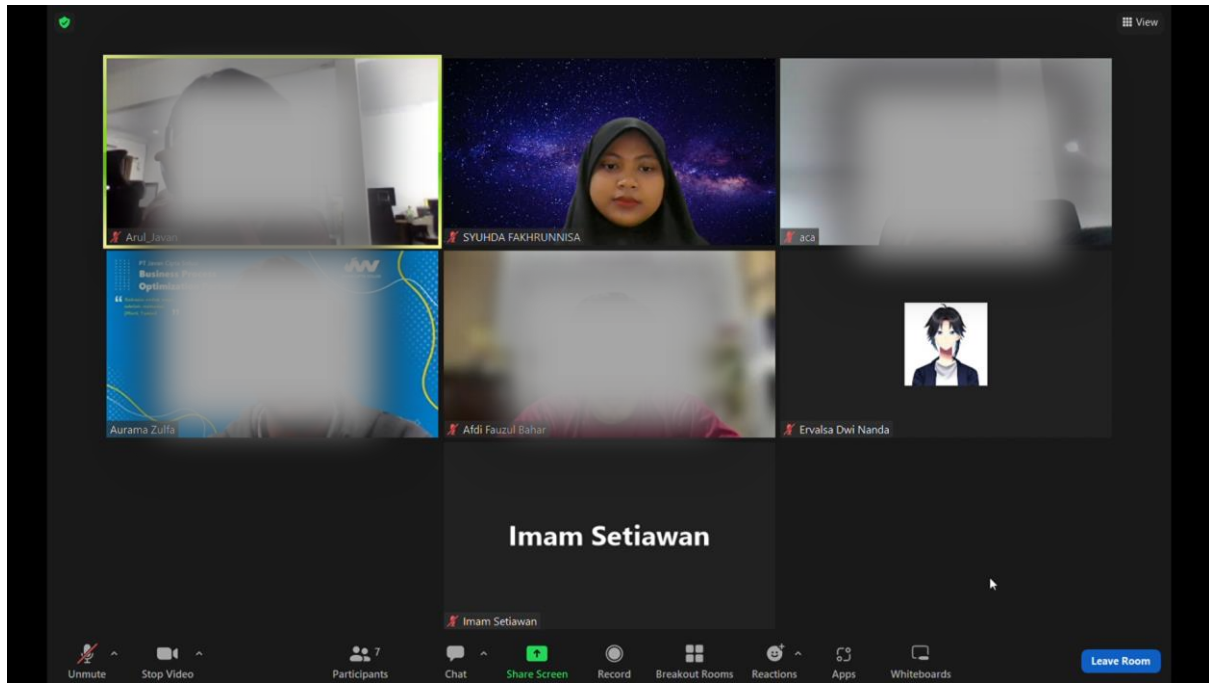


Gambar 3.18 Tahapan aktivitas dalam *scrum framework*

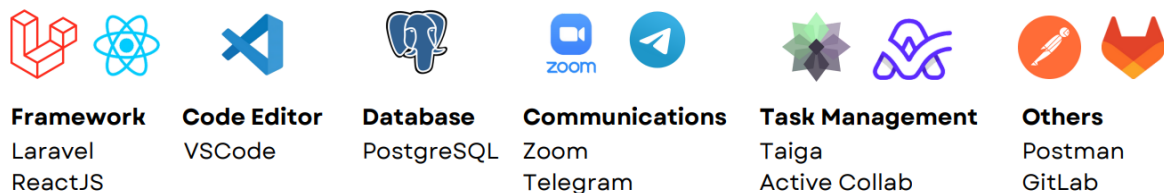
Sumber: (Scrum.org, 2020)

PT Javan Cipta Solusi memiliki struktur organisasi yang terorganisir dengan sangat baik. Penulis tergabung dalam *circle* tim operasional yakni sebagai *programmer*, di bawah pengawasan langsung oleh Evanta Yudhistira sebagai *tribe lead* dan Dicky Puja Pratama sebagai *operational lead*. Tim ini terdiri dari beberapa posisi, di antaranya *operational lead*, *tribe lead*, *product owner*, *system analyst*, dan *programmer*. Sebagai programmer, tugas harian yang dilakukan mencakup pengecekan *task-task* yang diberikan melalui platform Taiga, serta menyelesaikan setiap *task* sesuai *due date* yang telah ditetapkan. Selain itu, selama masa *sprint*, tim proyek melakukan *daily scrum*, sebagaimana terlihat pada Gambar 3.19, untuk memastikan perkembangan *task* harian, mencakup apa yang sudah dikerjakan kemarin, rencana pekerjaan

hari ini, dan identifikasi kendala dari berbagai aspek yang dihadapi selama pengerjaan *task*. Selain itu, selama proses magang, terdapat beberapa *tools* dan teknologi yang perlu disiapkan. Gambar 3.20 menampilkan *tools* dan teknologi yang akan digunakan dalam keseharian proyek untuk mendukung pekerjaan *programmer* serta tim operasional lainnya.



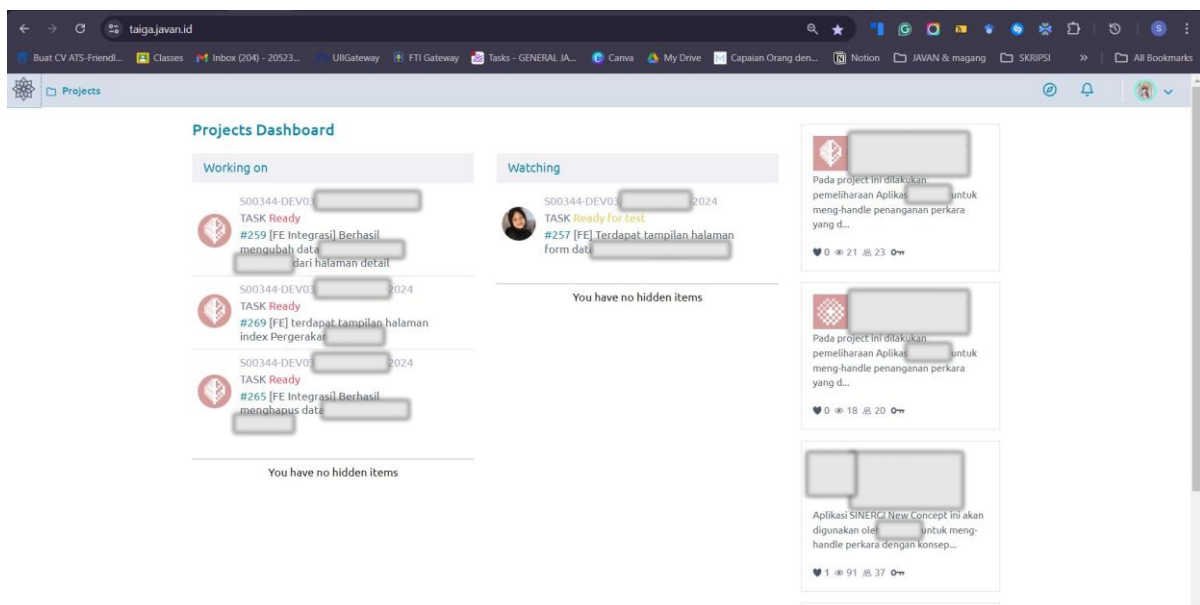
Gambar 3.19 *Daily scrum*



Gambar 3.20 *Tools* dan teknologi yang digunakan di PT Javan Cipta Solusi

Salah satu *tools* yang digunakan oleh PT Javan Cipta Solusi untuk manajemen *task* adalah Taiga. *Tool* ini berfungsi mengelola sebuah proyek agar dapat di-*monitoring* dengan mudah, baik dari segi capaian maupun *progress*, untuk memastikan proyek selesai tepat waktu. Gambar 3.21 menampilkan *task* yang di-*assign* akan muncul di *dashboard* Taiga dengan status awal 'Ready'. Ketika *programmer* mulai mengerjakan *task* tersebut, status harus diubah menjadi 'In Progress', dan setelah selesai, diubah menjadi 'Ready for Test' (RFT), lalu *assignee* diubah ke *analyst* terkait. Namun, jika *programmer* berperan sebagai *mentee*, *code* yang dikerjakan harus di-*review* terlebih dahulu oleh mentor atau *senior programmer* yang

ditentukan oleh *product owner*. Sebelum status diubah ke RFT, *mentee* perlu mengubah status *task* menjadi ‘Code Review’ dan mentor akan mengubah status *task* ke RFT setelah proses *code review* selesai. Setelah itu, *analyst* akan melakukan *testing* dan mengubah status *task* menjadi ‘Testing’. Jika *task* telah memenuhi *definition of done* (DoD) atau *acceptance criteria*, status *task* akan diubah menjadi ‘Closed’. Namun, jika ditemukan hal yang belum sesuai, *analyst* akan mengembalikan *task* dengan status ‘Feedback’, dan *programmer* harus memperbaiki *task* tersebut sebelum mengajukan RFT kembali. Untuk detail lengkap mengenai *task life cycle*, dapat dilihat pada Gambar 3.22.



Gambar 3.21 Dashboard Taiga

Task Life Cycle

Biasakan Task Selesai Sebelum Due , Lebih Cepat Lebih Baik

Task Assigned : 22 April 2024

Due Date : 24 April 2024



1. Programmer **Melakukan validasi** terhadap task yang di assign
2. **Menkonfirmasi atau Memberi Feedback** kepada analyst jika kurang jelas
3. **Berkonsultasi** ke techlead jika task belum di pahami secara teknis
4. **Mengerjakan Task** sesuai dengan spesifikasi yang di tentukan
5. **Memvalidasi pengerjaan** task dan memastikan sesuai DOD
6. **Berkonsultasi** Dengan Techlead Jika task terasa susah di tengah jalan pengerjaan agar selalu bisa achieve

1. Analyst **Melakukan Testing Pekerjaan** Programmer
2. Analyst **Memberikan feedback**, ketika hasil dari programmer tidak sesuai dengan ketentuan DOD yang di Sepakati
3. Programmer **Mengkonfirmasi Feedback** yang di berikan oleh analyst
4. Programmer **Mengerjakan Feedback** dan memastikan ulang jika task sudah sesuai dengan feedback yang di kasih
5. Programmer berkonsultasi jika ada kesulitan techlead
6. Analyst **Melakukan Testing** Kembali Sampai Feedback selesai

Gambar 3.22 Task life cycle

3.3 Pengembangan Proyek A

3.3.1 Pendefinisian Proyek

Di era digital yang terus berkembang, Proyek A dirancang untuk menjembatani kesenjangan antara generasi muda Indonesia yang mencari pekerjaan yang sesuai dengan keterampilan mereka dan perusahaan yang membutuhkan karyawan potensial. Dengan sistem ini, masyarakat muda dapat memanfaatkan platform yang mengintegrasikan profil dan portofolio mereka, membantu mereka menonjolkan keahlian yang unik, serta mengakses peluang pekerjaan yang relevan. Di sisi lain, perusahaan dapat menikmati proses rekrutmen yang lebih efisien dan selektif, sehingga memudahkan pemilihan karyawan yang sesuai dengan kebutuhan mereka. Melalui integrasi teknologi, sistem informasi ini bertujuan mendukung pertumbuhan ekonomi dan kemajuan karier bagi generasi muda Indonesia.

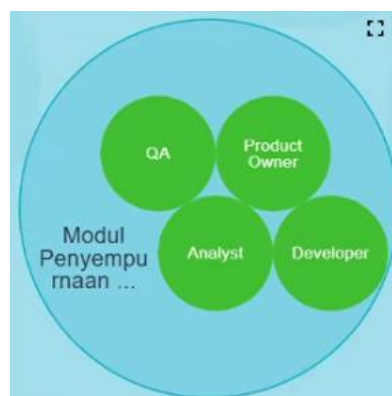
Tujuan Proyek A adalah sebagai berikut:

1. Memudahkan pencari kerja menemukan pekerjaan sesuai dengan keterampilan mereka.
2. Membantu perusahaan dalam mengefisienkan proses rekrutmen karyawan.
3. Menciptakan kesesuaian yang lebih baik antara pencari kerja dan perusahaan guna mengoptimalkan kepuasan dan produktivitas kedua belah pihak.

3.3.2 Inisialisasi Proyek

Peran

Pengembangan sebuah proyek perangkat lunak memerlukan berbagai peran dan teknologi untuk memastikan keberhasilan proyek. Beberapa peran dan teknologi diterapkan dalam pengembangan Proyek A. Selama magang, penulis tergabung dalam tim sebagai *mentee programmer*. Adapun peran-peran dari *tribe member* beserta akuntabilitasnya dalam Proyek A dapat dilihat pada Gambar 3.23.



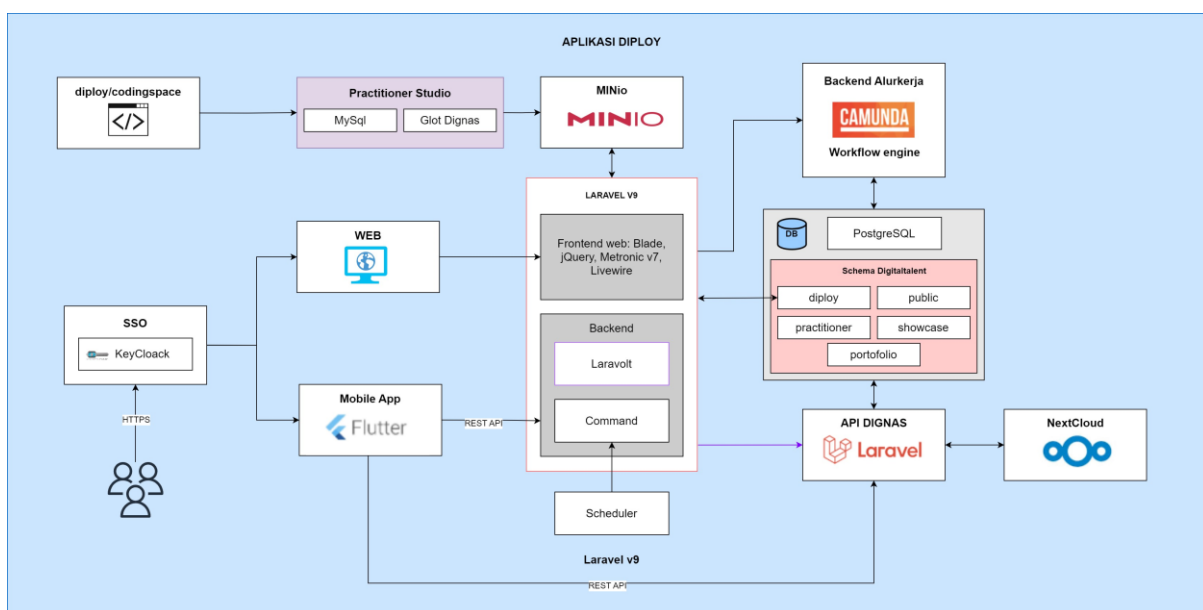
Gambar 3.23 Roles yang terlibat dalam proyek A

1. *Tribe lead*, bertanggung jawab sebagai pemimpin tim, dengan tugas utama meliputi koordinasi antar tim, fasilitasi pertemuan, serta *mentoring* dan pengembangan anggota tim. Selain itu, peran ini mencakup manajemen risiko dan hambatan, pengembangan budaya kerja yang positif, pemantauan kapasitas serta beban kerja tim, dan penyelesaian konflik. *Tribe lead* juga berperan dalam melakukan evaluasi sprint, pemecahan masalah teknis, serta memahami aspek teknis dari produk yang sedang dikembangkan.
2. *Product owner*, bertanggung jawab sebagai *leader* yang mengarahkan *Project Squad* untuk mencapai GOALS proyek. Peran ini meliputi memastikan agar para *programmer* tetap produktif dengan selalu menyediakan *task* yang tersedia dan menetapkan prioritas *task*-nya. *Product owner* juga menyusun dokumen perencanaan *sprint*, menjadwalkan *sprint planning* dan *sprint review*, serta membantu memastikan proses *testing* dilakukan dengan baik oleh analis. Selain itu, tugasnya termasuk *me-review* dokumen *deliverable* dan memiliki akuntabilitas terhadap perencanaan *user story*.
3. *Analyst*, bertanggung jawab untuk melakukan fase *shaping* dan analisis sesuai dengan arahan *product owner*. Tugas lainnya termasuk membuat *task* sesuai dengan standar yang telah ditetapkan, melakukan *testing* berdasarkan *acceptance criteria* yang diberikan, serta menyusun *Minutes of Meeting* (MoM) dan membuat *invitation meeting*. *Analyst* juga terlibat dalam pembuatan dan revisi dokumen *deliverable*, membuat *task* pada *Issue Sentry* dan QA Touch, serta membuat *history log issue*. Selain itu, *analyst* harus menjaga angka 'Ready For Test' harian agar tidak lebih dari 24 Jam.
4. *Developer* atau *programmer*, bertanggung jawab menyelesaikan *task* sesuai dengan tenggat waktu dan *acceptance criteria* yang telah ditentukan. Mereka terlibat aktif dalam perencanaan *sprint* dan *review sprint* untuk memberikan masukan terkait estimasi waktu serta tingkat kompleksitas pengembangan. Selain itu, *developer* turut berpartisipasi dalam pertemuan retrospektif *sprint*, memberikan umpan balik untuk meningkatkan efisiensi tim. Tanggung jawab lainnya termasuk mengidentifikasi, melaporkan, serta memperbaiki *bug* atau masalah teknis yang muncul selama proses pengembangan.

5. *Quality Assurance* (QA), bertanggung jawab merancang dan mengembangkan strategi pengujian. Tugas lainnya mencakup membuat dan memelihara skrip pengujian, melakukan pengujian fungsional dan non-fungsional, serta melakukan pengujian regresi. QA juga mendokumentasikan hasil pengujian, berkolaborasi dengan tim pengembang untuk mengidentifikasi, melaporkan, dan menyelesaikan *bug* dengan cepat dan efisien, serta memastikan bahwa setiap fitur mematuhi *acceptance criteria*. Selain itu, QA bertugas melaporkan dan menganalisis *bug*, membuat serta memelihara dokumen pengujian, serta melakukan *review* dan validasi hasil pengujian.

Teknologi

Pengembangan Proyek A menggunakan berbagai teknologi yang terintegrasi dalam satu sistem. Seperti yang terlihat pada Gambar 3.24, arsitektur sistem terdiri dari beberapa komponen utama. Di sisi *backend*, proyek menggunakan Laravel v9 sebagai *framework* utama. Tampilan *frontend* web dikembangkan menggunakan Blade sebagai *template engine*, didukung dengan jQuery dan Metronic v7 untuk meningkatkan interaktivitas antarmuka, serta Livewire. Untuk manajemen basis data, sistem menggunakan PostgreSQL. Sistem juga dilengkapi dengan Camunda sebagai *workflow engine* di sisi *back-end*. Sistem keamanan ditangani melalui SSO dengan KeyCloak, dan untuk penyimpanan *cloud* menggunakan NextCloud. Selain aplikasi web, sistem juga menyediakan aplikasi *mobile* yang dikembangkan menggunakan Flutter yang terhubung melalui REST API.



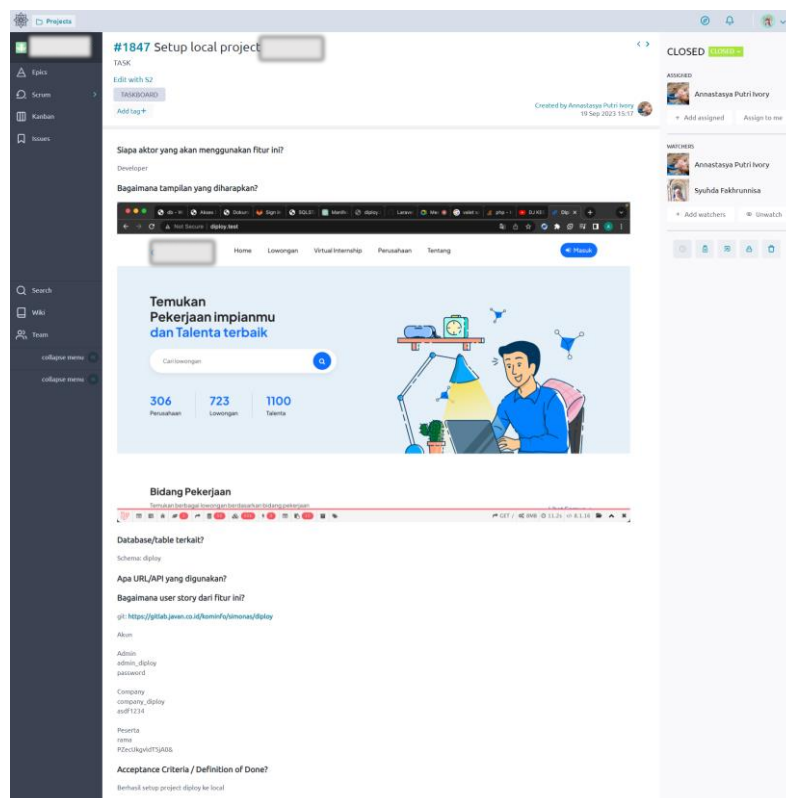
Gambar 3.24 Diagram arsitektur sistem Proyek A

3.3.3 Perencanaan dan Pelaksanaan Proyek

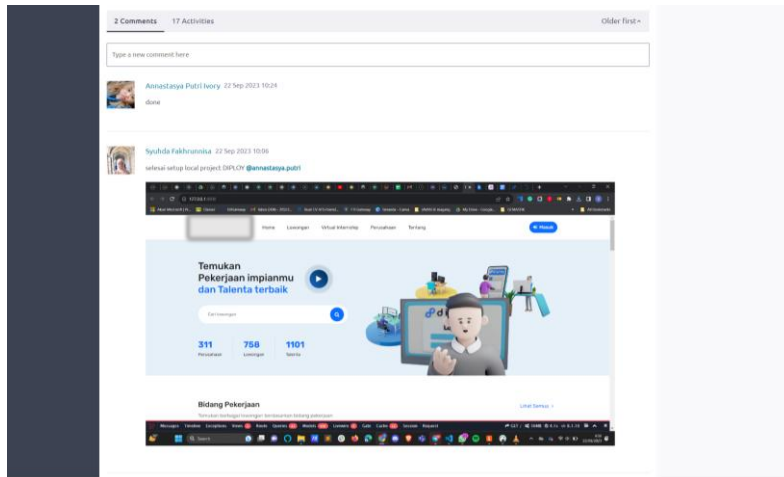
Ketika penulis bergabung, proyek sudah berada di tahap akhir *development*, sehingga keterlibatan penulis lebih berfokus pada fase *development* akhir. Pada fase ini, sebagian besar fitur utama telah dikembangkan, dan tim mulai beralih pada *bug fixing* serta penyempurnaan desain UI/UX berdasarkan *feedback* dari *stakeholder*. Penulis diberikan tugas untuk memperbaiki *issue* terkait tampilan *frontend*, menyempurnakan interaksi antar-komponen, serta melakukan beberapa penyesuaian pada elemen desain yang mengalami perubahan setelah *review sprint*. *Milestone* pertama untuk penyelesaian proyek ini ditargetkan di akhir tahun 2023. Namun, karena terdapat prioritas modul lain yang dianggap lebih *urgent*, pengembangan modul ini ditunda sementara waktu dan fokus tim beralih ke penyelesaian modul-modul tersebut. Selama periode ini, penulis berkontribusi dalam beberapa tugas, antara lain:

Melakukan Setup Proyek di *Local*

Sebelum diberikan tugas lebih lanjut, penulis menerima tugas awal untuk melakukan *setup* proyek di *local*. Seperti yang terlihat pada Gambar 3.25 dan Gambar 3.26 penulis ditugaskan untuk memastikan proyek dapat berjalan dengan baik di *local development environment*. Proses ini merupakan tahap penting yang harus diselesaikan sebelum memulai pengerjaan *task-task* berikutnya.



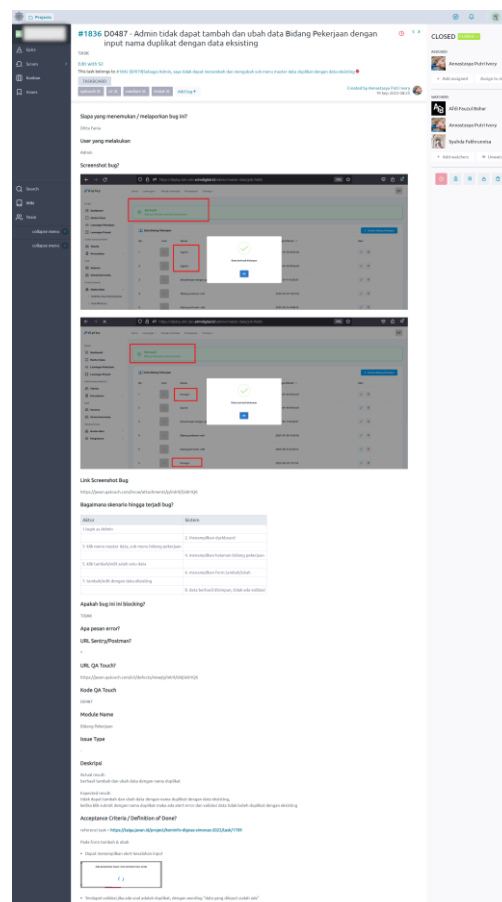
Gambar 3.25 *Task* dan detail instruksi *setup* proyek A di Taiga



Gambar 3.26 Dokumentasi hasil *setup* proyek A di *local environment*

Mengimplementasikan validasi input

Setelah berhasil melakukan *setup* proyek, penulis mendapat tugas untuk mengimplementasikan sistem validasi pada *form*. Seperti yang terlihat pada Gambar 3.27, penulis ditugaskan untuk menambahkan fitur validasi yang dapat mendeteksi dan memberikan peringatan ketika terdapat data duplikat pada saat pengisian formulir.



Gambar 3.27 Detail *task* implementasi validasi data duplikat di Taiga

Gambar 3.28 menunjukkan hasil implementasi validasi berupa *alert* yang muncul ketika sistem mendeteksi data yang sama telah terdaftar sebelumnya, memastikan tidak ada duplikasi data dalam sistem.

Syuhda Fakhrunnisa 25 Sep 2023 08:54
MR: https://gitlab.javan.co.id/kominfo/simonas/diploy/-/merge_requests/259
Mention: @afdi

SCREENSHOTS

The image displays three screenshots of a web application interface for 'diploy'. Each screenshot shows a form with a red error message and a modal alert box. The first screenshot shows the 'Edit Bidang Pekerjaan' form with a red error message 'Data yang diinput sudah ada' and a modal alert 'Ada kesalahan input. Cek kembali isian anda'. The second screenshot shows the 'Tambah Bidang Pekerjaan' form with a red error message 'Nama tidak boleh kosong' and a modal alert 'Ada kesalahan input. Cek kembali isian anda'. The third screenshot shows the 'Tambah Bidang Pekerjaan' form with a red error message 'Data yang diinput sudah ada' and a modal alert 'Ada kesalahan input. Cek kembali isian anda'.

Gambar 3.28 Hasil implementasi validasi data duplikat pada *form*

Selanjutnya, penulis juga mengimplementasikan validasi pembatasan karakter pada formulir *Virtual Internship*. Seperti yang terlihat pada Gambar 3.29, sistem harus dapat menampilkan *sweet alert* dan pesan *error* ketika pengguna memasukkan nama pekerjaan yang melebihi 100 karakter.

#1862 D0501 - [Virtual Intership] Terdapat sweetalert dan pesan error validasi saat menginput nama pekerjaan lebih dari 100 karakter saat membuat data baru

TASK
 Edit with S2
 This task belongs to #1896 [SP07] Sebagai perusahaan, saya dapat melihat validasi saat melakukan tambah data virtual intership.

Created by Annastasya Putri Ivory
 24 Sep 2023 17:28

Screenshot bug?

Link Screenshot Bug
<https://jean.qatouch.com/issue/attachments/2/mi/9/PCK32d>

Bagaimana skenario hingga terjadi bug?

Aktor	Sistem
1. login as perusahaan	2. Menampilkan dashboard
3. klik menu virtual intership, sub menu daftar virtual intership	4. menampilkan halaman virtual intership
5. klik tambah	6. menampilkan form tambah virtual intership
7. tambah dengan nama pekerjaan lebih dari 100 karakter, klik simpan	8. tidak menampilkan alert kesalahan input dan validasi

Apakah bug ini blocking?
 TIDAK

Endpoint URL
<https://dplay.sdm.dev.sdmidigital.id/company/intership/virtual-intership/create>

URL QA Touch?
<https://jean.qatouch.com/v2/defects/view/mi/9/PCK32d>

Kode QA Touch
 D0501

Module Name
 Daftar Virtual Intership

Issue Type
 -

Deskripsi
 Actual:
 Dapat menyimpan data dengan lebih 100 karakter
 Expected:
 Tidak dapat menyimpan data dengan lebih dari 100 karakter lalu muncul sweetalert dan pesan error validasi

Acceptance Criteria / Definition of Done?
 Pada form tambah,
 • Nilai dari field "Nama Pekerjaan" tidak boleh lebih dari 100 karakter
 • Jika memasukkan isian lebih dari 100 karakter, dapat menampilkan =
 = alert kesalahan input.

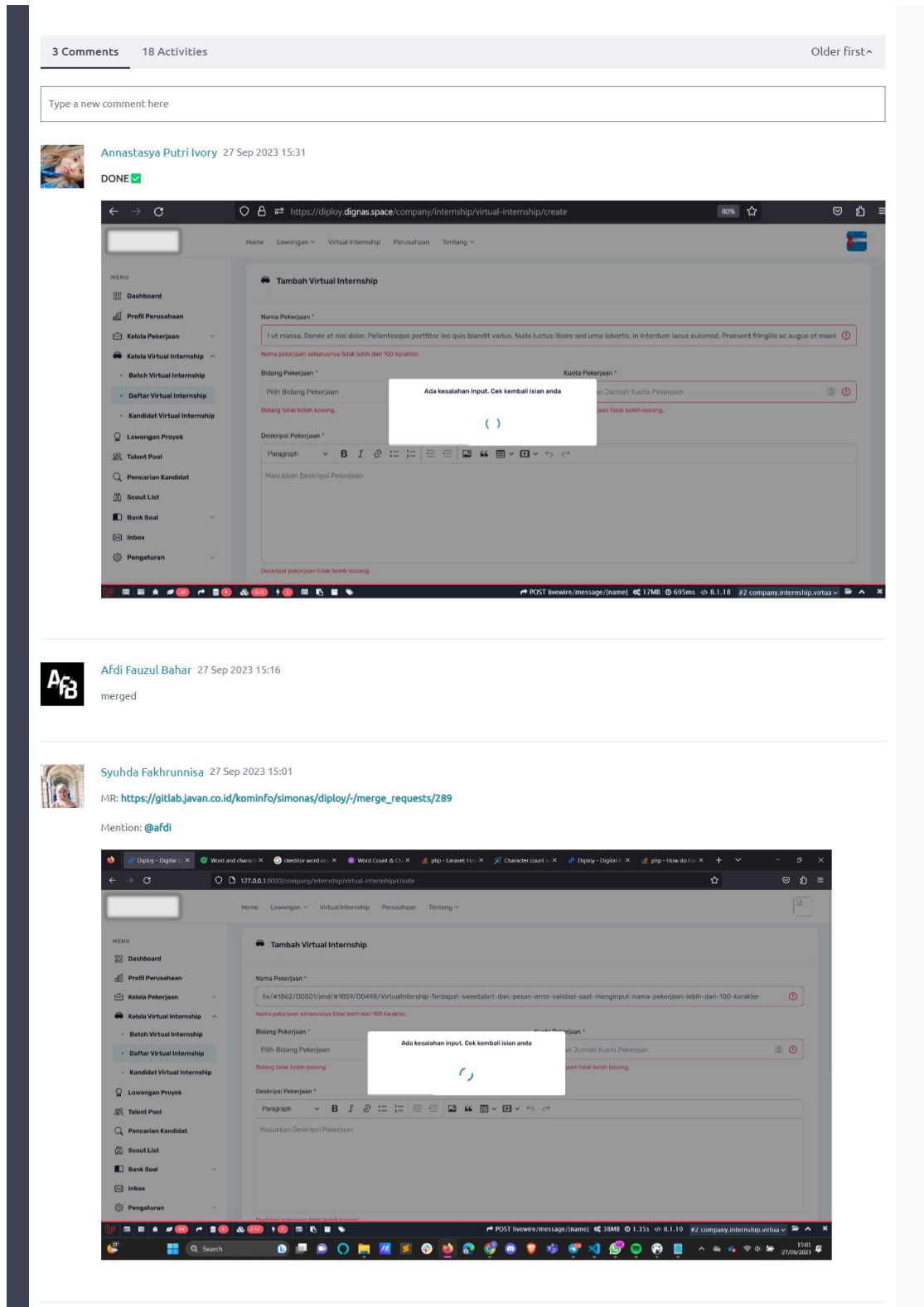
Ada kesalahan input. Cek kembali isian anda

Validasi form field dengan wording "Nama pekerjaan seharusnya tidak lebih dari 100 karakter"

Nama Kategori *
 Pelentesque porttitor leo quis blandit varius. Nu
 Nama kategori seharusnya tidak lebih dari 100 karakter

Gambar 3.29 Detail *task* implementasi validasi batasan karakter di Taiga

Gambar 3.30 memperlihatkan hasil implementasi berupa notifikasi dan pesan validasi yang muncul saat pengguna mencoba memasukkan teks yang melebihi batas karakter yang ditentukan.



Gambar 3.30 Hasil implementasi pesan validasi untuk pembatasan karakter pada formulir

Menambahkan *count text* pada *input*

Dalam pengembangan sistem validasi *input*, penulis mengerjakan beberapa fitur validasi untuk memastikan data yang dimasukkan sesuai dengan ketentuan. Salah satunya adalah implementasi penghitung karakter pada kolom deskripsi. Seperti yang terlihat pada Gambar 3.31, penulis ditugaskan untuk menambahkan fitur *count text* yang menampilkan jumlah karakter pada kolom deskripsi pekerjaan.

#1863 D0502 - [Virtual Internship] Terdapat count text pada inputan deskripsi pekerjaan pada saat menambah data baru

TASK
 Edit with S2
 This task belongs to #1896 [D017] sebagai perusahaan, saya dapat melihat validasi saat melakukan tambah data virtual internship

Slapa yang menemukan / melaporkan bug ini?
 Cahli Bagus Prakasa

User yang melakukan
 Perusahaan

Screenshot bug?

Link Screenshot Bug
<https://www.qatouch.com/issue/attachments/ymh9/5/61mm>

Bagaimana skenario hingga terjadi bug?

Aktor	Sistem
1. login as Perusahaan	2. Menampilkan dashboard
3. klik menu kelola VI, sub menu daftar VI	4. menampilkan halaman virtual internship
5. klik tambah salah satu data	6. menampilkan form tambah/ubah salah satu data
7. tambah field deskripsi pekerjaan	8. tidak menampilkan count text

Apakah bug ini ini blocking?
 TIDAK

Endpoint URL
<https://dploy.adm.dev.admidigital.id/company/internship/virtual-internship/create>

URL QA Touch?
<https://www.qatouch.com/v2/defects/view/ymh9/5/61mm>

Kode QA Touch
 D0502

Module Name
 Daftar Virtual Internship

Issue Type
 -

Deskripsi
 Actual:
 Tidak terdapat count text pada inputan deskripsi pekerjaan
 Expected:
 Terdapat count text pada inputan deskripsi pekerjaan, Contoh : di halaman kelola pekerjaan > daftar lowongan > klik tambah / ubah <https://print.sq.booq.com/rz>

Acceptance Criteria / Definition of Done?
 Pinda form tambah, dapat menampilkan count text dibawah form field "Deskripsi Pekerjaan" seperti dibawah ini

Deskripsi Pekerjaan *

total jumlah teks seperti di halaman ini

Jumlah Teks: 40

Gambar 3.31 Detail *task* implementasi penghitung karakter di Taiga

Gambar 3.32 menunjukkan hasil implementasi di mana sistem berhasil menghitung dan menampilkan jumlah karakter yang telah dimasukkan oleh pengguna.

3 Comments 18 Activities Older first

Type a new comment here

Annastasya Putri Ivory 29 Sep 2023 12:37
 DONE ✓
 Pada form tambah, dapat menampilkan count text dibawah form field "Deskripsi Pekerjaan" ✓

https://diplay.sdm.dev.sdmdigital.id/company/internship/virtual-internship/create

Home Lowongan Virtual Internship Perusahaan Tentang

MENU

- Dashboard
- Profil Perusahaan
- Kelola Pekerjaan
 - Kelola Virtual Internship
 - Batch Virtual Internship
 - Daftar Virtual Internship
 - Kandidat Virtual Internship
- Lowongan Proyek
- Talent Pool
- Pencarian Kandidat
- Scout List
- Bank Soal
- Inbox
- Pengaturan

Pilih Bidang Pekerjaan Masukkan Jumlah Kuota Pekerjaan

Deskripsi Pekerjaan *

Paragraph **B I**

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed lacus consectetur ex eget pellentesque. Aliquam erat volutpat. Suspendisse ut nequeis erat. Sed scelerisque lacus volutpat egesta portitor. Zhusellae ac sapien condimentum. lacus sed to facilis ea. Vestibulum etiam sodales fermentum. Etiam inpediet lacinia nec ultrices condimentum. Ut dui nisi, semper ut nisi sed. tristique semper mi. Mauris vulpute, orci a nequeis tristique. tibus orci tempor massa, non vulputate leo orci at erat. Aliquam et amet tibus interdum dictum leo id, auctor massa. Integer nec sollicitudin tortor. Curabitur diam elit, ullamcorper vitae imperdiet vitae, euismod ut massa. Donec et nisi dolor. Pellentesque porttitor leo quis blandit varius. Nulla luctus libero sed urna lobortis. In interdum lacus euismod. Praesent fringilla ac augue et maximus.

Jumlah Teks: 851
Deskripsi Pekerjaan harus minimal 500 karakter

Keterampilan *

Pilih Keterampilan

Tool yang digunakan *

Pilih Tool

Link Video Pembuka

Messages Timeline Exceptions Views Route Queries Models Livewire Gate Cache Session Request
 GET company/internship/virtual-internship/create 38MB 2.06s 8.1.20

Afidi Fauzul Bahar 29 Sep 2023 11:19
 merged

Syuhda Fakhrunnisa 27 Sep 2023 16:36
 MR: https://gitlab.javan.co.id/kominfo/simonas/diplay/-/merge_requests/294
 Mention: @afidi

Home Lowongan Virtual Internship Perusahaan Tentang

MENU

- Dashboard
- Profil Perusahaan
- Kelola Pekerjaan
 - Kelola Virtual Internship
 - Batch Virtual Internship
 - Daftar Virtual Internship
 - Kandidat Virtual Internship
- Lowongan Proyek
- Talent Pool
- Pencarian Kandidat
- Scout List
- Bank Soal
- Inbox
- Pengaturan

Tambah Virtual Internship

Nama Pekerjaan *

Masukkan Nama Pekerjaan

Bidang Pekerjaan * Kuota Pekerjaan *

Pilih Bidang Pekerjaan Masukkan Jumlah Kuota Pekerjaan

Deskripsi Pekerjaan *

Paragraph **B I**

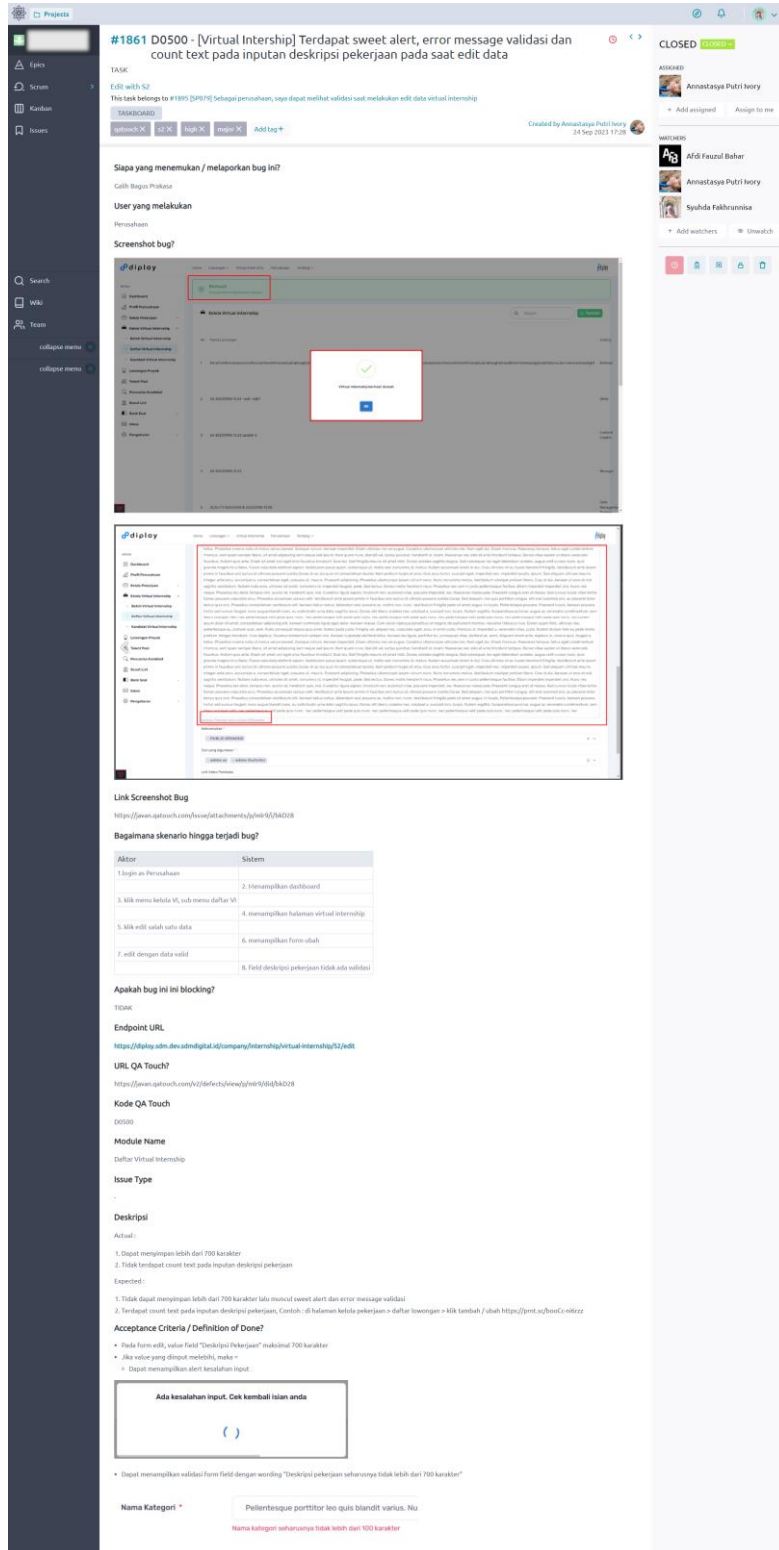
What is Lorem Ipsum?
 Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.

Jumlah Teks: 594
Deskripsi Pekerjaan harus minimal 500 karakter

38MB 4.18s 8.1.10

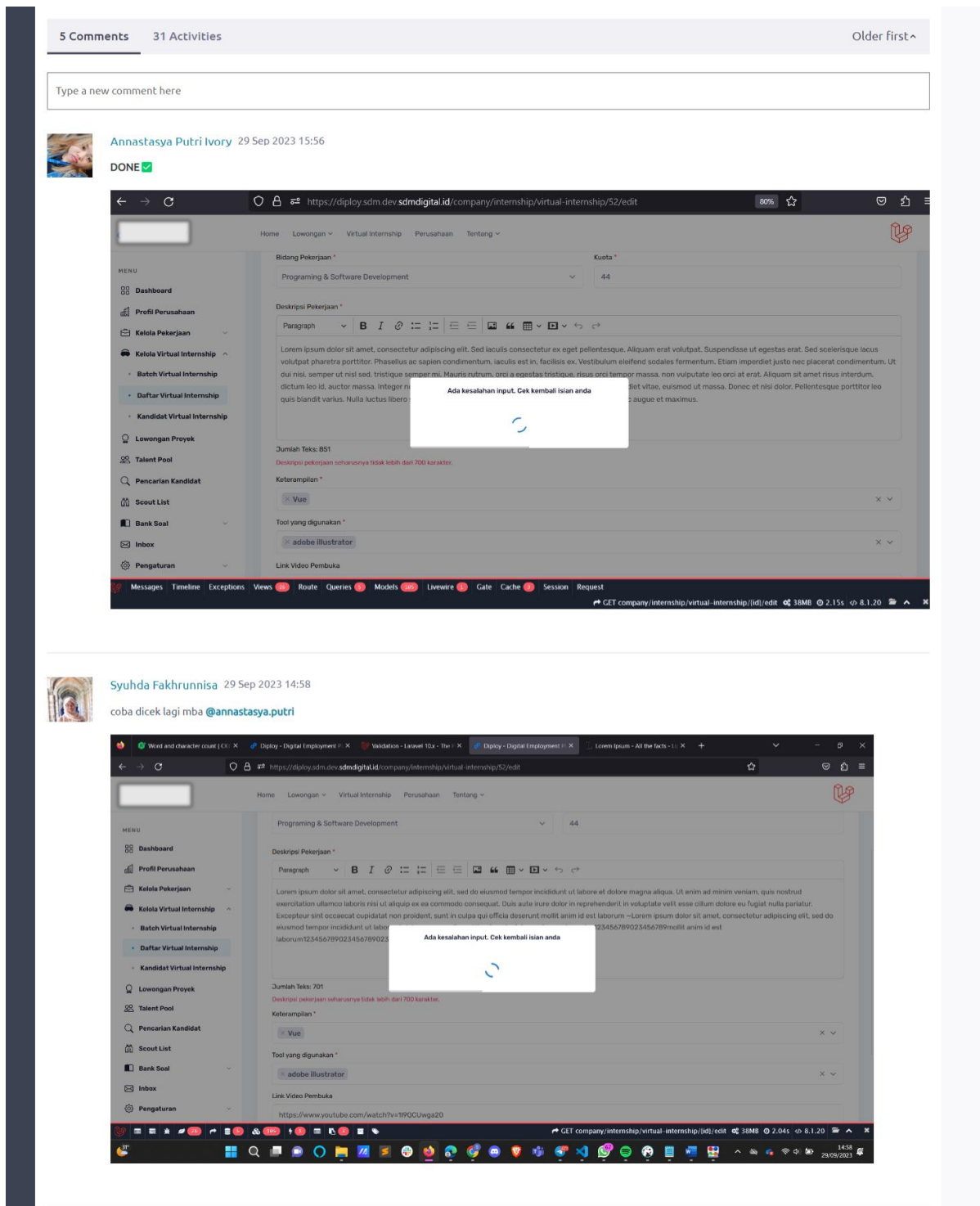
Gambar 3.32 Hasil implementasi penghitungan karakter pada kolom deskripsi pekerjaan

Selain itu, penulis juga diberikan *task* lain untuk mengintegrasikan beberapa fitur validasi sekaligus. Seperti yang terlihat pada Gambar 3.33, penulis ditugaskan untuk mengimplementasikan kombinasi *sweet alert*, *error message*, dan penghitung karakter dalam satu formulir.



Gambar 3.33 Detail *task* integrasi fitur validasi dan penghitung karakter pada formulir

Gambar 3.34 memperlihatkan hasil implementasi berupa kombinasi ketiga fitur tersebut, memberikan pengalaman pengguna yang lebih informatif saat mengisi formulir.



Gambar 3.34 Hasil implementasi sistem validasi dan penghitung karakter pada formulir

Sorting Data

Penulis mendapat *task* untuk mengimplementasikan fitur pengurutan data pada halaman lowongan pekerjaan. Seperti yang terlihat pada Gambar 3.35, penulis ditugaskan untuk menampilkan daftar lowongan pekerjaan secara terurut (*default*) dari yang terbaru.

#1906 D0534 - Peserta/kandidat dapat melihat lowongan pekerjaan terurut default dari yang terbaru

TASK

Edit with S2

This task belongs to #1824 [SP014] Sebagai peserta, saya dapat melihat lowongan pekerjaan

TASKBOARD

qatouch X S2 X high X minor X Add tag+

Created by Annastasya Putri Ivory
26 Sep 2023 10:17

Siapa yang menemukan / melaporkan bug ini?
Ditta Fania

User yang melakukan
Peserta

Screenshot bug?

Link Screenshot Bug
<https://javan.qatouch.com/issue/attachments/p/mlr9/f/1rnK5>

Bagaimana skenario hingga terjadi bug?

Aktor	Sistem
1. login as Peserta	
	2. Menampilkan dashboard
	3. Menampilkan lowongan pekerjaan tidak dari yang terbaru

Apakah bug ini blocking?
TIDAK

Endpoint URL
<https://diplay.sdm.dev.sdmdigital.id/user/dashboard>

URL QA Touch?
<https://javan.qatouch.com/v2/defects/view/p/mlr9/did/1rnK5>

Kode QA Touch
D0534

Module Name
Dashboard

Issue Type
-

Deskripsi

Actual result:
data tidak diurutkan default dari yang terbaru padahal tertulis 'Lowongan terbaru dari perusahaan'

Expected result:
lowongan terurut dari yang terbaru

Acceptance Criteria / Definition of Done?

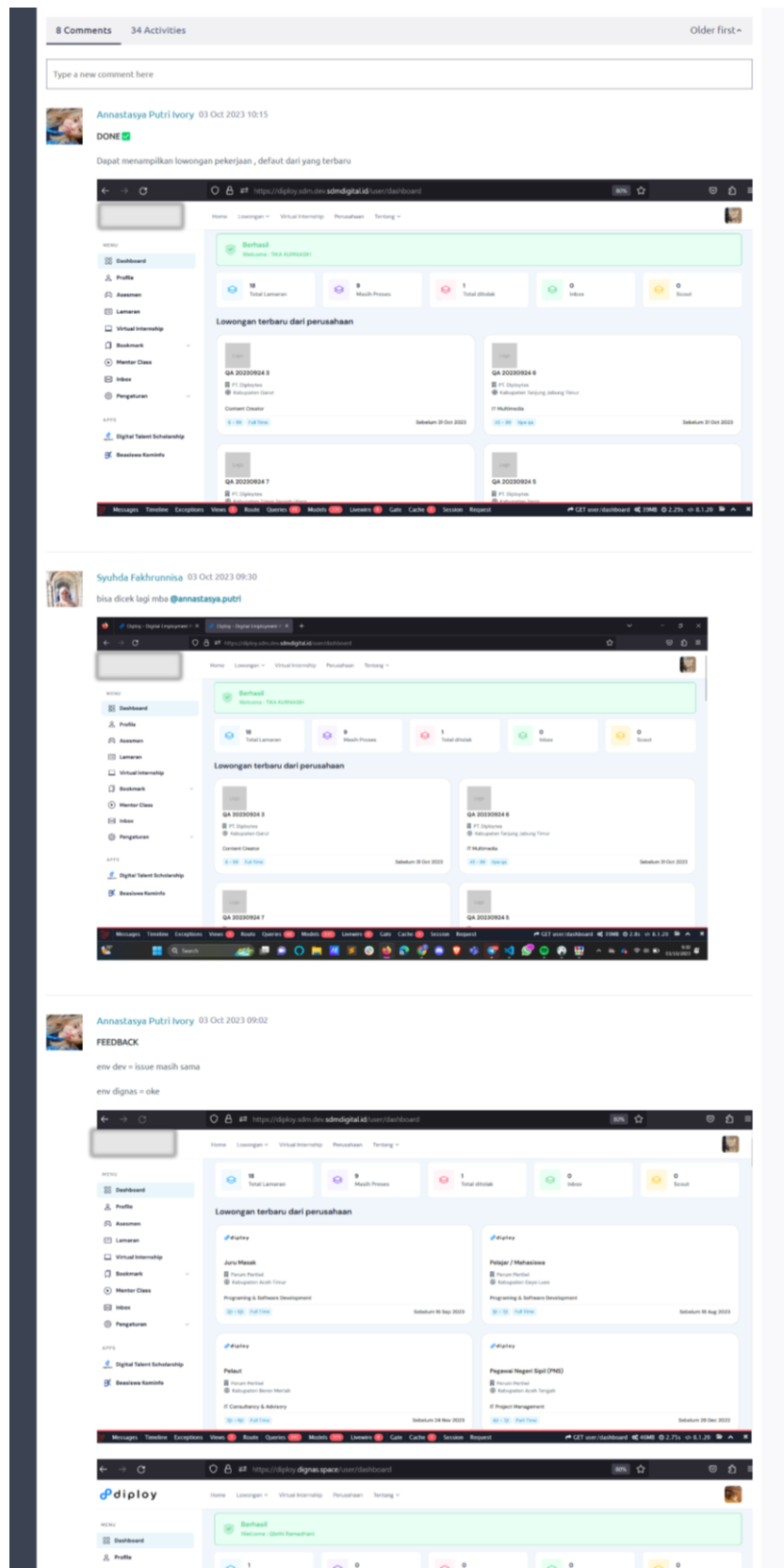
- Dapat menampilkan lowongan pekerjaan , default dari yang terbaru

ASSIGNED
Annastasya Putri Ivory
+ Add assigned Assign to me

WATCHERS
Afdi Fauzul Bahar
Annastasya Putri Ivory
Syuhda Fakhrunnisa
+ Add watchers Unwatch

Gambar 3.35 Detail *task* pengurutan data terurut *default* dari yang terbaru di Taiga

Gambar 3.36 menunjukkan hasil implementasi di mana lowongan pekerjaan ditampilkan berdasarkan urutan waktu, dengan lowongan terbaru muncul di posisi teratas.



Gambar 3.36 Hasil implementasi pengurutan data terurut *default* dari yang terbaru

Melakukan penyesuaian tampilan komponen/halaman

Penulis mendapat beberapa *task* untuk menyesuaikan tampilan komponen dan halaman agar lebih optimal. Seperti yang terlihat pada Gambar 3.37, penulis ditugaskan untuk menyesuaikan tampilan tombol *edit* yang perlu di-*disable* pada data bidang pekerjaan yang sudah digunakan atau direferensikan.

#1938 Tidak bisa edit data bidang pekerjaan yang sudah digunakan/direferensikan

TASK
Edit with S2
This task belongs to #1937 (SP181) Sebagai Admin, saya dapat melihat data bidang pekerjaan beserta log aktivitas

Created by Anastasya Putri Ivory
03 Oct 2023 11:13

Actor
Admin

Skenario

Aktor	Sistem
1. login as Admin	2. Menampilkan dashboard
3. klik menu master data, sub menu Bidang Pekerjaan	4. menampilkan halaman bidang pekerjaan
	5. button edit masih enable pada data yan digunakan/direferensikan

Existing Condition

Screenshot of the application showing a table of job fields. The table has columns for ID, Name, and Date. The 'Bidang pekerjaan' row (ID 7) has an 'edit' button that is not disabled.

Expected Result

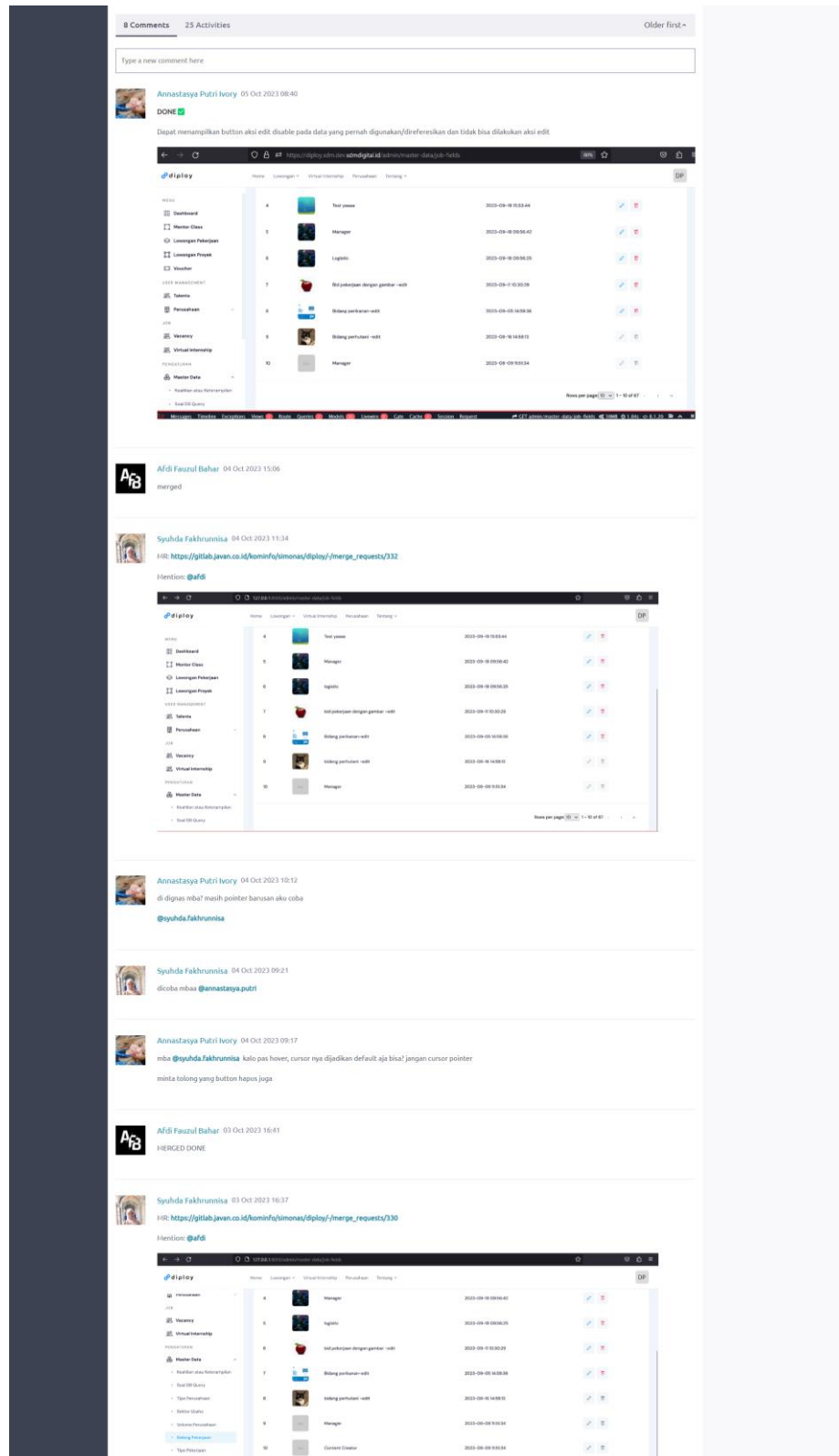
Screenshot of the application showing the same table. The 'edit' button for the 'Bidang pekerjaan' row (ID 7) is now disabled (greyed out).

Definition of Done

- Dapat menampilkan button aksi edit disable pada data yang pernah digunakan/direferensikan dan tidak bisa dilakukan aksi edit

Gambar 3.37 *Task* penyesuaian tampilan *disable* tombol *edit*

Gambar 3.38 menunjukkan hasil implementasi di mana tombol *edit* berhasil dinonaktifkan untuk data yang tidak dapat dimodifikasi, dan memenuhi *request* untuk *cursor*-nya dijadikan *cursor default* (bukan *cursor pointer*) saat melakukan *hover* pada tombol.



Gambar 3.38 Hasil implementasi penyesuaian tampilan *disable* tombol *edit*

Selanjutnya, penulis juga diberikan *task* untuk memperbaiki *wording* pada status soal tes. Seperti yang terlihat pada Gambar 3.39, terdapat kebutuhan untuk mengubah status soal tes menjadi lebih informatif.

#2005 (FE) Terdapat penggantian wording pada status soal tes

TASK

Edit with S2
This task belongs to #2002 (DP10) Sebagai Presenter, saya dapat melihat perubahan tampilan halaman soal tes laman.

Created by Anastasya Putri Ivory
09 Oct 2023 17:14

Actor
Peserta

Skenario

Aktor	Sistem
1. Login as Peserta	2. Menampilkan dashboard
3. Klik menu Laman	4. menampilkan halaman Laman
4. Klik "What soal tes" pada salah satu data	7. wording pada status soal perlu diganti

Existing Condition

Screenshot 1: Dashboard showing 'Soal Tes: IT Support Borneo (yang benar)' with a status of 'Soal Kompetensi Bidang'.

Screenshot 2: Detail view of the test question with a status of 'Soal Praktek'.

Endpoint URL
<https://dploy-admin.sdm.digilab.com/applicants/1/soal/10>

Expected Result

Screenshot 1: Dashboard showing 'Soal Tes: IT Support Borneo (yang benar)' with a status of 'Soal Kompetensi Bidang'.

Screenshot 2: Detail view of the test question with a status of 'Soal Praktek'.

Definition of Done

- Terdapat penggantian wording status soal tes "Soal Praktek" → "Soal Kompetensi Bidang"
- contoh: "Soal Kompetensi Bidang sudah dikerjakan", "Soal Praktek belum dikerjakan", "Sudah dikerjakan"

Gambar 3.39 Task perbaikan *wording* status

Gambar 3.40 memperlihatkan hasil perbaikan di mana *wording* status telah diubah sesuai dengan ketentuan yang diminta, yaitu mengubah *wording* status soal tes yang awalnya “Sudah disubmit” menjadi “Sudah dikerjakan”.

The image displays a screenshot of a web application interface, likely a learning management system or a test management tool. The interface shows a list of test questions with their status updated from "Sudah disubmit" (Submitted) to "Sudah dikerjakan" (Completed/Worked on).

The screenshot is divided into two main sections:

- Top Section:** Shows a comment by Annastasya Putri Ivory on 09 Oct 2023 13:31. The comment status is "DONE". The text of the comment reads: "Terdapat penggantian wording status soal tes 'disubmit' --> 'dikerjakan'". Below the comment is a screenshot of the application showing a list of test questions. The status of the questions has been updated from "Sudah disubmit" to "Sudah dikerjakan".
- Bottom Section:** Shows a comment by Afdi Fauzul Bahar on 09 Oct 2023 11:16. The comment status is "done merged". Below the comment is a screenshot of the application showing a list of test questions. The status of the questions has been updated from "Sudah disubmit" to "Sudah dikerjakan".

The application interface includes a sidebar menu with options like Dashboard, Profile, Asesmen, Lamaran, Virtual Internship, Bookmark, Mentor Class, Inbox, and Pengaturan. The main content area displays a list of test questions with their status and a "Selesaikan" (Complete) button.

Gambar 3.40 Hasil implementasi perbaikan *wording* status

Penulis juga mendapat *task* untuk menyesuaikan tampilan teks dan komponen modal. Seperti yang terlihat pada Gambar 3.41, penulis ditugaskan untuk meredesain ukuran *font* pada bagian deskripsi dan menyesuaikan tampilan modal konfirmasi.

#2004 (FE) Redesain ukuran font deskripsi dan modal konfirmasi

TASK

Edit with J2

This task belongs to #1002 (SP10) Sebagai Penulis, saya dapat melihat perubahan tampilan halaman soal tes lamaran @

TAKEBOARD

Created by Anastasya Putri Iwory
05 Oct 2023 17:11

Assigned to Anastasya Putri Iwory

Watchers: Afri Fauzi Bahar, Anastasya Putri Iwory, Syahda Fakhrunnisa

Aktor

Penulis

Skenario

Aktor	Sistem
1. login as Penulis	2. Memampilkan dashboard
3. klik menu Lamaran	4. menampilkan halaman Lamaran
6. klik "Detail soal tes" pada salah satu soal	7. menampilkan halaman soal tes
8. klik "Kunjikan" pada salah satu soal	9. menampilkan halaman deskripsi soal
11. klik "Jawab"	10. menampilkan deskripsi
	12. muncul modal konfirmasi
	13. redesain teks pada soal

Existing Condition

https://diploy.sdm.dewadigital.id/applicants/27/subatani

Endpoint URL

https://diploy.sdm.dewadigital.id/api/applicants/27/soal/subatani

Expected Result

https://diploy.sdm.dewadigital.id/applicants/27/soal/subatani

Definition of Done

- Berhasil mengganti font size deskripsi menjadi 12pt
- Berhasil mengganti font size deskripsi menjadi justify
- Berhasil mengganti font size deskripsi menjadi 1.5em (Konfirmasi) & 1.5em (Apakah anda yakin untuk memulai tes?)
- Kata "anda" huruf A nya besar

Gambar 3.41 *Task* redesain ukuran *font* deskripsi dan modal konfirmasi

Selain itu, penulis juga diberikan *task* untuk memperbaiki tampilan daftar soal. Seperti yang terlihat pada Gambar 3.43, terdapat kebutuhan untuk mengubah ikon dan menyesuaikan tata letak pada *list* soal.

#2007 (FE) Terdapat perubahan ikon waktu pengerjaan tes dan tata letak list soal practitioner

TASK

ESB with ID
This task belongs to #1002 (DP102) Sebagai Peminta, saya dapat melihat perubahan tampilan halaman soal tes layanan

TO DO BOARD

Created by Amestasya Putri Ivory
01 Oct 2023 11:14

Actor

Peminta

Skenario

Aktor	Sistem
1. Login as Peminta	2. Menampilkan dashboard
3. Klik menu Layanan	4. menampilkan halaman Layanan
5. Klik "What soal tes?" pada salah satu data	6. menampilkan halaman list soal tes
7. Klik terapkan pada soal practitioner	8. menampilkan halaman daftar soal practitioner
	9. tata letak perlu diubah

Existing Condition

Endpoint URL

<https://display-admin.adm.digital.id/user/applicants/what/practitioner>

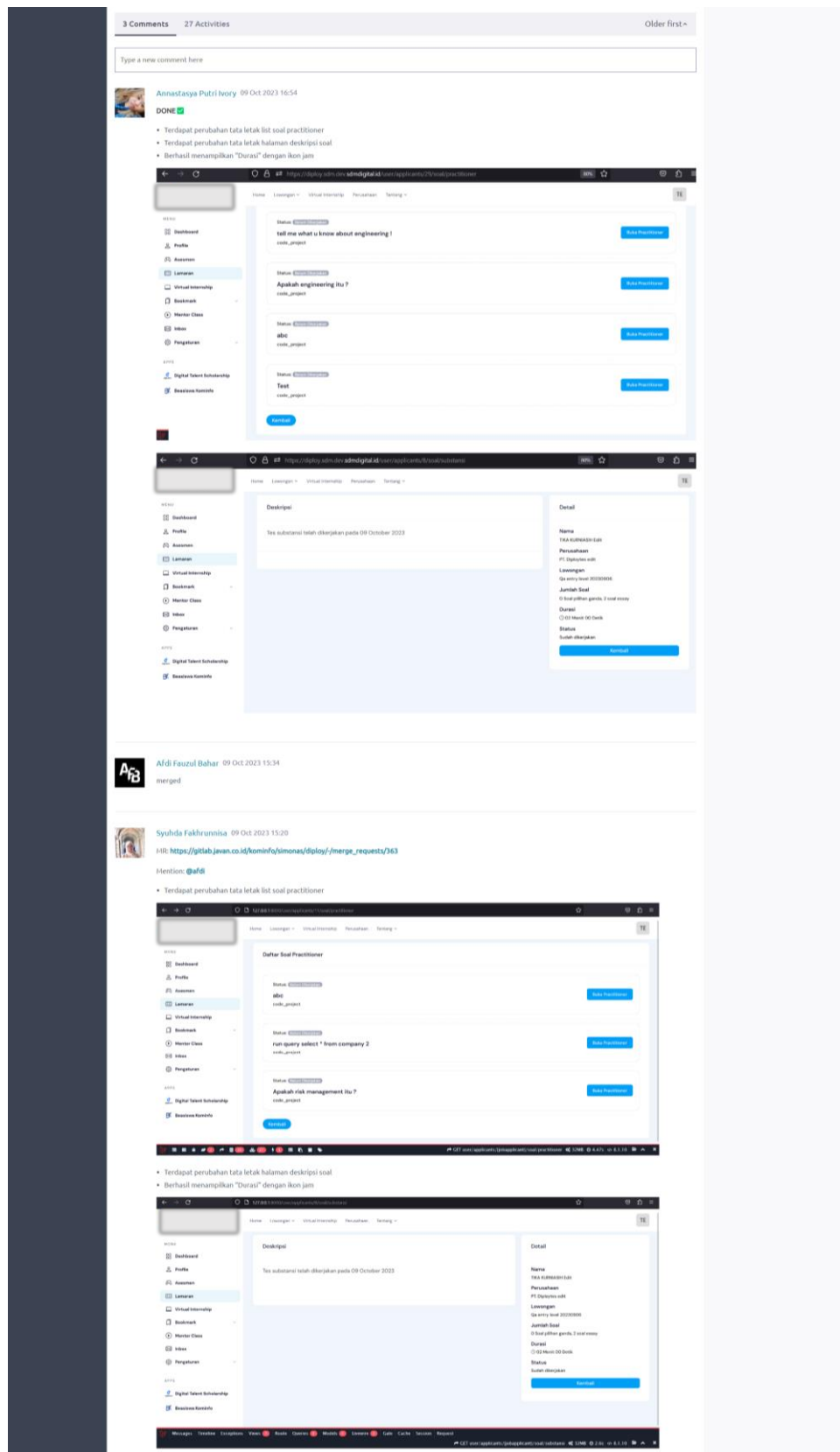
Expected Result

Definition of Done

- Terdapat perubahan tata letak list soal practitioner
- Terdapat perubahan tata letak halaman deskripsi soal
- Berhasil menampilkan "Time" dengan ikon jam

Gambar 3.43 *Task* perubahan ikon dan tata letak *list* soal

Gambar 3.44 memperlihatkan hasil perbaikan di mana ikon dan tata letak telah diubah sesuai dengan ketentuan yang diminta, sehingga dapat lebih memudahkan pengguna.



Gambar 3.44 Hasil implementasi perubahan ikon dan tata letak list soal

Pengujian oleh *Quality Assurance* (QA)

Setelah suatu *task* memiliki status *closed*, pengujian akan dilakukan oleh tim *Quality Assurance* (QA). QA menggunakan beberapa *tools*, termasuk QA Touch, yang merupakan platform manajemen pengujian komprehensif dari DCKAP. Alat ini dirancang untuk menangani berbagai aspek pengujian proyek secara efisien, mulai dari perencanaan, pelaksanaan, pelacakan *bug*, hingga pelaporan status aktivitas QA. QA Touch digunakan di berbagai industri, seperti perangkat lunak, ritel, *fintech*, dan lainnya, dengan solusi manajemen pengujian yang disesuaikan untuk kebutuhan spesifik .

Proses pengujian diawali dengan penyusunan daftar *test case* oleh QA berdasarkan *acceptance criteria* dari *task* yang telah dirancang dan mendokumentasikannya melalui QA Touch. Selanjutnya, *system analyst* akan membuat dan mendokumentasikan *issue* yang terkait dengan *test case* dan *task* tersebut. *System analyst* juga melakukan *monitoring* terhadap *task* terkait, dan ketika *task* sudah bisa dianggap selesai maka *issue* akan ditugaskan kepada QA untuk dilakukan pengujian validasi.

Pengujian yang dilakukan oleh QA mencakup pengujian fungsional, di mana semua fitur baru diuji secara fungsional berdasarkan spesifikasi dan *acceptance criteria* untuk memastikan bahwa fitur tersebut berfungsi dengan baik sebelum dirilis ke lingkungan produksi. Selain itu, dilakukan juga pengujian non-fungsional seperti performa, keamanan, dan kompatibilitas untuk memastikan aplikasi memenuhi standar kualitas yang ditetapkan. Pengujian regresi turut dilakukan secara rutin untuk memastikan perubahan kode baru tidak mengganggu fungsionalitas yang sudah ada. Seluruh hasil pengujian, termasuk *bug* yang ditemukan, langkah-langkah reproduksi, dan status perbaikannya, didokumentasikan dengan rinci untuk memastikan transparansi dan efektivitas dalam tindak lanjut. Kolaborasi antara QA dan tim pengembang juga berjalan secara intensif guna memastikan *bug* diidentifikasi, dilaporkan, dan diselesaikan dengan cepat.

3.3.4 Pemantauan dan Pengendalian Proyek

Proses pengendalian dan pemantauan dalam pengembangan Proyek A dilakukan dengan memanfaatkan tiga *tools* utama, yaitu GitLab, Taiga, dan Metabase. GitLab berfungsi sebagai *Project Management Information System* (PMIS) untuk pengelolaan dan pemantauan setiap tugas yang diberikan kepada tim (Oktavanusa, 2022). Pemanfaatan GitLab memungkinkan *lead* untuk memantau seluruh pekerjaan yang ditugaskan kepada *developer* dengan mudah. Selain itu, pemantauan proyek juga dilakukan menggunakan Taiga, di mana *lead* dapat

mengawasi pencapaian *task-task* yang terdaftar dalam *sprint*, apakah telah tercapai secara keseluruhan atau masih ada *task* yang terlambat dan mengalami kendala. Di sisi lain, *tools* utama yang sering digunakan oleh *lead* adalah Metabase. Platform intelijen bisnis *open-source* ini memungkinkan pengguna untuk mengajukan berbagai pertanyaan terkait data dan dapat diintegrasikan ke dalam aplikasi, memberikan kesempatan kepada pelanggan untuk menjelajahi data mereka sendiri (Metabase, 2024). PT Javan Cipta Solusi memanfaatkan Metabase ini untuk berbagai tujuan pemantauan, termasuk *monitoring tasks* tim, pengukuran skor capaian, pemantauan target harian, dan berbagai keperluan lainnya.

3.3.5 Penutupan Proyek

Proses penutupan aktivitas untuk Proyek A belum dapat dilakukan karena masih terdapat beberapa *issue* yang perlu diselesaikan. Pengecekan kualitas kode menggunakan SonarQube masih diperlukan untuk memastikan standar kode terpenuhi. Pelaksanaan *functional test* juga masih berlangsung dengan menggunakan *tools* baru, yaitu Apidog, menggantikan Postman yang sebelumnya digunakan oleh tim proyek. Beberapa dokumen *deliverable* pun masih membutuhkan penyelesaian dan penyempurnaan. Oleh karena itu, penyelesaian seluruh *issue* dan kebutuhan lainnya dari Proyek A akan dilanjutkan oleh karyawan lain yang masih aktif bekerja di PT Javan Cipta Solusi.

3.4 Pengembangan Proyek B

3.4.1 Pendefinisian Proyek

Proyek B merupakan Modul Administrasi Keuangan yang digunakan oleh *user* dari sebuah instansi pemerintah untuk mengelola data kegiatan (realisasi), perjalanan dinas dan kuitansi. Modul ini memungkinkan sinkronisasi seluruh data terkait kegiatan dan realisasinya dengan nilai pagu awal, hingga tahap pertanggungjawabannya, sehingga memudahkan proses *monitoring* secara menyeluruh.

Tujuan dari Modul Administrasi Keuangan adalah sebagai berikut:

1. Mengelola data kegiatan (realisasi), perjalanan dinas, dan kuitansi di lingkungan internal instansi terkait.
2. Melakukan *monitoring* nilai kegiatan yang disinkronkan oleh nilai pagu awal.

Nilai pagu awal merupakan batas maksimal anggaran atau alokasi dana yang sudah ditetapkan di awal suatu proyek atau kegiatan, sebelum pelaksanaan kegiatan tersebut dimulai. Dalam konteks administrasi keuangan, pagu awal berfungsi sebagai acuan agar pengeluaran

atau realisasi biaya suatu proyek tidak melebihi anggaran yang telah disetujui sebelumnya. Pagu ini biasanya ditetapkan berdasarkan perencanaan anggaran oleh instansi atau organisasi yang bertanggung jawab atas proyek tersebut.

3.4.2 Inisialisasi Proyek

Peran

Pengembangan sebuah proyek perangkat lunak memerlukan berbagai peran dan teknologi untuk memastikan keberhasilan proyek. Beberapa peran dan teknologi diterapkan dalam pengembangan Proyek B. Selama magang, penulis tergabung dalam tim sebagai *mentee (frontend) programmer*. Adapun rincian peran-peran dari *tribe member* beserta akuntabilitasnya adalah sebagai berikut.

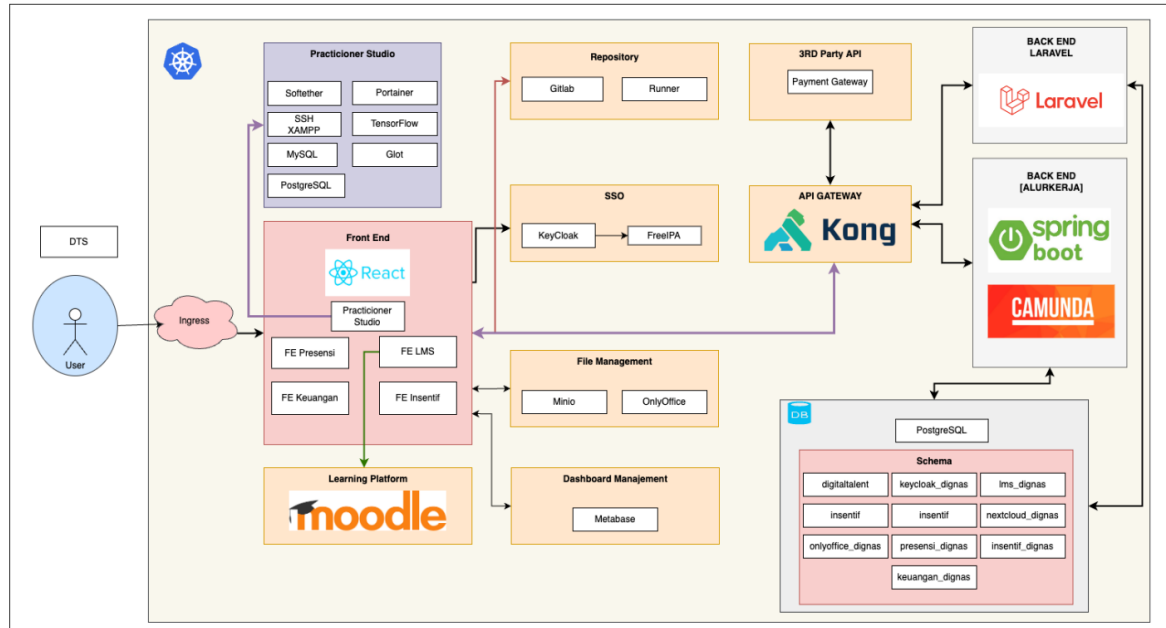
1. *Tribe lead*, bertanggung jawab sebagai pemimpin tim, dengan tugas utama meliputi koordinasi antar tim, fasilitasi pertemuan, serta *mentoring* dan pengembangan anggota tim. Selain itu, peran ini mencakup manajemen risiko dan hambatan, pengembangan budaya kerja yang positif, pemantauan kapasitas serta beban kerja tim, dan penyelesaian konflik. *Tribe lead* juga berperan dalam melakukan evaluasi sprint, pemecahan masalah teknis, serta memahami aspek teknis dari produk yang sedang dikembangkan.
2. *Product owner*, bertanggung jawab sebagai *leader* yang mengarahkan *Project Squad* untuk mencapai GOALS proyek. Peran ini meliputi memastikan agar para *programmer* tetap produktif dengan selalu menyediakan *task* yang tersedia dan menetapkan prioritas *task*-nya. *Product owner* juga menyusun dokumen perencanaan *sprint*, menjadwalkan *sprint planning* dan *sprint review*, serta membantu memastikan proses *testing* dilakukan dengan baik oleh analis. Selain itu, tugasnya termasuk me-review dokumen *deliverable* dan memiliki akuntabilitas terhadap perencanaan *user story*.
3. *Analyst*, bertanggung jawab untuk melakukan fase *shaping* dan analisis sesuai dengan arahan *product owner*. Tugas lainnya termasuk membuat *task* sesuai dengan standar yang telah ditetapkan, melakukan *testing* berdasarkan *acceptance criteria* yang diberikan, serta menyusun *Minutes of Meeting* (MoM) dan membuat *invitation meeting*. *Analyst* juga terlibat dalam pembuatan dan revisi dokumen *deliverable*, membuat *task* pada *Issue Sentry* dan QA Touch, serta membuat *history log issue*.

Selain itu, *analyst* harus menjaga angka ‘Ready For Test’ harian agar tidak lebih dari 24 Jam.

4. *Developer* atau *programmer*, bertanggung jawab menyelesaikan *task* sesuai dengan tenggat waktu dan *acceptance criteria* yang telah ditentukan. Mereka terlibat aktif dalam perencanaan *sprint* dan *review sprint* untuk memberikan masukan terkait estimasi waktu serta tingkat kompleksitas pengembangan. Selain itu, *developer* turut berpartisipasi dalam pertemuan retrospektif *sprint*, memberikan umpan balik untuk meningkatkan efisiensi tim. Tanggung jawab lainnya termasuk mengidentifikasi, melaporkan, serta memperbaiki *bug* atau masalah teknis yang muncul selama proses pengembangan.
5. *Quality Assurance (QA)*, bertanggung jawab merancang dan mengembangkan strategi pengujian. Tugas lainnya mencakup membuat dan memelihara skrip pengujian, melakukan pengujian fungsional dan non-fungsional, serta melakukan pengujian regresi. QA juga mendokumentasikan hasil pengujian, berkolaborasi dengan tim pengembang untuk mengidentifikasi, melaporkan, dan menyelesaikan *bug* dengan cepat dan efisien, serta memastikan bahwa setiap fitur mematuhi *acceptance criteria*. Selain itu, QA bertugas melaporkan dan menganalisis *bug*, membuat serta memelihara dokumen pengujian, serta melakukan *review* dan validasi hasil pengujian.

Teknologi

Pengembangan Proyek B menggunakan berbagai teknologi yang terintegrasi dalam satu sistem. Modul FE Keuangan dikembangkan menggunakan React sebagai *framework frontend*. Pada sisi *backend*, proyek ini menggunakan dua *framework* utama: Laravel dan Spring Boot, dengan Camunda untuk mengelola alur kerja (*workflow*). Basis data menggunakan PostgreSQL untuk penyimpanan data. Arsitektur ini juga dilengkapi dengan Kong sebagai *API Gateway* untuk mengelola dan mengamankan komunikasi antar layanan. Visualisasi lengkap terkait arsitektur dan teknologi yang digunakan dalam Proyek B dapat dilihat pada Gambar 3.45.



Gambar 3.45 Diagram arsitektur sistem Proyek B

3.4.3 Perencanaan dan Pelaksanaan Proyek

Ketika penulis bergabung, proyek sudah berada di tahap akhir *development*, sehingga keterlibatan penulis lebih berfokus pada fase *development* akhir. Pada fase ini, sebagian besar fitur utama telah dikembangkan, dan tim mulai beralih pada *bug fixing* serta penyempurnaan desain UI/UX berdasarkan *feedback* dari *stakeholder*. Penulis diberikan tugas untuk memperbaiki *issue* terkait tampilan *frontend*, menyempurnakan interaksi antar-komponen, serta melakukan beberapa penyesuaian pada elemen desain yang mengalami perubahan setelah *review sprint*. *Milestone* pertama untuk penyelesaian proyek ini ditargetkan di akhir tahun 2023. Namun, karena terdapat prioritas modul lain yang dianggap lebih *urgent*, pengembangan modul ini ditunda sementara waktu dan fokus tim beralih ke penyelesaian modul-modul tersebut. Selama periode ini, penulis berkontribusi dalam beberapa tugas, antara lain:

Melakukan Setup Proyek di *Local*

Tugas awal yang diterima penulis adalah melakukan *setup* proyek B di *local*. Seperti yang terlihat pada Gambar 3.46, penulis ditugaskan untuk memastikan proyek dapat berjalan dengan baik di *local development environment* sebelum memulai pengerjaan *task-task* berikutnya.

The screenshot shows a Jira task page for a project named 'Aplikasi Keuangan'. The task is titled '#2227 [FE] Terdapat Setup Aplikasi untuk Front End' and is in a 'CLOSED' state. It was created by Muhammad Royyan Saputra on 21 Sep 2023. The assignee is Syuhda Fakhrunnisa. The task description asks for actor roles and acceptance criteria. A screenshot of a local application setup is attached, showing a dashboard with metrics like 'Total Sales 25.8K' and 'Page Views 2.6M'.

Gambar 3.46 Detail *task setup* proyek B di Taiga

Melakukan Penyesuaian Tampilan Aplikasi

Penulis mendapat beberapa *task* untuk menyesuaikan tampilan pada modul keuangan. Seperti yang terlihat pada Gambar 3.47, penulis ditugaskan untuk menyesuaikan warna pada *header* tabel agar konsisten dengan tampilan keseluruhan.

#2297 Terdapat penyesuaian warna pada judul kolom

TASK

Edit with S2

This task belongs to #1055 [SP103] Sebagai user, saya dapat melihat halaman list Realisasi Keuangan

TASKBOARD

Add tag +

Created by Muhammad Royyan Saputra
12 Oct 2023 13:39

User

PPK

Skenario

1. Login as ppk
2. Klik semua menu yang memiliki tabel
3. Menampilkan halaman list tabelnya

Kondisi Saat ini

Pada kolom judul tabel masih belum sesuai dengan dts warna pada judul kolom

Kegiatan Keuangan

No	Tanggal Dibuat	Judul	Tipe	Dasar	Proses	Status Approval
1	14 Oktober 2023	uang anggaran tahunan	Perjalanan Dinas Dalam Negeri	-	Membuat Surat Tugas	Approval
2	12 Oktober 2023	Repudiandae nihil no	Perjalanan Dinas Dalam Negeri	-	Membuat Surat Tugas	Approval
3	12 Oktober 2023	perjalanan dinas	Perjalanan Dinas Dalam Negeri	Surat Tugas: 042/BSLSDM.3/KP.01.06/10/2023	Proses Spd	Tanpa Approval

Acceptance Criteria

1. Terdapat penyesuaian warna judul pada judul kolom sesuai dengan front dev
2. Login front dev : <http://front.dev.dtsnet.net/pelatihan/pelatihan>

javani@gmail.com
Kominfo123#

Pilih Pelatihan → pelatihan

List Rekap Pelatihan

No	ID	Nama Pelatihan	Alur Pendaftaran	Lokasi	Jadwal Pendaftaran	Jadwal
1	TA5940	Online Bebras Anak (Batch 1) Puslibzong IGP	Administrasi	Online	04 - 12 Okt 2023 12:00:00 - 12:00:00	16 - 21 O 12:00:00 -
2	DLA5939	Offline Pelatihan Selenium Oktober ALV 2023 (Batch 1) Puslibzong Aptika IGP	Test Substansi - Administrasi	Kota Jakarta Pusat	01 - 15 Okt 2023 12:00:00 - 12:00:00	15 - 31 O 12:00:00 -

Custom Fields

Analyst
List: Analyst
Muhammad Royyan Saputra

Programmer
List: Programmer

Code Reviewer
List: Code Reviewer
Imam Setiawan

Tester
List: Tester
Muhammad Royyan Saputra

Technical Writer
List: Technical Writer

QA

Mentee Analyst

Mentee Tester

Mentee Programmer
Syuhda Fakhrunnisa

ASSIGNED

Muhammad Royyan Saputra

+ Add assigned Assign to me

WATCHERS

Muhammad Royyan Saputra

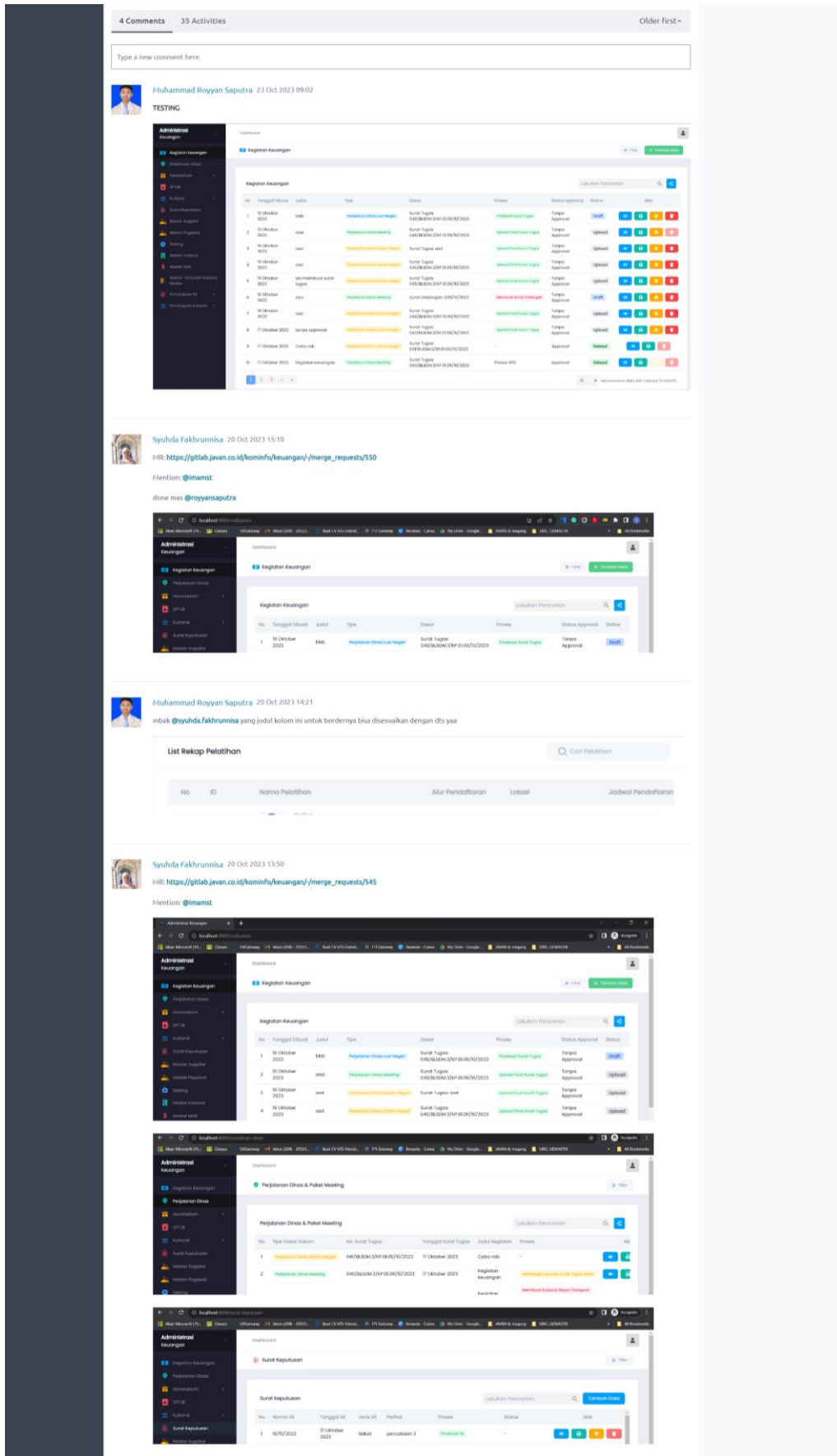
Imam Setiawan

Syuhda Fakhrunnisa

+ Add watchers Unwatch

Gambar 3.47 Task penyesuaian tampilan header tabel

Gambar 3.48 menunjukkan hasil implementasi di mana header tabel telah disesuaikan, sesuai dengan standar desain yang ditentukan.



Gambar 3.48 Hasil implementasi penyesuaian tampilan header tabel

Selain itu, penulis juga diberikan *task* untuk memperbaiki tampilan *loading* pada tombol finalisasi nota dinas. Pada Gambar 3.49, terlihat kebutuhan untuk memperbaiki indikator *loading* yang muncul saat tombol diakses. Seharusnya, ketika satu tombol di klik, maka indikator *loading* hanya akan muncul di tombol tersebut saja, tombol lain tidak perlu menampilkan indikator yang sama.

#2301 Terdapat perbaikan loading finalisasi nota dinas

TASK

Edit with S2

This task belongs to #2066 [SP094] Sebagai PBJ dapat melakukan approval nota dinas

Created by Muhammad Royyan Saputra
13 Oct 2023 07:58

User

PPK

Skenario

1. Login as ppk
2. Klik menu kuitansi
3. Pilih kuitansi kontraktual
4. Pilih proses finalisasi nota dinas
5. Klik button finalisasi

Kondisi Saat ini

Terdapat button save as draft dan finalisasi nota dinas loading keduanya ketika diklik

Dashboard

Yth. : Pusat Pengembangan Profesi dan Sertifikasi
Dari : PT. DCSFR
Perihal :
Klasifikasi : Klasifikasi
Tanggal :
Lampiran : Lampiran

Bersama ini kami sampaikan bahwa Pusat Penelitian dan Pengembangan Aplikasi Informatika dan Informasi dan Komunikasi Publik (Pusat Aplikasi dan IKP) Badan Litbang SDM akan mengadakan kegiatan yaitu "UAT Modul Keuangan" yang akan dilaksanakan secara daring. Untuk itu mohon bantuan Saudara dapat menyiapkan dan memproses pengadaan terkait dengan rincian terlampir sesuai ketentuan yang berlaku. Nilai total harga perkiraan sendiri (HPG) untuk pengadaan tersebut sebesar Rp100.000.000,00 (Seratus Juta Rupiah).
Demikian disampaikan. Atas perhatian dan bantuan Saudara, kami ucapkan terima kasih.

Sebelumnya Loading... Loading...

Acceptancer Criteria

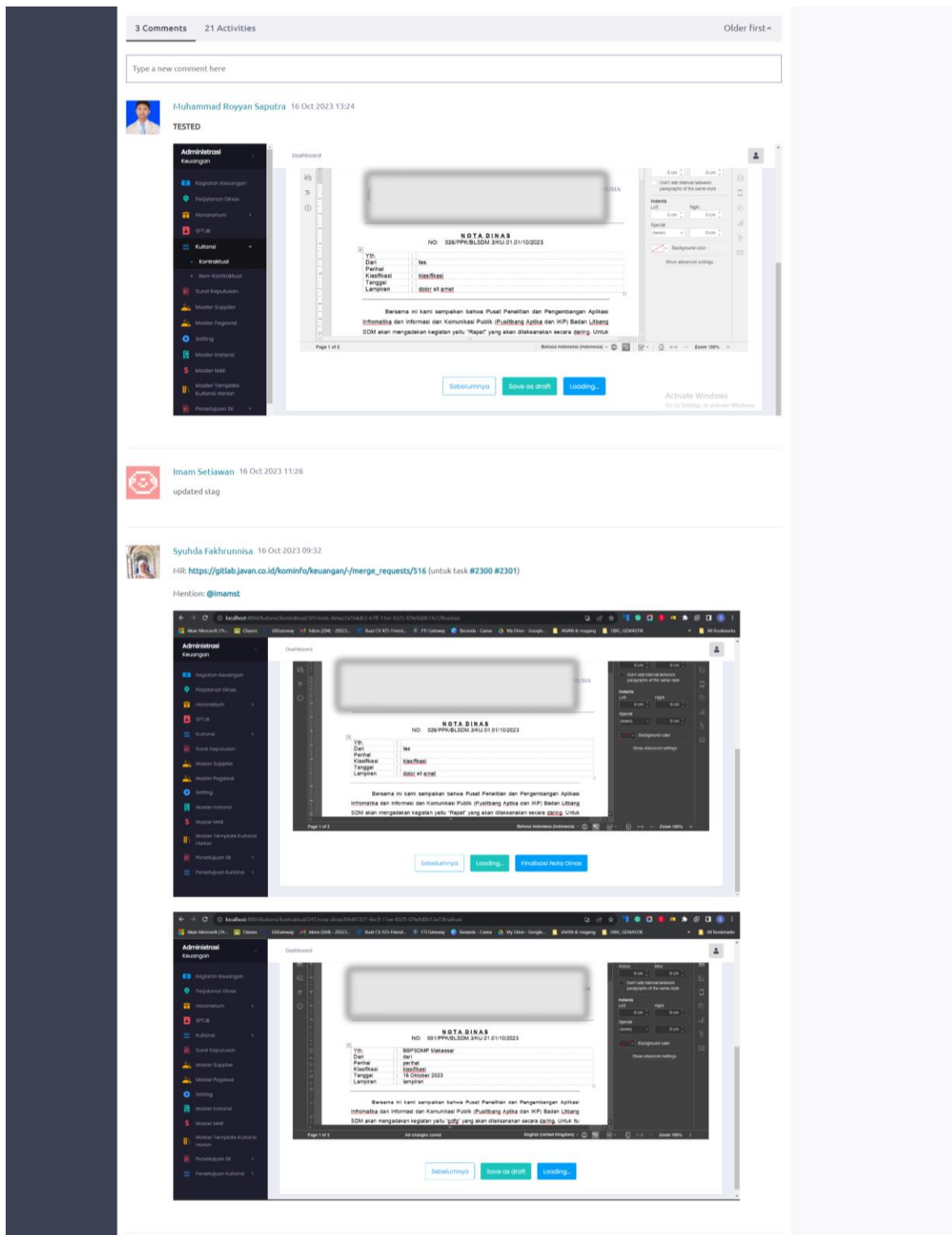
1. Button loading cukup pada finalisasi nota dinas atau save draft ketika klik salah satu nya

Custom Fields

Analyst List Analyst	Muhammad Royyan Saputra
Programmer List Programmer	
Code Reviewer List Code Reviewer	Imam Setiawan
Tester List Tester	Muhammad Royyan Saputra
Technical Writer List Technical Writer	
QA	
Mentee Analyst	
Mentee Tester	
Mentee Programmer	Syuhda Fakhrunnisa

Gambar 3.49 *Task* perbaikan indikator *loading* pada tombol finalisasi

Gambar 3.50 memperlihatkan hasil implementasi di mana indikator *loading* telah diperbaiki dan berfungsi sesuai dengan yang diharapkan.



Gambar 3.50 Hasil implementasi perbaikan indikator *loading* pada tombol finalisasi

Penulis juga mendapat *task* untuk menyesuaikan tampilan pada formulir. Seperti yang terlihat pada Gambar 3.51, penulis ditugaskan untuk menambahkan *default value* pada kolom deskripsi dan rincian di form honor narasumber.

#2059 Terdapat default value pada kolom Deskripsi dan rincian di form honor narasumber

TASK

Edit with S2

This task belongs to #2038 [SP079] Sebagai ppk dapat membuat honor narasumber dan edit narasumber

Created by Firatus Saadeh
06 Sep 2023 07:19

Scenario:

1. login as PPK
2. pilih menu Honorarium
3. klik sub menu Honorarium Narasumber
4. klik buton tambah data

Kondisi Saat Ini:

belum ada default value pada kolom Deskripsi

Input Data Honor Narasumber

No	Nama Kegiatan	Tanggal Kegiatan	Revisi	Deskripsi	Frekuensi	Saldo	Luasan (km ²)	PPH 20%	PPH 0%	Aksi
1	21/10/2023	10/10/2023		Muang Honoraria Pengajaran						

Acceptance Criteria:

1. terdapat default value pada kolom Deskripsi yaitu: Honor Narasumber Kegiatan, sehingga hasil akhirnya akan terdapat default value tersebut dan user masih bisa mengedit untuk menambahkan keterangan deskripsi.
2. Pada kolom rincian terdapat default yaitu Honor Narasumber + Rp.Xjam
3. Contoh dokumen seperti berikut

TA :
Nomor Bukti :
Mata Anggaran : 4495.AFA.001.052.A.522151
Evaluasi dan Updating Peta Okupasi - Belanja Jasa Profesi

KUITANSI / BUKTI PEMBAYARAN

Sudah terima dari : BADAN PENELITIAN DAN PENGEMBANGAN SUMBER DAYA MANUSIA (664297)

Jumlah Uang : **Rp. 2.000.000**

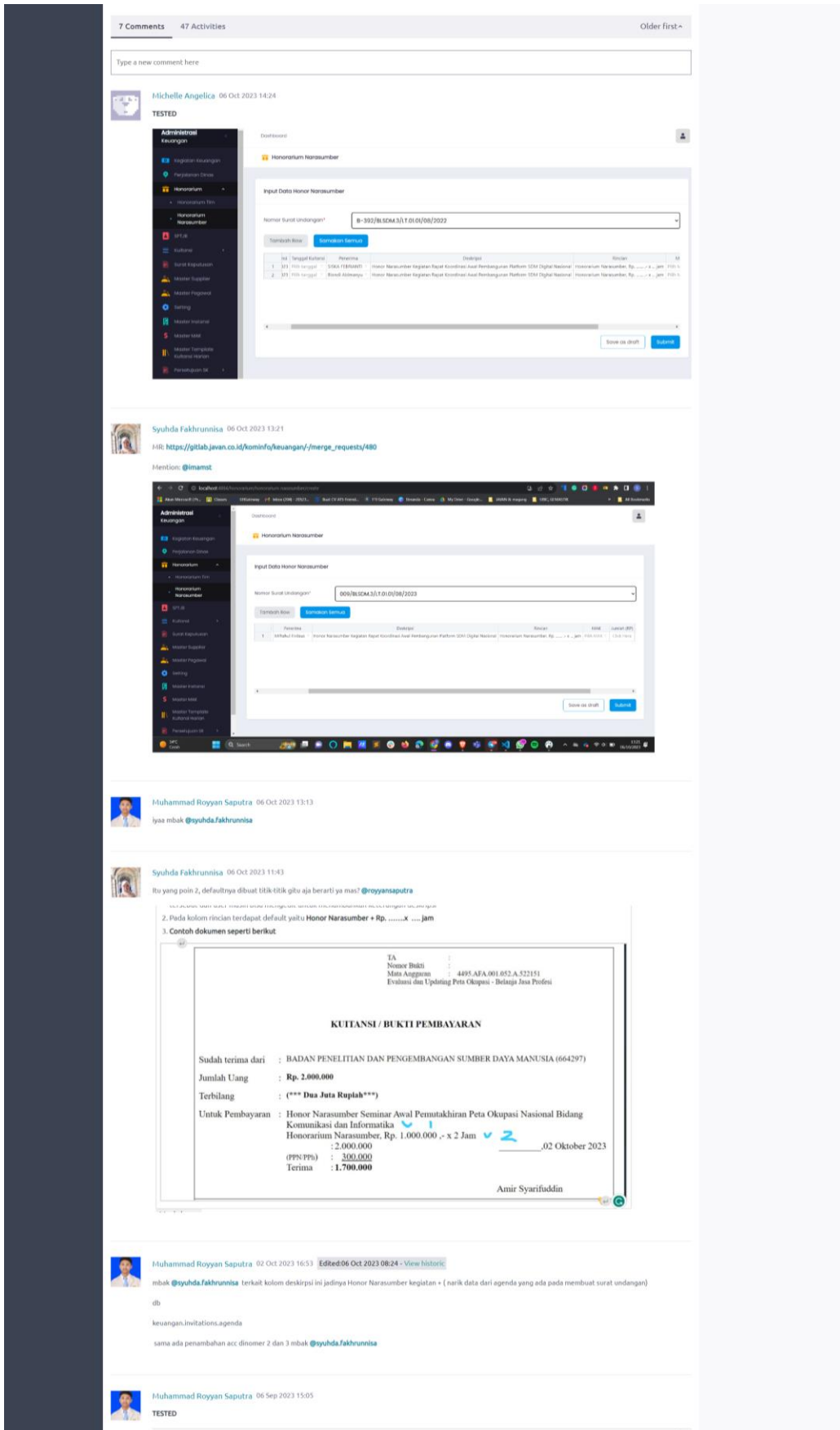
Terbilang : **(*** Dua Juta Rupiah***)**

Untuk Pembayaran : Honor Narasumber Seminar Awal Pemutakhiran Peta Okupasi Nasional Bidang Komunikasi dan Informatika
Honorarium Narasumber, Rp. 1.000.000,- x 2 Jam
: 2.000.000
(PPN-PPH) : 300.000
Terima : **1.700.000**

Amir Syarifuddin

Gambar 3.51 *Task* penambahan *default value* pada kolom deskripsi dan rincian

Gambar 3.52 menunjukkan hasil implementasi di mana *default value* telah berhasil ditambahkan sesuai dengan ketentuan, sehingga dapat mempersingkat waktu pengguna dalam pengisian *form* karena hanya mengisikan bagian-bagian penting yang diperlukan saja.



Gambar 3.52 Hasil implementasi penambahan *default value* kolom deskripsi dan rincian

Selain itu, penulis juga diberikan tugas untuk memperbaiki tampilan *field* ‘Nama Supplier’. Pada Gambar 3.53, terlihat kebutuhan untuk menyesuaikan tampilan kolom tersebut pada formulir.

The screenshot displays a Jira task page for a task titled "#2157 Terdapat perbaikan field nama supplier". The task is assigned to Muhammad Royyan Saputra and is in a "CLOSED" state. The task description includes a list of steps and acceptance criteria. Two screenshots of a form are provided, with the "Nama Supplier" field highlighted in red in both. The first screenshot shows the form with a dropdown menu for "Nama Supplier" and a "Pilih Nama Supplier" label. The second screenshot shows the form with a text input field for "Nama Supplier" and a "Lakukan Pencarian" label. Below the screenshots is a "Custom Fields" section with various roles and their assigned users.

Task Details:

- Task ID:** #2157
- Title:** Terdapat perbaikan field nama supplier
- Assigned To:** Muhammad Royyan Saputra
- Created:** 18 Sep 2023 10:43

Acceptance Criteria:

- Field nama supplier pada membuat nota dinas dan lampirannya diambil dari nama supplier yang ada pada membuat kuitansi kontraktual
- field nama supplier dibuat disable sehingga tidak dapat diedit

Form Screenshot 1 (Form Kuitansi Kontraktual):

- Form Kuitansi Kontraktual
- Nomor Nota Dinas: BLSDM.3/PPK/KU.01.01/09/2023
- Nama Supplier:** (highlighted field with dropdown menu)
- Pejabat Pembuat Komitmen Puilftbong Aptika dan IKP
- Perihal: Paket Meeting Penyelenggaraan Rapat Koordinasi Penyusunan Program dan Evaluasi Kegiatan Puilftbong Aptika dan IKP
- Tanggal Surat: 18 September 2023
- Lampiran: 1 (satu) berkas
- Klasifikasi: Biasa

Form Screenshot 2 (Form Kuitansi Kontraktual):

- Form Kuitansi Kontraktual
- Judul Kuitansi* (required): Masukkan Judul Kuitansi
- Approval PBU* (required): Pilih Approval PBU
- Nama Supplier* (required):** (highlighted field)
- Lakukan Pencarian
- Submit

Custom Fields:

Analyst List Analyst	Muhammad Royyan Saputra
Programmer List Programmer	Imam Setiawan
Code Reviewer List Code Reviewer	
Tester List Tester	Firratu Saadah
Technical Writer List Technical Writer	
QA	
Mentee Analyst	
Mentee Tester	Michelle Angelica
Mentee Programmer	Syuhda Fakhrunnisa

Gambar 3.53 Task penyesuaian tampilan *field* ‘Nama Supplier’

Penulis juga mendapat *task* untuk menyesuaikan tampilan pada daftar tabel persetujuan. Seperti yang terlihat pada Gambar 3.55, penulis ditugaskan untuk memperbaiki tampilan tombol pada kolom aksi di halaman persetujuan.

#2234 Terdapat halaman list approval kabag keuangan

TASK

Edit with S2

This task belongs to #2159 [SP079] Sebagai Kabag Keuangan saya dapat melakukan approval & Reject SK

TASKBOARD

Add tag +

Created by Muhammad Royyan Saputra
22 Sep 2023 10:50

Skenario

1. Login as kabag keuangan
 1. kabag.keuangan@test.com / asdf1234
2. Klik menu persetujuan kabag keuangan
3. Menampilkan halaman list kabag keuangan

Acceptance Criteria

1. Menampilkan semua data yang ada pada surat keputusan
2. Ketika sudah melakukan approval kabag keuangan datanya tetap tampil pada halaman list kabag keuangan
3. Ketika sudah melakukan approval button aksi menjadi disable

Dashboard

Persetujuan SK

Surat Keputusan

Lakukan Pencarian

No	Nomor SK	Jenis SK	Perihal	Proses	Aksi
1	nomor sk	Sekali	deskripsi	Approval Kepala	Detail Approve Reject
2	1234	Sekali	Deskripsi	Approval Sekretaris badan	Detail Approve Reject
3	12333333	Sekali	Paket Meeting Penyelenggaraan Rapat Koordinasi Penyusunan Program dan Evaluasi Kegiatan Puslitbang Aptika dan IKP	Approval Sekretaris badan	Detail Approve Reject

Tugas dan tanggung jawab Tim sebagaimana tersebut pada diktum KESATU adalah sebagai berikut: 1. mempersiapkan pelaksanaan penerbitan Jurnal Elektronik/E-Journal Masyarakat Telekomunikasi Dan Informatika (JATI) 2. melaksanakan kegiatan

Custom Fields

Analyst
List Analyst: Muhammad Royyan Saputra

Programmer
List Programmer: Imam Setiawan

Code Reviewer
List Code Reviewer: Imam Setiawan

Tester
List Tester: Muhammad Royyan Saputra

Technical Writer
List Technical Writer:

QA:

Mentee Analyst:

Mentee Tester: Michelle Angelica

Mentee Programmer: Syuhda Fakhrunnisa

Gambar 3.55 *Task* penyesuaian tombol pada daftar persetujuan

Gambar 3.56 menunjukkan hasil implementasi di mana tampilan tombol pada kolom aksi telah disesuaikan, yaitu dengan menghilangkan tombol hapus dan melakukan *disable button* dalam kondisi tertentu.

The image displays a series of screenshots from a web application, illustrating the implementation of button adjustments in a list of approval requests.

Top Screenshot: Shows a user comment by Muhammad Royyan Saputra (10 Oct 2023 13:19) with the status "TESTED". Below the comment is a screenshot of a "Dashboard" page titled "Peretujuan Nota Dinas". The table lists 10 items with columns for "No", "Judul", "Nama Vendor", "Proses", and "Aksi". The "Aksi" column contains buttons for "Detail" (blue), "Cancel" (orange), and "Delete" (yellow).

Middle Screenshot: Shows a user comment by Syuhda Fakhrunnisa (09 Oct 2023 17:00) with the status "Edited: 09 Oct 2023 17:02 - View historic". The comment includes a GitHub merge request link and mentions "@imamst". Below the comment is a screenshot of a "Dashboard" page titled "Peretujuan SK". The table lists 8 items with columns for "No", "Nomor SK", "Jenis SK", "Perihal", "Proses", and "Aksi". The "Aksi" column contains buttons for "Detail" (blue), "Cancel" (orange), and "Delete" (yellow).

Bottom Screenshot: Shows a user comment by Muhammad Royyan Saputra (09 Oct 2023 15:01) mentioning "@syuhda.fakhrunnisa" and stating that the delete button has been removed. Below the comment is a screenshot of the "Peretujuan SK" table. In this view, the "Delete" buttons are highlighted with red boxes, indicating they have been disabled or removed.

Gambar 3.56 Hasil implementasi penyesuaian tombol pada daftar persetujuan

Selain itu, penulis juga diberikan tugas untuk memperbaiki tampilan *field* tanggal pada formulir nota dinas. Seperti yang terlihat pada Gambar 3.57, terdapat kebutuhan untuk menyesuaikan *field* tanggal pengadaan. Dengan penambahan *field* tersebut, data yang diisikan pada *field* juga harus dipastikan tersimpan di *database*, jika dilakukan *submit* atau *save as draft*.

#2281 Terdapat perbaikan field tanggal pengadaan form nota dinas

TASK

Edit with S2

This task belongs to #2156 [SP174] Sebagai PPK dapat membuat notas dinas

TASKBOARD

Add tag+

Created by Muhammad Royyan Saputra
06 Oct 2023 09:24

CLOSED

ASSIGNED

Firratu Saadah

+ Add assigned Assign to me

WATCHERS

Muhammad Royyan Saputra

Imam Setiawan

Firratu Saadah

Syuhda Fakhrunnisa

+ Add watchers Unwatch

User

ppk

Skenario

1. Login as ppk
2. Klik menu kuitansi kontraktual
3. Pilih proses membuat dokumen nota dinas
4. Klik button aksi

Database Yang digunakan

Waktu Mulai -> keuangan.nota_dinas.waktu_mulai_pelaksanaan

Waktu Selesai -> keuangan.nota_dinas.waktu_selesai_pelaksanaan

Acceptance Criteria

1. Terdapat penyesuaian form tanggal pengadaan
2. Dapat melakukan submit dokumen pbj dengan form terbaru

Perbaikan form

12 / Maret / 2023

Lampiran

Klasifikasi

Nama Kegiatan

Harga Perkiraan Sendiri (Rp)

Waktu Mulai Pelaksanaan Waktu Selesai Pelaksanaan

Lokasi

Save as Draft Selanjutnya

Custom Fields

Analyst List Analyst	Muhammad Royyan Saputra
Programmer List Programmer	Imam Setiawan
Code Reviewer List Code Reviewer	Imam Setiawan
Tester List Tester	Firratu Saadah
Technical Writer List Technical Writer	
QA	
Mentee Analyst	
Mentee Tester	
Mentee Programmer	Syuhda Fakhrunnisa

Gambar 3.57 Task perbaikan *field* tanggal pada *form*

Gambar 3.58 memperlihatkan hasil implementasi di mana tampilan *field* tanggal telah diperbaiki sesuai dengan standar yang ditentukan dan telah dipastikan bahwa data berhasil tersimpan ke *database* ketika melakukan *submit* atau *save as draft*.

The image shows a screenshot of a web application interface. At the top, there is a comment section with a user profile and a 'Done' button. Below this, there is a form titled 'Keuangan' with various fields. The form is displayed in a browser window with a dark sidebar on the left. The form fields include:

- Nomor Mendaftar: 026/PM/2023/13/10/2023
- Nama Regulan: PT. Cipta
- Kategori: Ragot
- Ragot: 1 hari
- Nama Regulan: Ragot
- Harga Parkiran Sendiri (Rp): 1.000.000
- Waktu Mulai Parkiran: 17 Oktober 2023
- Waktu Selesai Parkiran: 18 Oktober 2023
- Lokasi: lokasi

At the bottom of the form, there are two buttons: 'Save as draft' and 'Submit'.

Gambar 3.58 Hasil implementasi perbaikan *field* tanggal pada *form*

Penulis juga mendapat *task* untuk menambahkan fitur cetak pada tabel surat keputusan. Seperti yang terlihat pada Gambar 3.59, penulis ditugaskan untuk menambahkan tombol cetak pada kolom aksi di halaman surat keputusan.

The screenshot shows a Jira task titled "#2256 Terdapat tampilan button cetak surat keputusan -". The task is assigned to Muhammad Royyan Saputra and was created on 03 Oct 2023. The task description includes a scenario and current conditions. A table of 'Surat Keputusan' is displayed, with the 'Aksi' column containing a 'Print' button (represented by a printer icon) for each row. The 'Print' button for the first row is highlighted with a red box.

No	Nomor SK	Tanggal SK	Jenis SK	Perihal	Proses	Status	Aksi
1	NOMOR 34 TAHUN 2021	29 September 2023	Sekali	PENERBITAN JURNAL ELEKTRONIK MASYAR...	Merrevisi SK	Draft	Print, Refresh, Delete
2	NOMOR 34 TAHUN 2021	29 September 2023	Sekali	PENERBITAN JURNAL ELEKTRONIK MASYAR...	Merrevisi SK	Draft	Print, Refresh, Delete
3	NOMOR 34 TAHUN 2021	29 September 2023	Sekali	PENERBITAN JURNAL ELEKTRONIK MASYAR...	Merrevisi SK	Draft	Print, Refresh, Delete
4	NOMOR 34 TAHUN 2021	29 September 2023	Sekali	PENERBITAN JURNAL ELEKTRONIK MASYAR...	Merrevisi SK	Draft	Print, Refresh, Delete
5	NOMOR 34 TAHUN 2021	29 September 2023	Sekali	PENERBITAN JURNAL ELEKTRONIK MASYAR...	Merrevisi SK	Draft	Print, Refresh, Delete
6	nomor sk	27 September 2023	Sekali	perihal	Merrevisi SK	Draft	Print, Refresh, Delete
7	nomor	27 September 2023	Sekali	perihal	Finalisasi SK	Draft	Print, Refresh, Delete
8	nomor	27 September 2023	Sekali	perihal	Finalisasi SK	Draft	Print, Refresh, Delete
9	524	27 September 2023	Sekali	Sania Leonard	Approval Kibag Keuangan	Sedang Proses	Print, Refresh, Delete
10	123214	25 September 2023	Sekali	PERHAL	Approval PPK	Sedang Proses	Print, Refresh, Delete

The 'Acceptance Criteria' section states: "1. Terdapat penambahan button cetak pada list kolom aksi". Below this, a screenshot of a web page titled 'A Web Page' shows a table with the same structure as the one above, but with a 'Print' button (represented by a printer icon) in the 'Aksi' column for each row.

Gambar 3.59 *Task* penambahan tombol cetak pada kolom aksi tabel

Gambar 3.60 menunjukkan hasil implementasi di mana tombol cetak telah berhasil ditambahkan pada kolom aksi sesuai dengan ketentuan yang diminta dan secara *default*, tombol akan menjadi *disable*.

The screenshot displays a web application interface with a table of data. The table has the following columns: No, Nomor SK, Tanggal SK, Jenis SK, Perihal, Proses, Status, and Aksi. The data rows are as follows:

No	Nomor SK	Tanggal SK	Jenis SK	Perihal	Proses	Status	Aksi
1	NOMOR 34 TAHUN 2021	29 September 2023	Sekali	PENERBITAN JURNAL ELEKTRONIK MASYAR...	Membuat SK	Draft	[Print] [Refresh] [Delete] [Add]
2	NOMOR 34 TAHUN 2021	29 September 2023	Sekali	PENERBITAN JURNAL ELEKTRONIK MASYAR...	Membuat SK	Draft	[Print] [Refresh] [Delete] [Add]
3	NOMOR 34 TAHUN 2021	29 September 2023	Sekali	PENERBITAN JURNAL ELEKTRONIK MASYAR...	Membuat SK	Draft	[Print] [Refresh] [Delete] [Add]
4	NOMOR 34 TAHUN 2021	29 September 2023	Sekali	PENERBITAN JURNAL ELEKTRONIK MASYAR...	Membuat SK	Draft	[Print] [Refresh] [Delete] [Add]
5	NOMOR 34 TAHUN 2021	29 September 2023	Sekali	PENERBITAN JURNAL ELEKTRONIK MASYAR...	Membuat SK	Draft	[Print] [Refresh] [Delete] [Add]
6	nomor sk	27 September 2023	Sekali	perihal	Membuat SK	Draft	[Print] [Refresh] [Delete] [Add]
7	nomor	27 September 2023	Sekali	perihal	Membuat SK	Draft	[Print] [Refresh] [Delete] [Add]
8	nomor	27 September 2023	Sekali	perihal	Membuat SK	Draft	[Print] [Refresh] [Delete] [Add]
9	534	27 September 2023	Sekali	Sonia Leonard	Approval Kibang Keuangan	Sedang Proses	[Print] [Refresh] [Delete] [Add]
10	12324	26 September 2023	Sekali	PERHAL	Approval PPL	Sedang Proses	[Print] [Refresh] [Delete] [Add]

The interface also shows a sidebar menu with options like 'Keuangan', 'Surat Keputusan', and 'Master Supplier'. A comment section at the top shows a comment by Michelle Angelica dated 04 Oct 2023 08:20 with the text 'TESTED'. Below the table, there is a merge request link: 'MR: https://gitlab.javan.co.id/kominfo/keuangan/-/merge_requests/473' and a mention '@imamst'.

Gambar 3.60 Hasil implementasi penambahan tombol cetak pada kolom aksi tabel

Mengimplementasikan validasi input

Penulis mendapat *task* untuk mengimplementasikan beberapa validasi pada *form*. Seperti yang terlihat pada Gambar 3.61, penulis ditugaskan untuk menambahkan validasi pada kolom NPWP dan nomor rekening *supplier* agar hanya dapat diisi dengan karakter angka.

#2076 D0146 - PPK tidak dapat menginput karakter selain angka pada kolom NPWP dan Nomor Rekening Supplier pada Form tambah data Master Supplier

TASK
 Edit with S2
 This task belongs to #2124 [SP117] Sebagai ppk dapat melakukan penyesuaian edit & pencarian master supplier

TASKBOARD
 qatouch X s2 X low X Add tag+

Created by Muhammad Royyan Saputra
 11 Sep 2023 09:35

Siapa yang menemukan / melaporkan bug ini?
 Bramaria Azhra Sambada

User yang melakukan

Screenshot bug?

Link Screenshot Bug
 -

Bagaimana skenario hingga terjadi bug?

1. Membuka web Keuangan (keuangan.sdm.stag.sdmdigital.id)
2. Login sebagai PPK
3. Klik menu Master Supplier
4. Klik tombol Tambah Data
5. Mengisi form Supplier pada kolom Nomor Rekening Wajib Pajak dan Nomor Rekening Supplier dengan karakter selain angka

Apakah bug ini ini blocking?
 TIDAK

Apa pesan error?

URL Sentry/Postman?
 *

URL QA Touch?
<https://javan.qatouch.com/v2/defects/view/p/6p1a/did/RtqxD>

Kode QA Touch
 D0146

Module Name
 -

Issue Type
 -

Deskripsi
 PPK dapat menginput karakter selain angka pada kolom NPWP dan Nomor Rekening Supplier
 Link task taiga: <https://taiga.javan.id/project/kominfo-dignas-aplikasi-administrasi-keuangan-2022/task/1285>

Acceptance Criteria / Definition of Done?

1. Inputan pada field NPWP dan Nomor Rekening supplier dibikin hanya bisa input angka saja

ASSIGNED
 Muhammad Royyan Saputra
 + Add assigned Assign to me


WATCHERS
 Muhammad Royyan Saputra
 Imam Setiawan
 Michelle Angelica
 Syuhda Fakhrunnisa
 + Add watchers Unwatch

Gambar 3.61 *Task* validasi *input* angka pada *form*


Gambar 3.62 menunjukkan hasil implementasi di mana sistem berhasil memvalidasi masukan agar sesuai dengan ketentuan yang diberikan, yaitu hanya diizinkan meng-*input*-kan angka saja.


9 Comments 42 Activities Older first^

Type a new comment here

 **Muhammad Royyan Saputra** 06 Oct 2023 09:43

TESTED

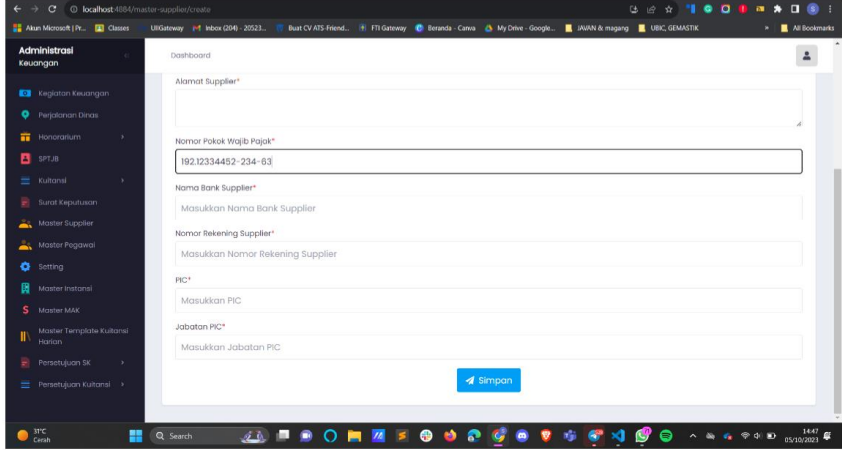



 **Syuhda Fakhrunnisa** 05 Oct 2023 14:47

MR: https://gitlab.javan.co.id/kominfo/keuangan/-/merge_requests/477

Mention: @imamst @royyansaputra @michelle.angelica

NPWP dapat input number dan karakter



 **Muhammad Royyan Saputra** 04 Oct 2023 10:28

pada form npwp dibikin dapat input number dan karakter ya mba @syuhda.fakhrunnisa minta tolong buat dicek

Gambar 3.62 Hasil implementasi validasi *input* angka pada *form*

Selain itu, penulis juga diberikan *task* untuk menambahkan validasi kelengkapan *form*. Seperti yang terlihat pada Gambar 3.63, sistem perlu menampilkan pesan *error* ketika pengguna mencoba menyimpan data tanpa mengisi seluruh kolom yang wajib diisi.

#2275 D0235 - PPK tidak dapat Submit data Nota Dinas ketika belum mengisi semua form

TASK

Edit with S2

This task belongs to #2156 [SP174] Sebagai PPK dapat membuat notas dinas

TASKBOARD

qatouch X S2 X high X Add tag+

Created by Muhammad Royyan Seputra 05 Oct 2023 14:35

Siapa yang menemukan / melaporkan bug ini?
Bramara Azhira Sambada

User yang melakukan

Screenshot bug?

Link Screenshot Bug
<https://javan.qatouch.com/issue/attachments/p/6p1a/f/RqJEW>

Bagaimana skenario hingga terjadi bug?

1. Membuka web Administrasi Keuangan (keuangan.odm.tag.sdm.digitaLid)
2. Login sebagai PPK
3. Klik menu Kuitansi
4. Klik sub menu Kontraktual
5. Klik tombol aksi warna kuning pada data dengan Proses 'Membuat Nota Dinas dan Lampirannya'
6. Klik tombol Submit

Apakah bug ini ini blocking?
TIDAK

Apa pesan error?

URL Sentry/Postman?
-

URL QA Touch?
<https://javan.qatouch.com/v2/defects/view/6p1a/did/RqJEW>

Kode QA Touch
D0235

Module Name
-

Issue Type
-

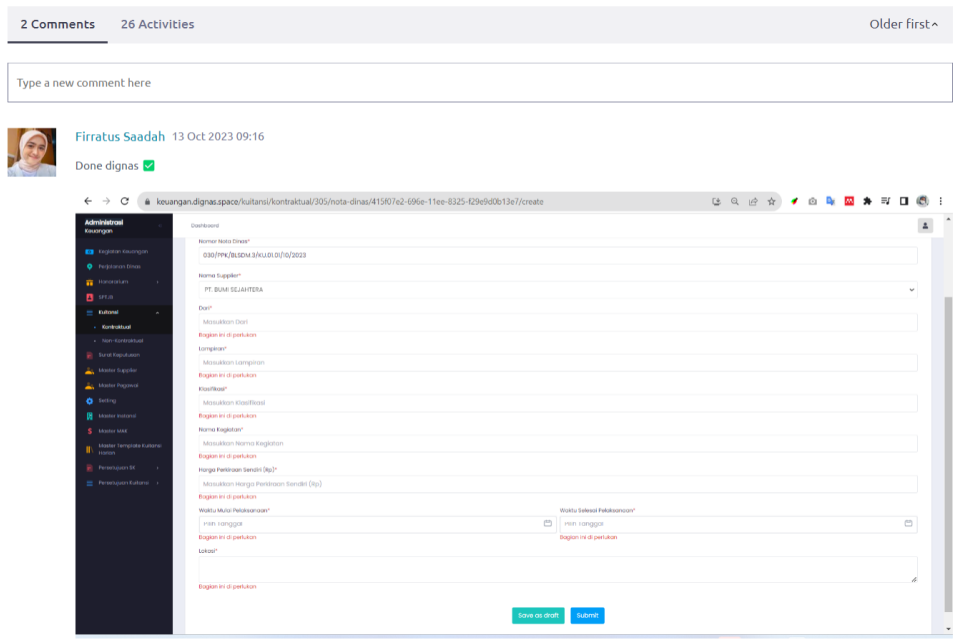
Deskripsi
Kondisi saat ini, tidak ada satupun kolom mandatory pada form Membuat Nota Dinas sehingga dapat submit form kosong. Seharusnya terdapat kolom mandatory sehingga tidak bisa submit Form yang belum diisi.

Acceptance Criteria / Definition of Done?

1. Terdapat label mandatory pada semua field membuat nota dinas
2. Terdapat alert "bagian ini diperlukan" jika belum mengisi field nya

Gambar 3.63 *Task* validasi kelengkapan *form* nota dinas

Gambar 3.64 memperlihatkan hasil implementasi di mana sistem berhasil menampilkan pesan validasi sesuai dengan kebutuhan, di mana jika pengguna belum mengisi kolom yang *mandatory*, maka akan diberikan validasi “Bagian ini diperlukan”.



2 Comments 26 Activities Older first

Type a new comment here

Firratu Saadah 13 Oct 2023 09:16
Done dignas ✓

keuangan.dignas.space/kuitansi/kontraktual/305/nota-dinas/4150742-696e-11ee-8325-429e90b13a77/create

Administrasi Keuangan

Dashboard

Nomor Nota Dinas*
030/PM/BLSDM/3/KU.01.01/0/2023

Nama Suplier*
PT. BUMI SELAJITERA

Dit*
Masukkan Dari
Bagian ini diperlukan

Lampiran*
Masukkan Lampiran
Bagian ini diperlukan

Klasifikasi*
Masukkan Klasifikasi
Bagian ini diperlukan

Nama Kegiatan*
Masukkan Nama Kegiatan
Bagian ini diperlukan

Harga Perkiraan Sendiri (Rp)*
Masukkan Harga Perkiraan Sendiri (Rp)
Bagian ini diperlukan

Waktu Mulai Pelaksanaan*
masi tanggal
Bagian ini diperlukan

Waktu Selesai Pelaksanaan*
masi tanggal
Bagian ini diperlukan

Lokasi*
Bagian ini diperlukan

Save as draft Submit

mas @imamst tolong info kalo udah naik ke staging ya, belum bisa test ke staging soalnya issue yg auto reload

Syuhda Fakhrunnisa 09 Oct 2023 11:53
MR: https://gitlab.javan.co.id/kominfo/keuangan/-/merge_requests/484
Mention: @imamst

localhost:4834/kuitansi/kontraktual/253/nota-dinas/60416812-32ad-11ee-9d9a-429e90b13a77/create

Administrasi Keuangan

Dashboard

Dari*
Masukkan Dari
Bagian ini diperlukan

Lampiran*
Masukkan Lampiran
Bagian ini diperlukan

Klasifikasi*
Masukkan Klasifikasi
Bagian ini diperlukan

Nama Kegiatan*
Masukkan Nama Kegiatan
Bagian ini diperlukan

Harga Perkiraan Sendiri (Rp)*
Masukkan Harga Perkiraan Sendiri (Rp)
Bagian ini diperlukan

Lokasi*
Masukkan Lokasi
Bagian ini diperlukan

Gambar 3.64 Hasil implementasi validasi kelengkapan *form* nota dinas

Penulis juga mendapat *task* untuk menambahkan validasi pada *form* honor narasumber. Seperti yang terlihat pada Gambar 3.65, penulis ditugaskan untuk mengimplementasikan pesan *alert* ketika pengguna belum mengisi kolom-kolom yang bersifat *mandatory* pada saat mengedit data honor narasumber.

#2296 Terdapat penyesuaian edit honor narasumber

TASK

Edit with S2

This task belongs to #2038 [SP079] Sebagai ppk dapat membuat honor narasumber dan edit narasumber

TASKBOARD

Add tag+

Created by Muhammad Royyan Saputra
12 Oct 2023 13:14

User

ppk

Skenario

1. Login as ppk
2. Klik menu honorarium
3. Pilih honor narasumber
4. Klik button edit
5. Edit data

Kondisi Saat ini

Belum menampilkan alert mandatory pada saat melakukan edit honor narasumber

Input Data Honor Narasumber

Nomor Surat Undangan WM-100-1023

	Nomor Kuitansi	Tanggal Kuitansi	Penerima	Deskripsi	Rincian	MAK	Jumlah (RP)	PPH (RP)	PPN (RP)
1	004/KH/09/2023	Invalid date	JULAHA			PPH MAK			
2	005/KH/09/2023	Invalid date	HOTIEH			PPH MAK			
3	023/KH/09/2023	29 September 2023	JULAHA	Honor Narasumber Kegiatan	rincian	6546464	120000	0	0
4	024/KH/09/2023	29 September 2023	HOTIEH	Honor Narasumber Kegiatan	rincian	6546464	120000	0	0

Kembali Ubah

Acceptance Criteria

1. Terdapat penyesuaian field mandatory yaitu pada kolom
 1. Nomor Kuitansi
 2. Tanggal Kuitansi
 3. Penerima
 4. Deskripsi
 5. Rincian
 6. MAK
2. Yang Tidak Mandatory
 1. Jumlah
 2. PPH
 3. PPN
3. Ketika melakukan edit honorarium narasumber data group bisa melakukan ubah tanpa harus mengisi field yang tidak mandatory
4. Terdapat alert untuk mengisi field yang mandatory

Custom Fields

Analyst: Muhammad Royyan Saputra

Programmer: [Dropdown]

Code Reviewer: Imam Setiawan

Tester: Muhammad Royyan Saputra

Technical Writer: [Dropdown]

QA: [Dropdown]

Mentee Analyst: [Dropdown]

Mentee Tester: [Dropdown]

Mentee Programmer: Syuhda Fakhrunnisa

Gambar 3.65 Task validasi kolom *mandatory* pada *form* honor narasumber

Gambar 3.66 menunjukkan hasil implementasi di mana sistem berhasil menampilkan pesan *alert* peringatan ketika terdapat kolom *mandatory* yang belum diisi.

Melakukan *Searching*, *Sorting*, dan *Filtering* Data

Penulis mendapat *task* untuk mengimplementasikan fitur *searching*, *sorting*, dan *filtering* data. Seperti yang terlihat pada Gambar 3.67, penulis ditugaskan untuk menambahkan fitur *searching* berdasarkan nama instansi pada halaman Master Instansi.

#2112 D0182 - PPK dapat filtering Instansi berdasarkan Nama Instansi (Badan Pengembangan Sumber Daya Manusia Komunikasi dan Informatika)

TASK

Edit with S2

This task belongs to #2129 [SP065] Sebagai PPK dapat melakukan penyesuaian filter master instansi

TASKBOARD

Created by Muhammad Royyan Saputra 12 Sep 2023 15:01

Siapa yang menemukan / melaporkan bug ini?

Bramara Azhra Sambada

User yang melakukan

PPK

Screenshot bug?

Link Screenshot Bug

<https://javan.qatouch.com/issue/attachments/p/6p1a/vwy51>

Bagaimana skenario hingga terjadi bug?

1. Membuka web Keuangan (keuangan.sdm.stag.sdmdigital.id)
2. Login sebagai PPK
3. Klik menu Master Instansi
4. Klik tombol Filter pada bagian kanan atas
5. Mengisi Form Filter kolom Nama Instansi dengan (Badan Pengembangan Sumber Daya Manusia Komunikasi dan Informatika)
6. Klik tombol Simpan pada Form Filter

Apakah bug ini blocking?

TIDAK

Apa pesan error?

URL Sentry/Postman?

*

URL QA Touch?

<https://javan.qatouch.com/v2/defects/view/p/6p1a/did/vwy51>

Kode QA Touch

D0182

Module Name

-

Issue Type

-

Deskripsi

Halaman menampilkan "Data tidak dapat ditemukan", instansi yang dicari ada pada halaman Master Instansi

Acceptance Criteria / Definition of Done?

1. Dapat melakukan pencarian pada master instansi berdasarkan nama instansi

Gambar 3.67 *Task* implementasi *searching* data instansi

Gambar 3.68 menunjukkan hasil implementasi di mana sistem berhasil melakukan *searching* data sesuai dengan nama instansi yang dimasukkan.

The image displays a social media thread with the following content:

- Comment 1:** By **Huhammad Royyan Saputra** on 16 Oct 2023 10:23. Status: **TESTED**. Includes a video showing a search filter modal in a web application. The modal has a search input for 'Kode Instansi' and a list of results: 04, 07, 08, 09.
- Comment 2:** By **Imam Setiawan** on 13 Oct 2023 14:05. Text: "updated staging".
- Comment 3:** By **Syuhda Fakhrunnisa** on 13 Oct 2023 09:18. Includes a GitHub link: https://gitlab.javan.co.id/kominfo/keuangan/-/merge_requests/506. Mentions: @imamst. Text: "kode instansi dan nama instansi filternya dibuat dropdown dan bisa search". Includes a video showing the search filter modal with the search input set to '04' and results: 03, 04, 05.
- Text:** "Dapat melakukan pencarian pada master instansi berdasarkan kode dan nama instansi".
- Comment 4:** By **Syuhda Fakhrunnisa** on 12 Oct 2023 16:05. Edited: 12 Oct 2023 16:06. Mentions: @royyansaputra. Text: "oww okeee baiklahh".
- Comment 5:** By **Huhammad Royyan Saputra** on 12 Oct 2023 16:02. Text: "oh ya mbak @syuhda.fakhrunnisa aku lupa menambahkan dod nya untuk tampilan filternya dibikin kek gini ya yang sama di kegiatan keuangan". Includes a video showing a search filter modal for 'Kegiatan Keuangan' with fields for 'Type', 'Kategori', and 'Proses'.

Gambar 3.68 Hasil implementasi *searching* data instansi

Selain itu, penulis juga diberikan *task* untuk mengimplementasikan *sorting* data dokumen. Seperti yang terlihat pada Gambar 3.69, terdapat kebutuhan untuk mengurutkan data pada halaman Dokumen PBJ.

#2136 Terdapat tampilan list proses membuat dokumen PBJ

TASK

Edit with S2

This task belongs to #2134 [SP103] Sebagai PBJ dapat membuat dokumen PBJ

TASKBOARD

Add tag +

Created by Muhammad Royyan Saputra
14 Sep 2023 14:34

Skenario

1. Login as PBJ
 1. `keuangan_pbj@test.com / asdf1234`
2. Klik Dokumen PBJ
3. Menampilkan halaman list dokumen PBJ

API Yang Digunakan

<https://javan-diskusi.postman.co/workspace/Dignas-Kominfo-a26a7175-30f8-4f35-bb3f-df07bb014fa0/request/15618906-f483b4bf-1c8a-4bfd-a35c-eb9e6f0fad5a>

Acceptance Criteria

1. Terdapat tampilan menu dokumen PBJ dan halaman list dokumen PBJ Sebagai berikut
2. Terdapat nama menu Persetujuan Kuitansi dan Dokumen PBJ
3. Terdapat list pada menu Dokumen PBJ sebagai berikut
4. List yang ditampilkan sebagai berikut
5. List data pada dokumen pbj dibikin descending data yang terbaru dimunculkan diawal

Filter

Proses

Lakukan Pencarian

- Membuat Nota Dinas
- Finalisasi Nota Dinas
- Melakukan Approval Dinas
- Membuat Dokumen PBJ
- Finalisasi Dokumen PBJ
- Melakukan Approval Dokumen PBJ
- Membuat Dokumen PPK
- Finalisasi Dokumen PPK
- Upload Dokumen Kontraktual

Dokumen PBJ

No	MDJ	Nama	WJSD	Proses	Aksi
1	Automa generate	PT Bumi Sejahtera		Membuat Dokumen-Dokumen PBJ	

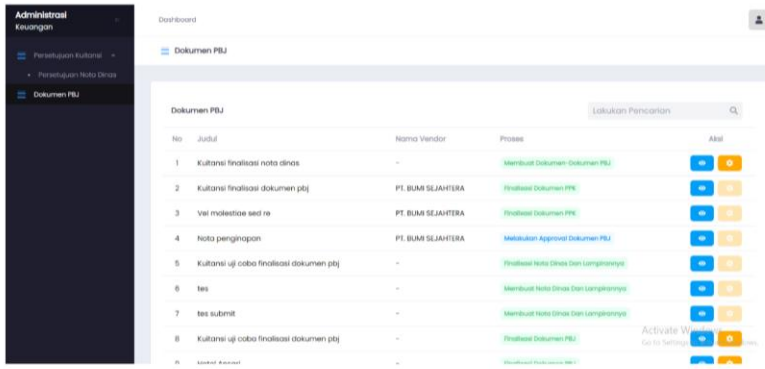
Gambar 3.69 *Task* sorting data Dokumen PBJ

Gambar 3.70 memperlihatkan hasil implementasi di mana data telah berhasil diurutkan sesuai dengan ketentuan yang diberikan, yaitu secara *descending* berdasarkan tanggal terbaru.

9 Comments 56 Activities Older first

Type a new comment here

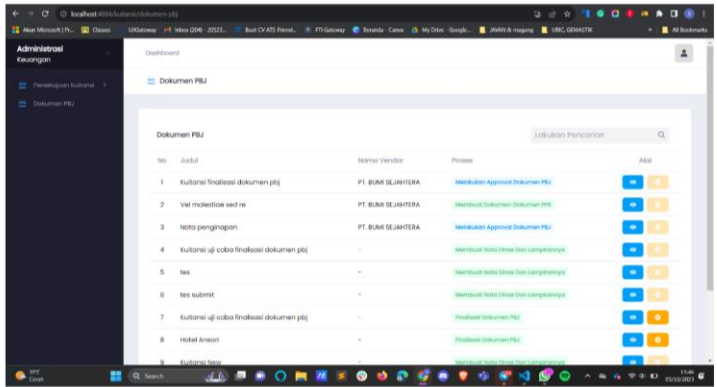
Muhammad Royyan Saputra 06 Oct 2023 10:20
TESTED



No	Judul	Nama Vendor	Proses	Aksi
1	Kultansi finalisasi nota dinas	-	Membuat Dokumen Dokumen PBJ	← →
2	Kultansi finalisasi dokumen pbj	PT. BUMI SEJAHTERA	Finalisasi Dokumen PBJ	← →
3	Vali molesitae sed re	PT. BUMI SEJAHTERA	Finalisasi Dokumen PBJ	← →
4	Noto pengirapan	PT. BUMI SEJAHTERA	Melakukan Approval Dokumen PBJ	← →
5	Kultansi uji coba finalisasi dokumen pbj	-	Finalisasi Nota Dinas Dan Lengkapnya	← →
6	tes	-	Membuat Nota Dinas Dan Lengkapnya	← →
7	tes submit	-	Membuat Nota Dinas Dan Lengkapnya	← →
8	Kultansi uji coba finalisasi dokumen pbj	-	Finalisasi Dokumen PBJ	← →

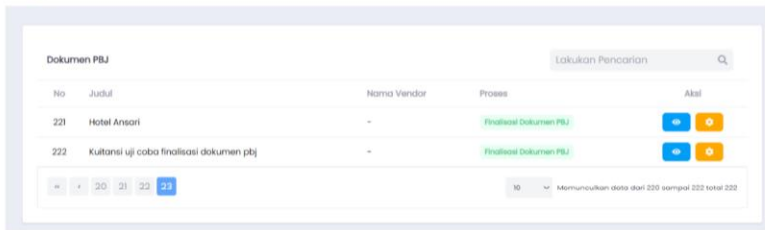
Syuhda Fakhrunnisa 05 Oct 2023 15:47
MR: https://gitlab.javan.co.id/kominfo/keuangan-/merge_requests/478 | Mention: @imamat
done ya mas @royyansaputra

List data descending/ yang terbaru



No	Judul	Nama Vendor	Proses	Aksi
1	Kultansi finalisasi dokumen pbj	PT. BUMI SEJAHTERA	Melakukan Approval Dokumen PBJ	← →
2	Vali molesitae sed re	PT. BUMI SEJAHTERA	Membuat Dokumen Dokumen PBJ	← →
3	Noto pengirapan	PT. BUMI SEJAHTERA	Melakukan Approval Dokumen PBJ	← →
4	kultansi uji coba finalisasi dokumen pbj	-	Membuat Nota Dinas Dan Lengkapnya	← →
5	tes	-	Membuat Nota Dinas Dan Lengkapnya	← →
6	tes submit	-	Membuat Nota Dinas Dan Lengkapnya	← →
7	Kultansi uji coba finalisasi dokumen pbj	-	Finalisasi Dokumen PBJ	← →
8	Hotel Ansari	-	Finalisasi Dokumen PBJ	← →

Muhammad Royyan Saputra 04 Oct 2023 09:41
mbak @syuhda.fakhrunnisa untuk list ini minta tolong dibikin descending yaa list datanya karena untuk sekarang masih belum descending dan ketika approval nota dinas datanya akan masuk ke sub menu dokumen pbj dan data nya masuk di list akhir



No	Judul	Nama Vendor	Proses	Aksi
221	Hotel Ansari	-	Finalisasi Dokumen PBJ	← →
222	Kultansi uji coba finalisasi dokumen pbj	-	Finalisasi Dokumen PBJ	← →

Gambar 3.70 Hasil implementasi *sorting* data Dokumen PBJ

Penulis juga mendapat *task* untuk mengimplementasikan fitur *filtering* berdasarkan proses pada dokumen PBJ. Seperti yang terlihat pada Gambar 3.71, penulis ditugaskan untuk menambahkan fitur *filtering* berdasarkan status proses dokumen.

#2265 Terdapat filter proses pada halaman dokumen pbj

TASK

Edit with S2

This task belongs to #2134 [SP103] Sebagai PBJ dapat membuat dokumen PBJ

TASKBOARD

Add tag +

Created by Muhammad Royyan Saputra
04 Oct 2023 10:45

User

PBJ

Skenario

- Login as pbj
1. keuangan_pbj@test.com / asdf1234
- Klik menu dokumen pbj
- Klik button filter
- Menampilkan filter proses yang sudah dipilih

API Yang digunakan

<https://api.dignas.space/docs/api#/paths/keuangan-master-receipts/get>

Acceptance Criteria

- Terdapat button filter pada halaman persetujuan nota dinas
- Dapat menampilkan hasil proses yang sudah dipilih

Custom Fields

Analyst
List Analyst: Muhammad Royyan Saputra

Programmer
List Programmer: Imam Setiawan

Code Reviewer
List Code Reviewer: Imam Setiawan

Tester
List Tester: Muhammad Royyan Saputra

Technical Writer
List Technical Writer: [Dropdown]

QA
[Dropdown]

Mentee Analyst
[Dropdown]

Mentee Tester
Michelle Angelica

Mentee Programmer
Syuhda Fakhrunnisa

Gambar 3.71 *Task* implementasi *filtering* berdasarkan proses dokumen

Gambar 3.72 menunjukkan hasil implementasi di mana sistem berhasil melakukan *filtering* data sesuai dengan proses yang dipilih.

The image displays a web application interface for document management. The top section shows a comment by Michelle Angelica on 06 Oct 2023 13:06, with the status 'TESTED'. Below this is a dashboard for 'Administrasi Keuangan' with a sidebar menu containing 'Peretujuan Kultansi', 'Peretujuan Nota Dinas', and 'Dokumen PBU'. The main content area is titled 'Peretujuan Nota Dinas' and contains a table with the following data:

No	Judul	Nama Vendor	Proses	Aksi
1	kultansi finalisasi nota dinas	PT. Cipto	Membuat Dokumen-Dokumen PBU	[Left Arrow] [Right Arrow]
2	Kultansi Kontraktul uji	PT. BUMI SEJAHTERA	Membuat Dokumen-Dokumen PBU	[Left Arrow] [Right Arrow]
3	kultansi generate	PT. BUMI SEJAHTERA	Membuat Dokumen-Dokumen PBU	[Left Arrow] [Right Arrow]
4	Kultansi nota dinas	PT. BUMI SEJAHTERA	Membuat Dokumen-Dokumen PBU	[Left Arrow] [Right Arrow]
5	Sample title	-	Membuat Dokumen-Dokumen PBU	[Left Arrow] [Right Arrow]
6	tes kultansi	-	Membuat Dokumen-Dokumen PBU	[Left Arrow] [Right Arrow]
7	kultansi Kontrak 1	-	Membuat Dokumen-Dokumen PBU	[Left Arrow] [Right Arrow]

The bottom section shows a screenshot of the same application with a 'Filter' dialog box open. The dialog box has a 'Proses' dropdown menu and a list of document entries. The entries in the dialog box are:

No	Judul	Nama Vendor	Proses	Aksi
1	test03	-	Membuat Dokumen PBU	[Left Arrow] [Right Arrow]
2	kultansi finalisasi nota	PT. BUMI SEJAHTERA	Membuat Dokumen PBU	[Left Arrow] [Right Arrow]
3	Kultansi finalisasi dok	PT. BUMI SEJAHTERA	Membuat Dokumen PBU	[Left Arrow] [Right Arrow]
4	Val molesitose sad re	-	Membuat Dokumen PBU	[Left Arrow] [Right Arrow]
5	Nota penginapan	PT. BUMI SEJAHTERA	Membuat Dokumen PBU	[Left Arrow] [Right Arrow]
6	Kultansi uji coba finalisasi dokumen pibj	-	Membuat Dokumen PBU	[Left Arrow] [Right Arrow]
7	tes	-	Membuat Dokumen PBU	[Left Arrow] [Right Arrow]
8	tes submit	-	Membuat Dokumen PBU	[Left Arrow] [Right Arrow]
9	Kultansi uji coba finalisasi dokumen pibj	-	Membuat Dokumen PBU	[Left Arrow] [Right Arrow]

The bottom part of the image shows the application after filtering, displaying a list of documents with the following data:

No	Judul	Nama Vendor	Proses	Aksi
1	test03	PT. BUMI SEJAHTERA	Finalisasi Dokumen PBU	[Left Arrow] [Right Arrow]
2	kultansi finalisasi dokumen pibj	PT. BUMI SEJAHTERA	Finalisasi Dokumen PBU	[Left Arrow] [Right Arrow]
3	Val molesitose sad re	PT. BUMI SEJAHTERA	Finalisasi Dokumen PBU	[Left Arrow] [Right Arrow]
4	Kultansi Kontraktual	-	Finalisasi Dokumen PBU	[Left Arrow] [Right Arrow]
5	pibj test	-	Finalisasi Dokumen PBU	[Left Arrow] [Right Arrow]
6	Sample title	PT. Cipto	Finalisasi Dokumen PBU	[Left Arrow] [Right Arrow]
7	Kultansi Perijabatan dinas meeting	PT. BUMI SEJAHTERA	Finalisasi Dokumen PBU	[Left Arrow] [Right Arrow]

Gambar 3.72 Hasil implementasi *filtering* berdasarkan proses dokumen

Selanjutnya, penulis diberikan tugas untuk mengimplementasikan *filtering* data pegawai. Seperti yang terlihat pada Gambar 3.73, terdapat kebutuhan untuk *filter* data berdasarkan beberapa kriteria seperti tipe identitas, jabatan, nama instansi, dan nama penyelenggara.

#2292 Dapat melakukan filter master pegawai

TASK

Edit with S2

This task belongs to #1846 [SP014] Sebagai PPK dapat melakukan search, filter, pagination dan sorting pada master pegawai

TASKBOARD

Add tag +

Created by Muhammad Royyan Saputra
12 Oct 2023 10:48

User

PPK

Skenario

1. Login as ppk
2. Klik menu master pegawai
3. Menampilkan master pegawai
4. Klik button filter master pegawai

Database Yang Digunakan

Tipe identitas → keuangan.public.tipe_identitas

jabatan → keuangan.public.m_jabatan

nama instansi → keuangan.public.m_instansi

nama penyelenggara → keuangan.public.m_penyelenggara

tahun jabatan → keuangan.public.jabatan_pegawai.tahun_jabatan

Kondisi Saat ini

Belum bisa melakukan pada semua kolom berikut

Tipe identitas , jabatan, nama instansi dan nama penyelenggara, tahun jabatan

Nomor identitas
Masukkan Nomor Identitas

Tipe Identitas
Masukkan Tipe Identitas

Nama
Masukkan Nama

Jabatan
Pilih Jabatan

Tahun Jabatan
Masukkan Tahun Jabatan

Pangkat
Masukkan Pangkat

Golongan
Masukkan Golongan

Eselon
Masukkan Eselon

Nama Instansi
Pilih Nama Instansi

Nama Penyelenggara
Pilih Nama Penyelenggara

Haous Filter

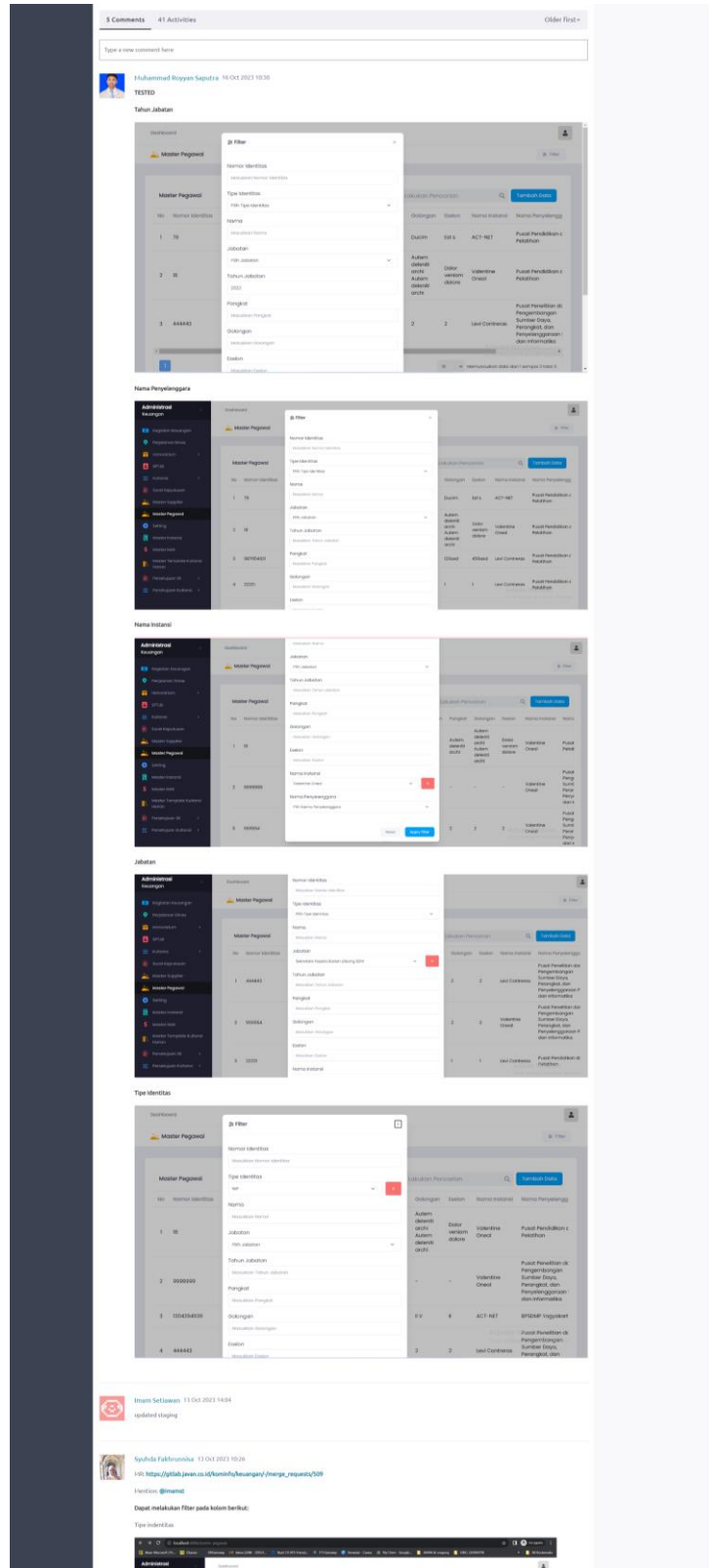
Simpan

Acceptance Criteria

1. Dapat melakukan filter pada kolom berikut
 1. Tipe identitas
 2. jabatan
 3. nama instansi
 4. nama penyelenggara
 5. tahun jabatan

Gambar 3.73 Task implementasi *filtering* data pegawai

Gambar 3.74 memperlihatkan hasil implementasi di mana sistem berhasil melakukan *filtering* data pegawai sesuai dengan kriteria yang dimasukkan, ke dalam *field-field* yang disediakan pada *form filter*.



Gambar 3.74 Hasil implementasi *filtering* data pegawai

Melakukan Simpan Data

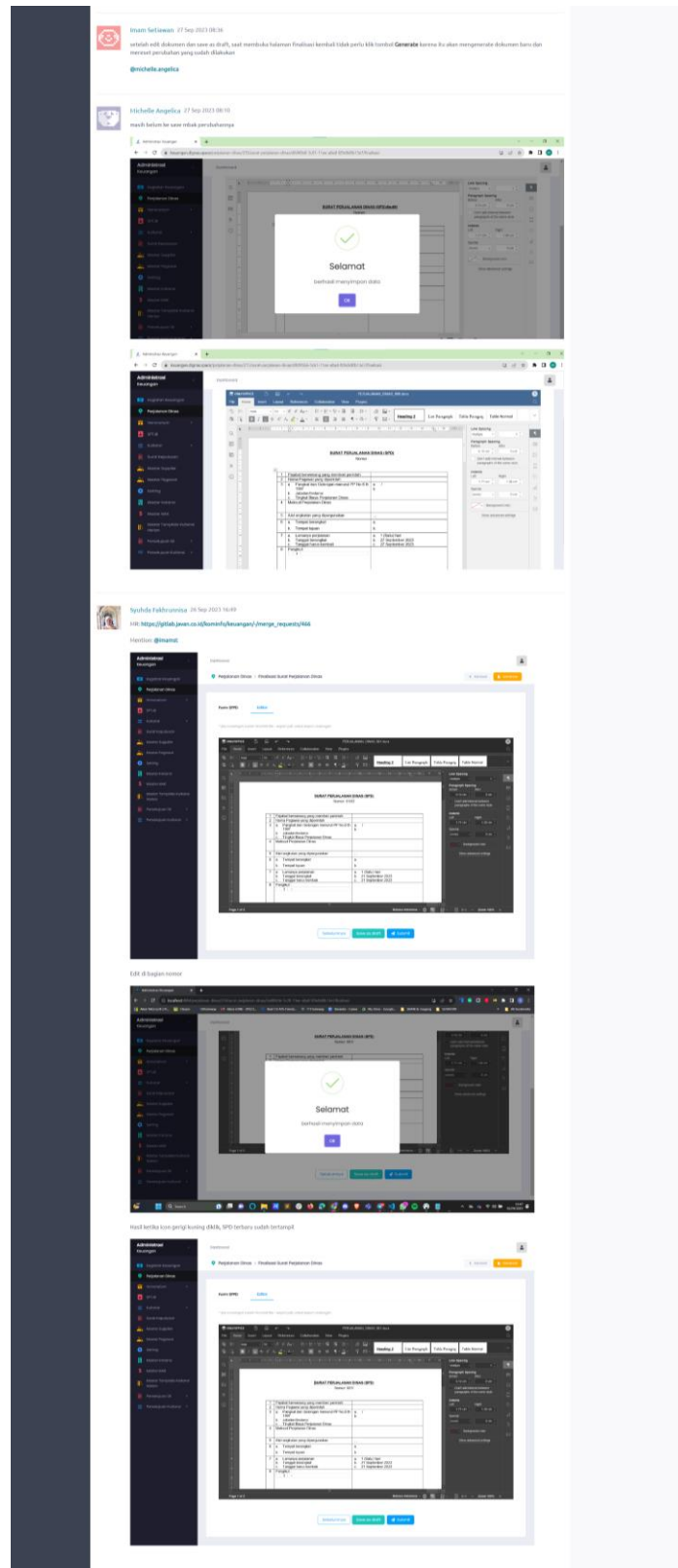
Penulis mendapat *task* untuk mengimplementasikan fitur penyimpanan data *draft* pada beberapa *form*. Seperti yang terlihat pada Gambar 3.75, penulis ditugaskan untuk menambahkan fitur *save draft* pada *form* finalisasi Surat Perjalanan Dinas.

The screenshot shows a Jira task page with the following details:

- Task ID:** #2219 D0151 - PPK dapat save draft finalisasi Surat Perjalanan Dinas
- Task Description:** Edit with S2. This task belongs to #1930 [SP079] Sebagai PPK dapat mengakses penyesuaian menu perjalanan dinas.
- Taskboard:** qatouch X, s2 X, high X, Add tag +
- Created by:** Muhammad Royyan Saputra, 21 Sep 2023 14:31
- Status:** CLOSED
- Assigned to:** Muhammad Royyan Saputra
- Watchers:** Imam Setiawan, Michelle Angelica, Syuhda Fakhrunnisa
- Issue Description:**
 - Siapa yang menemukan / melaporkan bug ini? Bramara Azhra Sambada
 - User yang melakukan: PPK
 - Screenshot bug? A screenshot of the 'Form SPPD' editor showing a red box around the text 'SURAT PERJALANAN DINAS (BPK) Draft'.
- Link Screenshot Bug:** <https://javan.qatouch.com/issue/attachments/p/6p1a/482wb>
- Bagaimana skenario hingga terjadi bug?**
 - Membuka web Keuangan (keuangan.sdm.stag.sdmdigital.id)
 - Login sebagai PPK
 - Klik menu Perjalanan Dinas
 - Klik ikon aksi warna kuning pada data dengan status Proses "Melakukan Finalisasi Surat Perjalanan Dinas"
 - Klik tombol Generate
 - Mengedit surat Perjalanan Dinas sesuai kebutuhan
 - Klik tombol Save as draft
 - Klik tombol OK pada pop up notifikasi
 - Klik ikon aksi warna kuning pada data dengan status Proses "Melakukan Finalisasi Surat Perjalanan Dinas"
 - Klik tombol Generate (mengenerate ulang yang menandakan bahwa perubahan sebelumnya tidak tersimpan)
- Apakah bug ini blocking?** TIDAK
- Apa pesan error?**
- URL Sentry/Postman?**
- URL QA Touch?** <https://javan.qatouch.com/v2/defects/view/p/6p1a/did/482wb>
- Kode QA Touch:** D0151
- Module Name:** -
- Issue Type:** -
- Deskripsi:** Saat melakukan save draft, ketika dibuka kembali data surat perjalanan dinas, perubahan yang dibuat sebelumnya tidak tampil pada file surat perjalanan dinas
- Acceptance Criteria / Definition of Done?** Dapat menyimpan perubahan dilakukan pada surat Perjalanan Dinas dengan status proses "Melakukan Finalisasi Surat Perjalanan Dinas"

Gambar 3.75 Task save draft finalisasi Surat Perjalanan Dinas

Gambar 3.76 menunjukkan hasil implementasi di mana sistem berhasil menyimpan data *draft*, dan notifikasi sukses akan ditampilkan ketika data berhasil disimpan.



Gambar 3.76 Hasil implementasi *save draft* finalisasi Surat Perjalanan Dinas

Selain itu, penulis juga diberikan *task* untuk menambahkan *redirect* halaman setelah menyimpan data. Seperti yang terlihat pada Gambar 3.77, setelah data berhasil disimpan, sistem perlu *redirect* pengguna ke halaman awal.

#2271 D0231 - PPK dapat melihat halaman Kuitansi Kontraktual setelah melakukan Save Draft data "Membuat Nota Dinas"

TASK

Edit with S2

This task belongs to #2156 [SP174] Sebagai PPK dapat membuat notas dinas

TASKBOARD

qatouch X S2 X low X Add tag+

Created by Muhammad Royyan Saputra
05 Oct 2023 14:35

Siapa yang menemukan / melaporkan bug ini?
Bramara Azhra Sambada

User yang melakukan

Screenshot bug?

Link Screenshot Bug

<https://javan.qatouch.com/issue/attachments/p/6p1af/VG9wQ>

Bagaimana skenario hingga terjadi bug?

1. Membuka web Administrasi Keuangan (keuangan.sdm.stag.sdmdigital.id)
2. Login sebagai PPK
3. Klik menu Kuitansi
4. Klik sub menu Kontraktual
5. Klik tombol aksi warna kuning pada data dengan Proses 'Membuat Nota Dinas dan Lampirannya'
6. Mengisi form sesuai kebutuhan
7. Klik tombol Save as draft
8. Klik OK pada pop up notifikasi

Apakah bug ini blocking?
TIDAK

Apa pesan error?

URL Sentry/Postman?

*

URL QA Touch?

<https://javan.qatouch.com/v2/defects/view/p/6p1af/did/VG9wQ>

Kode QA Touch

D0231

Module Name

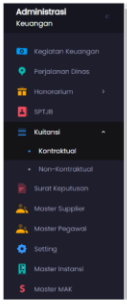
-

Issue Type

-

Deskripsi

Kondisi saat ini, ketika sudah melakukan save draft, diarahkan ke halaman



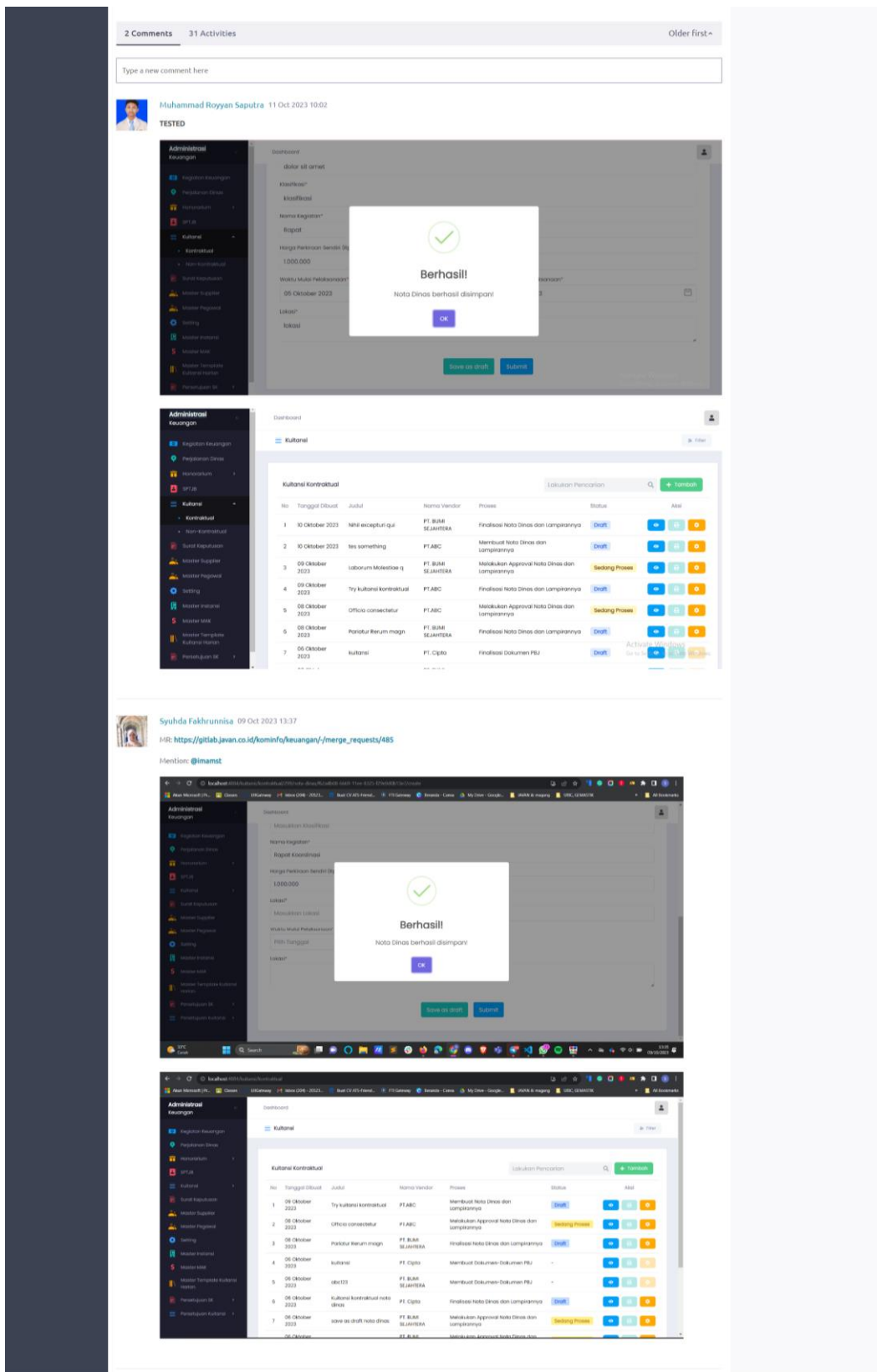
dengan uri <https://keuangan.sdm.stag.sdmdigital.id/kuitansi> yang tidak menampilkan data apapun. Seharusnya <https://keuangan.sdm.stag.sdmdigital.id/kuitansi/kontraktual>

Acceptance Criteria / Definition of Done?

1. Ketika melakukan save draft nota dinas diarahkan kehalaman list list kontraktual

Gambar 3.77 Task save draft dan redirect halaman

Gambar 3.78 memperlihatkan hasil implementasi di mana sistem berhasil menyimpan data dan *redirect* pengguna ke halaman yang telah ditentukan.



Gambar 3.78 Hasil implementasi *save draft* dan *redirect* halaman

Pengujian oleh Quality Assurance (QA)

Setelah suatu *task* memiliki status *closed*, pengujian akan dilakukan oleh tim *Quality Assurance* (QA). QA menggunakan beberapa *tools*, termasuk QA Touch, yang merupakan platform manajemen pengujian komprehensif dari DCKAP. Alat ini dirancang untuk menangani berbagai aspek pengujian proyek secara efisien, mulai dari perencanaan, pelaksanaan, pelacakan *bug*, hingga pelaporan status aktivitas QA. QA Touch digunakan di berbagai industri, seperti perangkat lunak, ritel, *fintech*, dan lainnya, dengan solusi manajemen pengujian yang disesuaikan untuk kebutuhan spesifik (Touch, 2024).

Proses pengujian diawali dengan penyusunan daftar *test case* oleh QA berdasarkan *acceptance criteria* dari *task* yang telah dirancang dan mendokumentasikannya melalui QA Touch. Selanjutnya, *system analyst* akan membuat dan mendokumentasikan *issue* yang terkait dengan *test case* dan *task* tersebut. *System analyst* juga melakukan *monitoring* terhadap *task* terkait, dan ketika *task* sudah bisa dianggap selesai maka *issue* akan ditugaskan kepada QA untuk dilakukan pengujian validasi.

Pengujian yang dilakukan oleh QA mencakup pengujian fungsional, di mana semua fitur baru diuji secara fungsional berdasarkan spesifikasi dan *acceptance criteria* untuk memastikan bahwa fitur tersebut berfungsi dengan baik sebelum dirilis ke lingkungan produksi. Selain itu, dilakukan juga pengujian non-fungsional seperti performa, keamanan, dan kompatibilitas untuk memastikan aplikasi memenuhi standar kualitas yang ditetapkan. Pengujian regresi turut dilakukan secara rutin untuk memastikan perubahan kode baru tidak mengganggu fungsionalitas yang sudah ada. Seluruh hasil pengujian, termasuk *bug* yang ditemukan, langkah-langkah reproduksi, dan status perbaikannya, didokumentasikan dengan rinci untuk memastikan transparansi dan efektivitas dalam tindak lanjut. Kolaborasi antara QA dan tim pengembang juga berjalan secara intensif guna memastikan *bug* diidentifikasi, dilaporkan, dan diselesaikan dengan cepat.

3.4.4 Pemantauan dan Pengendalian Proyek

Proses pengendalian dan pemantauan dalam pengembangan Proyek B dilakukan dengan memanfaatkan tiga *tools* utama, yaitu GitLab, Taiga, dan Metabase. GitLab berfungsi sebagai *Project Management Information System* (PMIS) untuk pengelolaan dan pemantauan setiap tugas yang diberikan kepada tim (Oktavanusa, 2022). Pemanfaatan GitLab memungkinkan *lead* untuk memantau seluruh pekerjaan yang ditugaskan kepada *developer* dengan mudah. Selain itu, pemantauan proyek juga dilakukan menggunakan Taiga, di mana *lead* dapat

mengawasi pencapaian *task-task* yang terdaftar dalam *sprint*, apakah telah tercapai secara keseluruhan atau masih ada *task* yang terlambat dan mengalami kendala. Di sisi lain, *tools* utama yang sering digunakan oleh *lead* adalah Metabase. Platform intelijen bisnis *open-source* ini memungkinkan pengguna untuk mengajukan berbagai pertanyaan terkait data dan dapat diintegrasikan ke dalam aplikasi, memberikan kesempatan kepada pelanggan untuk menjelajahi data mereka sendiri (Metabase, 2024). PT Javan Cipta Solusi memanfaatkan Metabase ini untuk berbagai tujuan pemantauan, termasuk *monitoring tasks* tim, pengukuran skor capaian, pemantauan target harian, dan berbagai keperluan lainnya.

3.4.5 Penutupan Proyek

Proses penutupan aktivitas untuk Proyek B belum dapat dilakukan karena masih terdapat beberapa *issue* yang perlu diselesaikan. Pengecekan kualitas kode menggunakan SonarQube masih diperlukan untuk memastikan standar kode terpenuhi. Pelaksanaan *functional test* juga masih berlangsung dengan menggunakan *tools* baru, yaitu Apidog, menggantikan Postman yang sebelumnya digunakan oleh tim proyek. Beberapa dokumen *deliverable* pun masih membutuhkan penyelesaian dan penyempurnaan. Oleh karena itu, penyelesaian seluruh *issue* dan kebutuhan lainnya dari Proyek B akan dilanjutkan oleh karyawan lain yang masih aktif bekerja di PT Javan Cipta Solusi.

3.5 Pembuatan Dokumen *Source Code*

Penulis menyusun dokumen *source code* untuk Proyek A, C, D, E, dan F, dengan detail *timeline* dapat dilihat pada Tabel 3.2, yang menunjukkan bahwa penulis berkontribusi selama kurang lebih satu bulan lamanya. Dokumen *source code* merupakan bagian dari dokumen *deliverables* yang diminta oleh klien, bersamaan dengan dokumen lainnya. Di antara semua dokumen *deliverables* yang diminta, dokumen *source code* adalah salah satu yang melibatkan *programmer* dalam penyusunannya. Dokumen *source code* ini berisi kode per halaman yang disertai dengan komentar di setiap baris, serta informasi terkait folder tempat penyimpanan kode *front-end* dan *back-end*. Contoh dokumen *source code* yang telah disusun dapat dilihat pada Gambar 3.79. Gambar 3.79 ini menampilkan *source code* salah satu halaman *dashboard* aplikasi. Pada dokumen tersebut, diberikan informasi terkait *route*, letak folder, dan *code* yang berhubungan dengan halaman tersebut baik dari sisi *back-end* maupun *front-end*. Setiap *code* juga akan diberikan komentar penjelasan untuk memberikan pemahaman yang lebih optimal kepada para pengembang.

Tabel 3.2 Aktivitas penyusunan dokumen *source code*

No.	Deskripsi Kegiatan	Waktu	Durasi
1.	Mengerjakan dokumen <i>source code</i> proyek D (Aplikasi Presensi Peserta Ujian)	26 Oktober - 30 November 2023	1 Bulan
2.	Mengerjakan dokumen <i>source code</i> proyek E (<i>Showcase Portfolio</i> Peserta)	2-27 November 2023	1 Bulan
3.	Mengerjakan dokumen <i>source code</i> proyek C (Layanan API dan <i>Widget</i>)	7-9 November 2023	3 Hari
4.	Mengerjakan dokumen <i>source code</i> proyek F (Web <i>Crawling</i> Data Sosial Media)	14-15 November 2023	2 Hari
5.	Mengerjakan dokumen <i>source code</i> proyek A (Sistem Informasi <i>Jobseeker</i>)	16-23 November 2023	1 Minggu

The image shows a document with three sections: 1.1. Route, 1.2. Folder, and 1.3. Menampilkan Dashboard Presensi. Section 1.1 shows a route definition for the index action. Section 1.2 shows the file paths for the view and controller. Section 1.3 shows the PHP code for the index function, which returns a view with various data points.

```

1.1. Route
    1. view:
        Route::get('/', 'Admin\DashboardController@index')->name('index');
    2. listData: -

1.2. Folder
    FE : resources/views/admin/dashboard/dashboard.blade.php
    BE : app/Http/Controllers/Admin/DashboardController.php

1.3. Menampilkan Dashboard Presensi

#Nama file : DashboardController.php
#Fungsi: index()
#Deskripsi: Program ini dibuat untuk menampilkan dashboard untuk
view di halaman /dashboard

public function index()
{
    // Menghitung dan operasi beberapa data yang dipanggil

    // Mengembalikan tampilan (view) dengan berbagai data
    return view('admin.dashboard.dashboard', [
        'total_pelatihan' => $total_pelatihan,
        'total_event' => $total_event,
        'top5event' => $top5event,
        'top5pelatihan' => $top5pelatihan,
        'total_registrasi_event' => $total_registrasi_event,
        'total_hadir_event' => $total_hadir_event,
        'total_tidak_hadir_event' => $total_tidak_hadir_event,
        'total_registrasi_pelatihan' =>

```

Gambar 3.79 Contoh dokumen *source code*

BAB IV

REFLEKSI PELAKSANAAN MAGANG

Selama menjalani lebih dari enam bulan masa magang di PT Javan Cipta Solusi, penulis mendapatkan banyak pengalaman berharga. Beberapa hal utama yang dipelajari selama magang mencakup pemahaman mengenai pemilihan teknologi yang tepat untuk setiap proyek, proses dokumentasi proyek, penerapan metode *Scrum* dalam pengembangan perangkat lunak, serta penggunaan SonarQube untuk *unit test* dan Apidog untuk *functional test*. Selain itu, penulis juga merasakan manfaat lain yang diperoleh dari keterlibatan aktif dalam berbagai aktivitas magang. Pengalaman dan pembelajaran tersebut memberikan kesempatan untuk melakukan refleksi diri, baik dalam hal kelebihan yang dimiliki maupun area yang masih memerlukan perbaikan. Dengan refleksi ini, penulis dapat memahami potensi diri dan berkomitmen untuk menjadi pribadi yang lebih baik serta siap menghadapi tantangan profesional di masa mendatang.

4.1 Evaluasi Proses Magang

Proses magang ini dievaluasi untuk mengidentifikasi pengetahuan dan keterampilan yang bisa diterapkan di masa depan. Evaluasi juga mencakup beberapa saran yang ditujukan kepada perusahaan, dengan tujuan agar perusahaan dapat terus berkembang dan meningkatkan kualitas proses magang di masa yang akan datang. Penjelasan terkait setiap evaluasi akan diuraikan lebih lanjut pada subbab berikutnya.

4.1.1 Pemilihan Teknologi pada Setiap Proyek

Pemilihan teknologi yang tepat menjadi salah satu aspek penting dalam pengembangan aplikasi web. Selama masa magang di PT Javan Cipta Solusi, penulis terlibat dalam berbagai proyek dengan kebutuhan yang beragam, menekankan pentingnya seleksi teknologi yang cermat. PT Javan Cipta Solusi umumnya menggunakan Laravel untuk *back-end* dan ReactJS untuk *front-end* dalam proyek-proyeknya. Pemilihan teknologi ini didasarkan pada beberapa faktor, salah satunya adalah kebutuhan dari klien agar aplikasinya mendukung arsitektur *microservice*. Dalam pengembangan aplikasi berbasis *microservice*, *front-end* umumnya memanfaatkan Vue atau React. Namun, sebagian besar proyek di Javan memilih React karena perusahaan ini telah mengembangkan *library* bernama 'Alurkerja', yang saat ini hanya mendukung React. Selain itu, Javan juga memiliki *library* khusus untuk Laravel bernama

‘Laravolt’, yang dirancang untuk memudahkan pengembangan aplikasi dan menghilangkan *boilerplate*. Dengan menggunakan Laravolt, *developer* dapat dengan mudah mengimplementasikan fitur seperti CRUD, *search*, *filter*, dan *sort* ke dalam aplikasi dengan *effort* yang lebih sedikit, sehingga mempercepat proses pengembangan.

Di sisi lain, meskipun PT Javan Cipta Solusi umumnya menggunakan ReactJS dalam pengembangan proyek-proyeknya, penelitian terbaru dari Karić & Durmić (2024) menunjukkan bahwa VueJS lebih unggul dalam menawarkan performa terbaik untuk pengembangan aplikasi web, khususnya untuk aplikasi skala kecil hingga menengah, walaupun ReactJS tetap menjadi pilihan populer di kalangan pengembang. Maka, perlu dilakukan proses analisis lebih lanjut mengenai performa kedua *framework* tersebut terkait penerapannya dalam proyek pengembangan aplikasi web.

Pengujian Menggunakan Lighthouse

Untuk menganalisis perbandingan setiap teknologi *framework*, dilakukan pengujian menggunakan Lighthouse, sebuah alat *open-source* yang dapat menilai performa aplikasi web. Lighthouse mengukur empat metrik utama, yaitu kinerja (*performance*), aksesibilitas (*accessibility*), praktik terbaik (*best practices*), dan SEO (Muna et al., 2022). Metrik-metrik ini dipilih untuk memberikan panduan komprehensif dalam meningkatkan kualitas web karena faktor-faktor ini berdampak besar pada pengalaman pengguna serta kinerja situs dalam berbagai aspek. Siahaan & Vianto (2022) menguraikan enam kriteria spesifik untuk mengukur kinerja *website*, namun versi terbaru dari Lighthouse menyederhanakannya menjadi lima kriteria, termasuk *First Contentful Paint* (FCP), *Largest Contentful Paint* (LCP), *Total Blocking Time* (TBT), *Cumulative Layout Shift* (CLS), dan *Speed Index* (SI), dengan tingkat kinerja yang diukur berdasarkan waktu atau pergeseran *layout*.

a. Kinerja (*performance*)

Lighthouse 10	
Audit	Weight
First Contentful Paint	10%
Speed Index	10%
Largest Contentful Paint	25%
Total Blocking Time	30%
Cumulative Layout Shift	25%

Gambar 4.1 Bobot metrik kinerja spesifik Lighthouse versi 10

Sumber: (Developers, 2019b)

Pada pengukuran kinerja, Lighthouse memberikan bobot yang berbeda untuk setiap metrik spesifik. Berdasarkan Gambar 4.1, tiga metrik yang memiliki bobot tertinggi yaitu TBT (30%), LCP (25%), dan CLS (25%) yang merupakan bagian dari *Core Web Vitals*, yaitu metrik utama yang digunakan Google untuk menilai performa situs. Sementara FCP dan SI masing-masing memiliki bobot 10%.

1. *First Contentful Paint* (FCP)

First Contentful Paint (FCP) dengan bobot 10% mengukur waktu yang diperlukan sebuah *website* untuk menampilkan konten pertama, seperti teks, gambar atau elemen lainnya. FCP dihitung sebagai selisih antara waktu FCP dan waktu awal muat halaman (Kravchenko et al., 2023), menggunakan formula (4.1). Semakin kecil nilai FCP, semakin cepat halaman web akan menampilkan konten pertama.

$$\text{FCP} = \text{tFCP} - \text{tStart} \quad (4.1)$$

Keterangan:

tFCP = waktu saat konten pertama ditampilkan

tStart = waktu awal pemuatan halaman

Lighthouse mengategorikan nilai FCP menjadi tiga tingkat, yaitu performa baik (hijau) jika waktu *load* 0-1,8 detik, perlu perbaikan (oranye) untuk waktu 1,8-3 detik, dan buruk (merah) jika melebihi 3 detik.

2. *Largest Contentful Paint* (LCP)

Largest Contentful Paint (LCP) dengan bobot 25% mengukur waktu yang dibutuhkan untuk me-*render* elemen terbesar yang terlihat dalam *viewport*, seperti gambar besar, video atau elemen penting lainnya yang menarik perhatian pengguna. LCP dihitung sebagai selisih antara waktu LCP dan waktu awal muat halaman (Kravchenko et al., 2023), menggunakan formula (4.2).

$$\text{LCP} = \text{tLCP} - \text{tStart} \quad (4.2)$$

Keterangan:

tLCP = waktu ketika elemen terbesar selesai di-*render*

tStart = waktu awal pemuatan halaman

Performa LCP dianggap baik (hijau) jika waktu *load* 0-2,5 detik, perlu perbaikan (oranye) untuk 2,5-4 detik, dan buruk (merah) jika lebih dari 4 detik.

3. *Total Blocking Time (TBT)*

Total Blocking Time (TBT) dengan bobot tertinggi 30%, mengukur total waktu yang dibutuhkan *website* untuk memblokir semua respon *input* pengguna selama pemuatan halaman, yaitu dari awal pemuatan sampai fitur interaktif berjalan, di mana formula (4.3) menunjukkan model matematis untuk TBT (Kravchenko et al., 2023).

$$TBT = \sum_{i=1}^N BT_i \quad (4.3)$$

Keterangan:

BT_i = waktu *blocking* untuk *task* ke- i yang $>50ms$

N = jumlah *task* yang memblokir halaman

Hasil dari TBT dianggap baik (hijau) jika berada di rentang 0-200 milidetik, perlu perbaikan (oranye) untuk 200-600 milidetik, dan buruk (merah) jika melebihi 600 milidetik.

4. *Cumulative Layout Shift (CLS)*

Cumulative Layout Shift (CLS) dengan bobot 25% mengukur stabilitas visual halaman dengan menghitung seberapa banyak elemen-elemen bergeser selama proses pemuatan hingga semua elemen mencapai keadaan stabil. CLS dihitung dengan formula (4.4). Untuk menghitung skornya, browser melihat ukuran area pandang dan pergerakan *elemen yang tidak stabil* di area pandang antara dua *frame* yang di-*render*. Skor pergeseran tata letak ini adalah hasil dari dua ukuran gerakan tersebut, yaitu *fraksi dampak* dan *fraksi jarak* (Web.dev, 2023).

$$CLS = \Sigma(\text{impact fraction} \times \text{distance fraction}) \quad (4.4)$$

Keterangan:

impact fraction = rasio area yang bergeser / *viewport area*

distance fraction = rasio jarak pergeseran / *viewport dimension*

CLS yang menghasilkan nilai di bawah 0,1 menunjukkan performa baik (hijau), 0,1-0,25 menandakan perlu perbaikan (oranye), dan nilai di atas 0,25 dikategorikan buruk (merah).

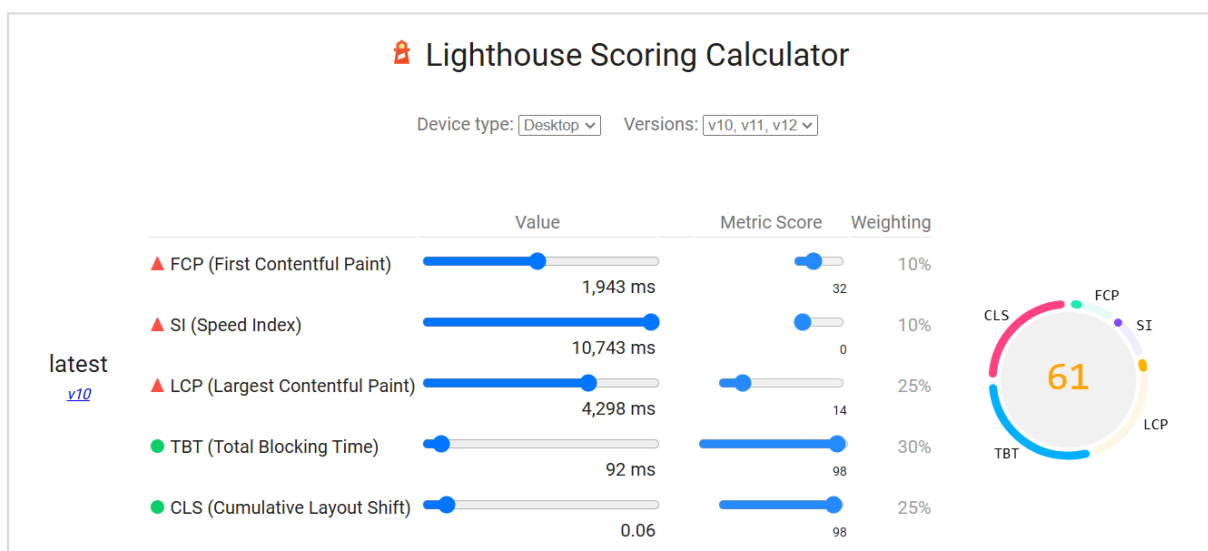
5. *Speed Index* (SI)

Speed Index dengan bobot 10% mengukur seberapa cepat konten visual ditampilkan selama halaman dimuat. Pertama, Lighthouse menangkap video halaman yang dimuat di browser dan menghitung transisi visual antara *frame*, kemudian menggunakan modul Speedline Node.js untuk menghitung *Speed Index* (Kravchenko et al., 2023). Skor SI adalah perbandingan indeks kecepatan halaman dan indeks kecepatan situs web nyata, berdasarkan data dari Arsip HTTP (Developers, 2019b), dengan kategori baik (hijau) untuk waktu 0-3,4 detik, perlu perbaikan (oranye) untuk 3,4-5,8 detik, dan buruk (merah) jika melebihi 5,8 detik.

Skor akhir performa dihitung menggunakan *weighted average* dari kelima metrik tersebut dengan formula (4.5). Setiap metrik dikonversi ke skor 0-100 sebelum dikalikan dengan bobotnya masing-masing. Semakin tinggi skor akhir mengindikasikan performa yang lebih baik, dengan interpretasi: baik (hijau, 90-100), perlu perbaikan (oranye, 50-89), dan buruk (merah, 0-49).

$$\text{Performance Score} = (\text{FCP} \times 0.10) + (\text{LCP} \times 0.25) + (\text{TBT} \times 0.30) + (\text{CLS} \times 0.25) + (\text{SI} \times 0.10) \quad (4.5)$$

Namun, untuk memudahkan perhitungan, Lighthouse sebenarnya telah menyediakan *scoring calculator* seperti yang terlihat pada Gambar 4.2. Kalkulator Lighthouse ini akan menampilkan nilai metrik dan skor berdasarkan pembobotan, serta visualisasi hasil dalam bentuk diagram lingkaran.



Gambar 4.2 Lighthouse *scoring calculator*

Sumber: (Developers, 2019b)

b. Aksesibilitas (*accessibility*)

Skor *accessibility* Lighthouse merupakan rata-rata tertimbang dari semua audit aksesibilitas yang didasarkan pada *aXe user impact assessments*. Sistem penilaian *aXe* mengelompokkan tingkat dampak menjadi empat kategori, yaitu *minor*, *moderate*, *serious*, dan *critical*. Setiap audit aksesibilitas dinilai dengan sistem lulus atau gagal, berbeda dengan audit *Performance* yang memungkinkan penilaian parsial. Dalam audit aksesibilitas, sebuah halaman tidak akan mendapatkan poin jika hanya sebagian yang memenuhi kriteria. Sebagai contoh, jika beberapa tombol pada sebuah halaman memiliki nama yang dapat diakses (*accessible names*) sementara yang lain tidak, maka halaman tersebut akan mendapatkan skor 0 untuk audit “*Buttons do not have an accessible name*” (Developers, 2019a).

Bobot penilaian untuk setiap audit aksesibilitas ditunjukkan pada Gambar 4.3 dan Gambar 4.4, di mana audit dengan bobot lebih tinggi memiliki dampak yang lebih besar terhadap skor akhir. Sementara itu, audit manual yang ditampilkan pada Gambar 4.5 tidak disertakan dalam tabel pembobotan karena tidak akan memengaruhi skor.

Audit	Weight
[<code>accesskey</code>] values are unique	7
[<code>aria-*</code>] attributes match their roles	10
Values assigned to <code>role=""</code> are valid ARIA roles	1
<code>button</code> , <code>link</code> , and <code>menuitem</code> elements have accessible names	7
Elements with <code>role="dialog"</code> or <code>role="alertdialog"</code> have accessible names	7
[<code>aria-hidden="true"</code>] is not present on the document <code><body></code>	10
[<code>aria-hidden="true"</code>] elements do not contain focusable descendants	7
ARIA input fields have accessible names	7
ARIA <code>meter</code> elements have accessible names	7
ARIA <code>progressbar</code> elements have accessible names	7
[<code>role</code>]s have all required [<code>aria-*</code>] attributes	10
Elements with an ARIA [<code>role</code>] that require children to contain a specific [<code>role</code>] have all required children	10
[<code>role</code>]s are contained by their required parent element	10
[<code>role</code>] values are valid	7
Elements with the <code>role=text</code> attribute don't have focusable descendants	7
ARIA toggle fields have accessible names	7
ARIA <code>tooltip</code> elements have accessible names	7
ARIA <code>treeitem</code> elements have accessible names	7
[<code>aria-*</code>] attributes have valid values	10

Gambar 4.3 Bobot untuk setiap audit aksesibilitas Lighthouse (1)

Sumber: (Developers, 2019a)

[aria-*] attributes are valid and not misspelled	10
Buttons have an accessible name	10
The page contains a heading, skip link, or landmark region	7
Background and foreground colors have a sufficient contrast ratio	7
<dl>'s contain only properly-ordered <dt> and <dd> groups, <script>, <template> or <div> elements	7
Definition list items are wrapped in <dl> elements	7
Document has a <title> element	7
[id] attributes on active, focusable elements are unique	7
ARIA IDs are unique	10
No form fields have multiple labels	3
<frame> or <iframe> elements have a title	7
Heading elements appear in a sequentially-descending order	3
<html> element has a [lang] attribute	7
<html> element has a valid value for its [lang] attribute	7
<html> element has an [xml:lang] attribute with the same base language as the [lang] attribute	3
Image elements have [alt] attributes	10
Image elements do not have [alt] attributes that are redundant text	1
Input buttons have discernible text	10
<input type="image"> elements have [alt] text	10
Elements with visible text labels have matching accessible names	7
Form elements have associated labels	7
Links are distinguishable without relying on color	7
Links have a discernible name	7
Lists contain only elements and script supporting elements (<script> and <template>)	7
List items () are contained within , or <menu> parent elements	7
The document does not use <meta http-equiv="refresh">	10
[user-scalable="no"] is not used in the <meta name="viewport"> element and the [maximum-scale] attribute is not less than 5	10
<object> elements have alternate text	7
Select elements have associated label elements	7
Skip links are focusable	3
No element has a [tabindex] value greater than 0	7
Tables have different content in the summary attribute and <caption>	1
Tables use <caption> instead of cells with the [colspan] attribute to indicate a caption	7
<td> elements in a large <table> have one or more table headers	10
Cells in a <table> element that use the [headers] attribute refer to table cells within the same table	7
<th> elements and elements with [role="columnheader"/"rowheader"] have data cells they describe	7
[lang] attributes have a valid value	7
<video> elements contain a <track> element with [kind="captions"]	10

Gambar 4.4 Bobot untuk setiap audit aksesibilitas Lighthouse (2)

Sumber: (Developers, 2019a)

```

'accessibility': {
  title: str_(UIStrings.a11yCategoryTitle),
  description: str_(UIStrings.a11yCategoryDescription),
  manualDescription: str_(UIStrings.a11yCategoryManualDescription),
  supportedModes: ['navigation', 'snapshot'],
  // Audit weights are meant to match the aXe scoring system of
  // minor, moderate, serious, and critical.
  // See the audits listed at dequeuniversity.com/rules/axe/4.7.
  // Click on an audit and check the right hand column to see its severity.
  auditRefs: [
    // Manual audits
    {id: 'focusable-controls', weight: 0},
    {id: 'interactive-element-affordance', weight: 0},
    {id: 'logical-tab-order', weight: 0},
    {id: 'visual-order-follows-dom', weight: 0},
    {id: 'focus-traps', weight: 0},
    {id: 'managed-focus', weight: 0},
    {id: 'use-landmarks', weight: 0},
    {id: 'offscreen-content-hidden', weight: 0},
    {id: 'custom-controls-labels', weight: 0},
    {id: 'custom-controls-roles', weight: 0},
    // Hidden audits
    {id: 'empty-heading', weight: 0, group: 'hidden'},
    {id: 'identical-links-same-purpose', weight: 0, group: 'hidden'},
    {id: 'landmark-one-main', weight: 0, group: 'hidden'},
    {id: 'label-content-name-mismatch', weight: 0, group: 'hidden'},
    {id: 'table-fake-caption', weight: 0, group: 'hidden'},
    {id: 'td-has-header', weight: 0, group: 'hidden'},
  ],
},

```

Gambar 4.5 Bobot untuk setiap audit aksesibilitas Lighthouse (3)

Sumber: (GoogleChrome, 2024a)

c. Praktik terbaik (*best practices*)

Untuk *best practices*, Lighthouse menjalankan pemeriksaan terhadap praktik terbaik terkait keamanan dan implementasi kode. Pemeriksaan ini mencakup berbagai aspek seperti penggunaan HTTPS, pengelolaan *resource* eksternal, dan deteksi potensi risiko keamanan. Setiap pemeriksaan yang berhasil memberikan poin positif, sedangkan yang gagal akan mengurangi poin. Semua audit dalam kategori *best practices* terlihat pada Gambar 4.6, di mana jika setiap audit diterapkan dengan benar akan meningkatkan skor keseluruhan hingga enam poin (GoogleChrome, 2024b). Skor akhirnya didasarkan pada persentase dari total pemeriksaan yang berhasil lolos.

```

'best-practices': {
  title: str_(UIStrings.bestPracticesCategoryTitle),
  supportedModes: ['navigation', 'timespan', 'snapshot'],
  auditRefs: [
    // Trust & Safety
    {id: 'is-on-https', weight: 5, group: 'best-practices-trust-safety'},|
    {id: 'redirects-http', weight: 1, group: 'best-practices-trust-safety'},
    {id: 'geolocation-on-start', weight: 1, group: 'best-practices-trust-safety'},
    {id: 'notification-on-start', weight: 1, group: 'best-practices-trust-safety'},
    {id: 'csp-xss', weight: 0, group: 'best-practices-trust-safety'},
    // User Experience
    {id: 'paste-preventing-inputs', weight: 3, group: 'best-practices-ux'},
    {id: 'image-aspect-ratio', weight: 1, group: 'best-practices-ux'},
    {id: 'image-size-responsive', weight: 1, group: 'best-practices-ux'},
    {id: 'viewport', weight: 1, group: 'best-practices-ux'},
    {id: 'font-size', weight: 1, group: 'best-practices-ux'},
    // Browser Compatibility
    {id: 'doctype', weight: 1, group: 'best-practices-browser-compat'},
    {id: 'charset', weight: 1, group: 'best-practices-browser-compat'},
    // General Group
    {id: 'js-libraries', weight: 0, group: 'best-practices-general'},
    {id: 'deprecations', weight: 5, group: 'best-practices-general'},
    {id: 'third-party-cookies', weight: 5, group: 'best-practices-general'},
    {id: 'errors-in-console', weight: 1, group: 'best-practices-general'},
    {id: 'valid-source-maps', weight: 0, group: 'best-practices-general'},
    {id: 'inspector-issues', weight: 1, group: 'best-practices-general'},
  ],
},

```

Gambar 4.6 Bobot untuk setiap audit *best practices* Lighthouse

Sumber: (GoogleChrome, 2024a)

d. SEO

Penilaian SEO diperoleh berdasarkan pemeriksaan terhadap elemen-elemen yang memengaruhi optimasi mesin pencari, seperti adanya *tag* meta, *heading* yang terstruktur dengan baik, dan URL yang ramah SEO. Sebagaimana ditunjukkan pada Gambar 4.7, semua audit dalam kategori SEO memiliki bobot yang sama, kecuali *structured data*, yang merupakan audit manual tanpa skor. Penerapan yang benar pada setiap audit ini akan meningkatkan skor keseluruhan hingga delapan poin (GoogleChrome, 2024b).

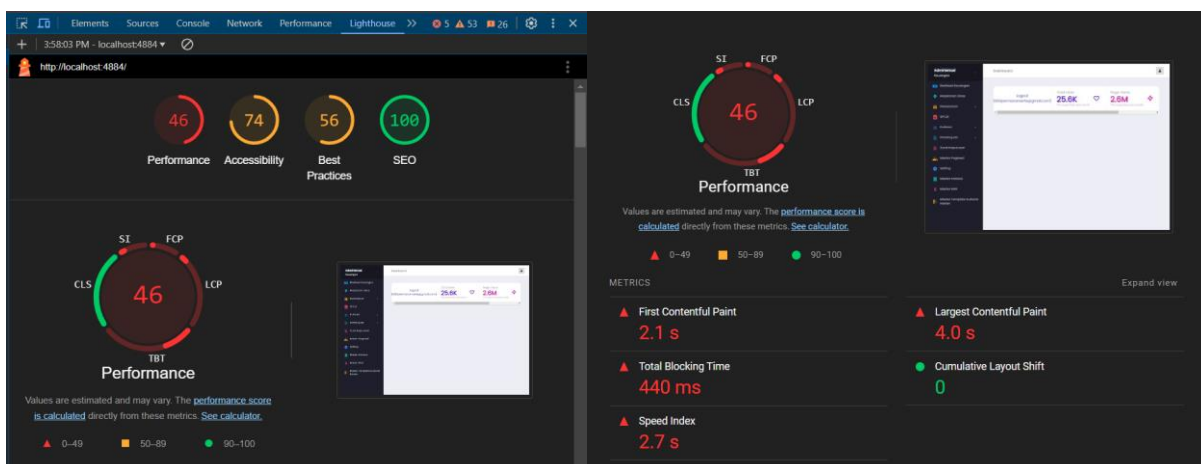
```
'seo': {
  title: str_(UIStrings.seoCategoryTitle),
  description: str_(UIStrings.seoCategoryDescription),
  manualDescription: str_(UIStrings.seoCategoryManualDescription),
  supportedModes: ['navigation', 'snapshot'],
  auditRefs: [
    // Should be at least 31% of the score, such that this audit failing
    // results in the SEO category failing.
    // Solve for w:
    // w / (w + T) >= 0.31
    // where T is the sum of all the other weights.
    {id: 'is-crawable', weight: 93 / 23, group: 'seo-crawl'},
    {id: 'document-title', weight: 1, group: 'seo-content'},
    {id: 'meta-description', weight: 1, group: 'seo-content'},
    {id: 'http-status-code', weight: 1, group: 'seo-crawl'},
    {id: 'link-text', weight: 1, group: 'seo-content'},
    {id: 'crawlable-anchors', weight: 1, group: 'seo-crawl'},
    {id: 'robots-txt', weight: 1, group: 'seo-crawl'},
    {id: 'image-alt', weight: 1, group: 'seo-content'},
    {id: 'hreflang', weight: 1, group: 'seo-content'},
    {id: 'canonical', weight: 1, group: 'seo-content'},
    // Manual audits
    {id: 'structured-data', weight: 0},
  ],
},
```

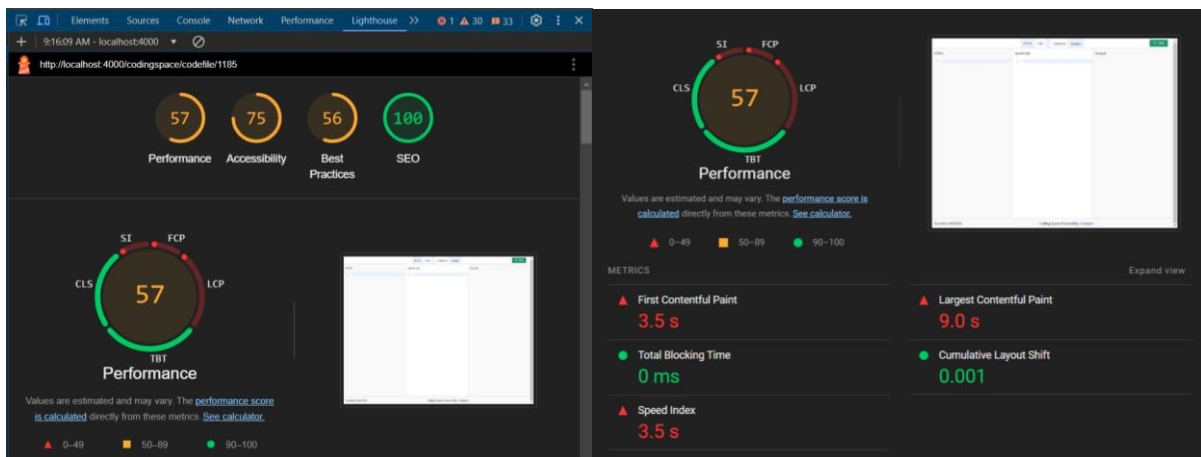
Gambar 4.7 Bobot untuk setiap audit SEO Lighthouse

Sumber: (GoogleChrome, 2024a)

Perbandingan Proyek B (ReactJS) dan Proyek G (VueJS)

Pengujian ini diterapkan pada dua proyek, proyek B yang menggunakan ReactJS dan proyek G yang menggunakan VueJS. Hasil pengujian menggunakan Lighthouse untuk proyek B dapat dilihat pada Gambar 4.8, sementara hasil untuk proyek G ditampilkan pada Gambar 4.9. Gambar 4.8 dan Gambar 4.9 menunjukkan hasil pengujian berdasarkan empat metrik yang diukur oleh Lighthouse, yaitu kinerja (*performance*), aksesibilitas (*accessibility*), praktik terbaik (*best practices*), dan SEO, serta menunjukkan hasil pengujian Lighthouse berdasarkan kinerja (*performance*) dengan lima kriteria metrik yang spesifik.

Gambar 4.8 Hasil pengujian empat metrik Lighthouse dan lima metrik kinerja (*performance*) pada proyek B (ReactJS)



Gambar 4.9 Hasil pengujian empat metrik Lighthouse dan lima metrik kinerja (*performance*) pada proyek G (VueJS)

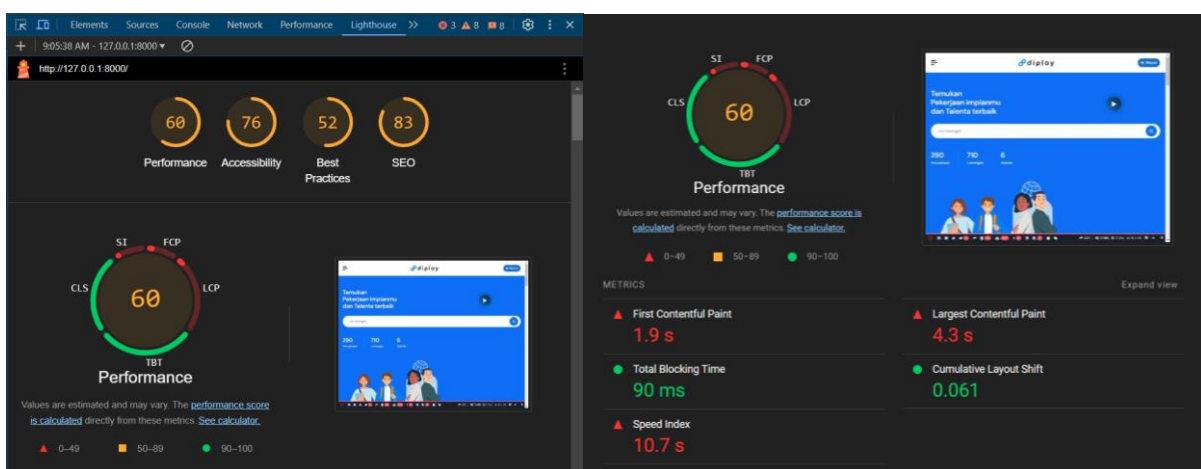
Analisis hasil pengujian Lighthouse menunjukkan perbedaan signifikan antara kedua proyek. Berdasarkan Gambar 4.8 dan Gambar 4.9, terlihat bahwa VueJS memiliki skor kinerja lebih baik secara keseluruhan dibandingkan ReactJS (57 vs 46). VueJS juga unggul dalam aksesibilitas, dengan skor 75 dibandingkan ReactJS 74, sementara keduanya memperoleh hasil yang sama pada Praktik Terbaik dan SEO. Namun, dalam metrik kinerja spesifik, seperti yang ditunjukkan pada bagian kanan Gambar 4.8, ReactJS lebih cepat dalam memuat konten pertama (*First Contentful Paint* 2,1 detik) dan *Largest Contentful Paint* (4,0 detik). Sementara itu, pada Gambar 4.9 terlihat bahwa VueJS unggul dalam *Total Blocking Time* (TBT), dengan waktu 0 milidetik, jauh lebih baik dibandingkan ReactJS yang mencapai 440 milidetik. Untuk *Cumulative Layout Shift* (CLS), kedua teknologi menunjukkan stabilitas tata letak yang sangat baik, di mana ReactJS mencapai skor sempurna 0 dan VueJS dengan 0,001. Pada *Speed Index* (SI), hasil pengujian di Gambar 4.8 menunjukkan ReactJS lebih cepat dengan 2,7 detik dibandingkan VueJS yang mencatat waktu 3,5 detik pada Gambar 4.9.

Perbandingan Proyek A (Blade) dan Proyek G (VueJS)

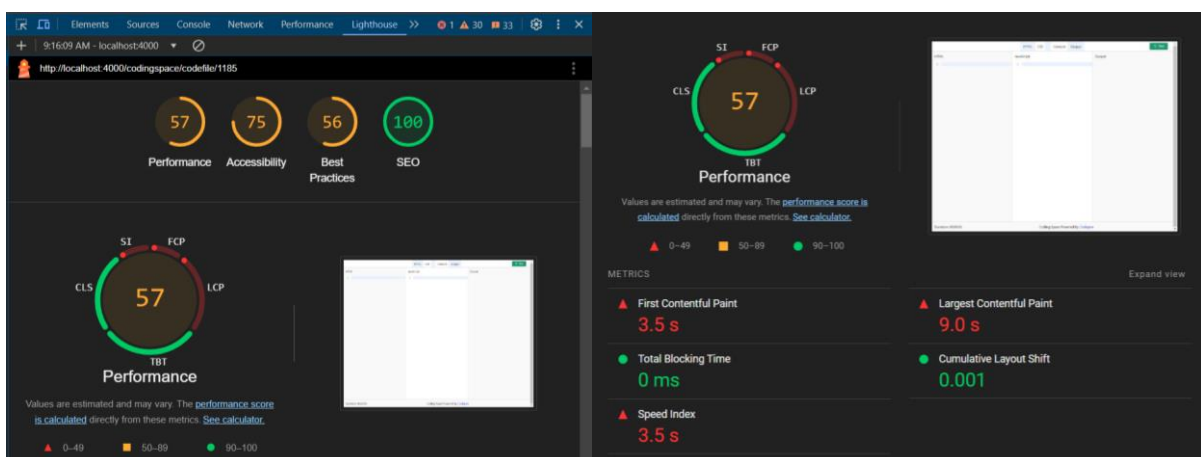
Selain itu, menurut Tang (2019), Laravel dan VueJS memiliki keterkaitan yang kuat, terutama sejak Laravel versi 5.3, di mana VueJS menjadi *framework front-end default*. Laravel sebagai *framework back-end* juga menyediakan alat *front-end* lengkap seperti JavaScript, Sass, dan `package.json` untuk pengelolaan dependensi. Meskipun Blade adalah *engine template default* Laravel, VueJS sudah terintegrasi secara langsung, sehingga memungkinkan pengembang untuk memanfaatkan kedua teknologi ini secara fleksibel. Mengingat VueJS dan Blade sama-sama berperan dalam pengembangan antarmuka pengguna, maka penulis

berinisiatif untuk melakukan pengujian perbandingan performa antara kedua teknologi tersebut menggunakan Lighthouse.

Pengujian ini diterapkan pada dua proyek, proyek A yang menggunakan Blade Templating Engine dan proyek G yang menggunakan VueJS. Hasil pengujian menggunakan Lighthouse untuk proyek A dapat dilihat pada Gambar 4.10, sementara untuk proyek G ditampilkan pada Gambar 4.11. Gambar 4.10 dan Gambar 4.11 menunjukkan hasil pengujian berdasarkan empat metrik utama Lighthouse, yaitu kinerja (*performance*), aksesibilitas (*accessibility*), praktik terbaik (*best practices*), dan SEO, serta menunjukkan hasil pengujian Lighthouse berdasarkan kinerja (*performance*) dengan lima kriteria metrik yang lebih spesifik.



Gambar 4.10 Hasil pengujian empat metrik Lighthouse dan lima metrik kinerja (*performance*) pada proyek A (Blade)



Gambar 4.11 Hasil pengujian empat metrik Lighthouse dan lima metrik kinerja (*performance*) pada proyek G (VueJS)

Analisis hasil pengujian Lighthouse menunjukkan perbedaan performa antara proyek A (Blade) dan G (VueJS). Berdasarkan Gambar 4.10 dan Gambar 4.11, terlihat bahwa Blade

memiliki skor kinerja lebih baik secara keseluruhan dibandingkan VueJS (60 vs 57). Dalam hal aksesibilitas, seperti yang ditunjukkan pada bagian kiri kedua gambar, Blade sedikit unggul dengan skor 76 dibanding VueJS 75, namun VueJS lebih baik dalam Praktik Terbaik (56 vs 52) dan SEO (100 vs 83). Untuk metrik kinerja spesifik, hasil pengujian pada bagian kanan Gambar 4.10 menunjukkan Blade lebih cepat dalam *First Contentful Paint* (1,9 detik) dan *Largest Contentful Paint* (4,3 detik) dibandingkan VueJS yang mencatat waktu masing-masing 3,5 detik dan 9,0 detik pada Gambar 4.11. Sebaliknya, data pada Gambar 4.11 memperlihatkan VueJS unggul dalam *Total Blocking Time* dengan 0 milidetik, dibandingkan Blade yang mencapai 90 milidetik, serta menunjukkan stabilitas tata letak yang lebih baik dengan *Cumulative Layout Shift* 0,001 dibandingkan Blade 0,061. Pada *Speed Index*, hasil di Gambar 4.11 menunjukkan VueJS mencatat waktu 3,5 detik, lebih cepat dibandingkan Blade yang mencapai 10,7 detik seperti terlihat pada Gambar 4.3.

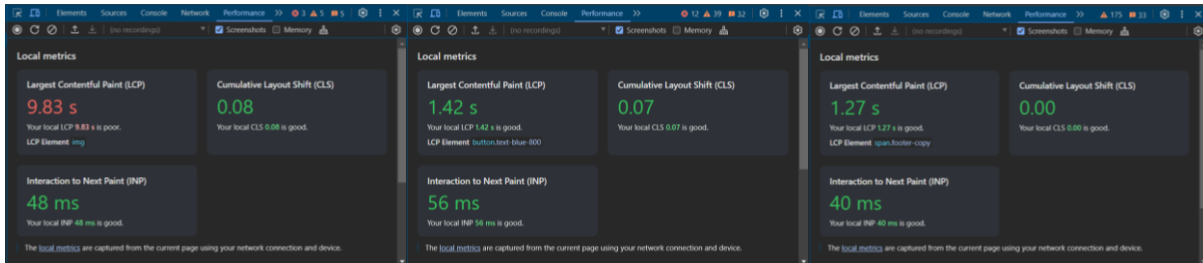
Pengujian Menggunakan Chrome DevTools Performance Tab

Untuk mendapatkan gambaran yang lebih komprehensif, penulis juga menggunakan Chrome DevTools *Performance tab*, yang menambahkan analisis lebih lanjut secara *real-time* dengan kriteria seperti *Largest Contentful Paint* (LCP), *Cumulative Layout Shift* (CLS), dan *Interaction to Next Paint* (INP) (Kaleev, 2024). Pengujian ini dilakukan karena versi terbaru Lighthouse (versi 10) telah menggantikan salah satu metrik, yaitu *Time to Interactive* (TTI) dengan *Interaction to Next Paint* (INP). Meskipun INP tidak secara langsung ditampilkan dalam hasil pengujian utama, metrik ini masih dapat dilihat pada tab *Performance* di Chrome. INP mengukur responsivitas keseluruhan halaman terhadap interaksi pengguna dengan memantau latensi pada setiap interaksi seperti klik, ketukan, atau *input* dari *keyboard* yang terjadi selama kunjungan pengguna ke halaman. Nilai akhir INP mencerminkan interaksi terlama yang diamati selama sesi pengguna, dengan kriteria sebagai berikut:

- INP di bawah atau sama dengan 200 milidetik menunjukkan halaman memiliki responsivitas yang baik.
- INP antara 200 dan 500 milidetik menandakan perlu adanya peningkatan dalam responsivitas halaman.
- INP di atas 500 milidetik berarti halaman memiliki responsivitas yang buruk.

Berdasarkan Gambar 4.12, hasil pengujian menggunakan Chrome DevTools *Performance tab* menunjukkan perbedaan kinerja yang signifikan di antara ketiga proyek. Pada bagian kiri Gambar 4.12 terlihat proyek A yang menggunakan Blade mendapatkan nilai LCP

sebesar 9,83 detik, CLS 0,08, dan INP 48 milidetik. Di bagian tengah, proyek B yang menggunakan ReactJS memperoleh hasil lebih baik dengan LCP 1,42 detik, CLS 0,07, dan INP 56 milidetik. Sementara itu, pada bagian kanan, proyek G yang menggunakan VueJS menunjukkan performa terbaik dengan nilai LCP 1,27 detik, CLS 0,00, dan INP 40 milidetik.



Gambar 4.12 Hasil pengujian kinerja (*performance*) proyek A (Blade), B (ReactJS), dan G (VueJS) menggunakan Chrome DevTools *Performance tab*

Hasil pengujian ini memperlihatkan bahwa proyek G dengan VueJS unggul dalam semua aspek pengukuran Chrome DevTools. VueJS mencatat waktu LCP tercepat, menunjukkan kemampuannya dalam memuat konten utama dengan efisien. Dari segi stabilitas tata letak, VueJS mencapai skor CLS sempurna (0,00), mengindikasikan tidak adanya pergeseran *layout* yang dapat mengganggu pengalaman pengguna. VueJS juga mengungguli yang lain dalam responsivitas dengan INP terendah (40 milidetik), menunjukkan kemampuannya dalam memberikan respons cepat terhadap interaksi pengguna. Sebaliknya, Blade menunjukkan keterbatasan dengan LCP tertinggi, meskipun masih memiliki performa INP yang cukup baik.

Pembahasan dan Kesimpulan

Berdasarkan serangkaian pengujian yang komprehensif menggunakan Lighthouse dan Chrome DevTools *Performance tab*, hasil analisis kinerja tiga teknologi *framework* (Blade, ReactJS, dan VueJS) telah disajikan dalam Tabel 4.2, Tabel 4.3, dan Tabel 4.4. Dalam penyajian data tersebut, sistem indikator warna digunakan untuk memudahkan interpretasi, di mana **hijau** menandakan kinerja baik atau cepat, **oranye** menunjukkan kinerja sedang atau area yang memerlukan perbaikan, dan **merah** mengindikasikan kinerja yang lambat atau buruk, di mana dalam setiap tabel hasil pengujian, terdapat tanda **centang hijau** yang menandakan nilai terbaik pada setiap kategori pengujian. Penjelasan mengenai indikator warna ini telah diuraikan juga pada Tabel 4.1, yang merangkum lima kriteria kinerja yang diukur oleh Lighthouse, termasuk *First Contentful Paint* (FCP), *Largest Contentful Paint* (LCP), *Total Blocking Time* (TBT), *Cumulative Layout Shift* (CLS), dan *Speed Index* (SI).

Tabel 4.1 Kriteria metrik kinerja pada Lighthouse

No.	Judul	Tujuan	Teknologi
1.	<i>First Contentful Paint</i> (FCP)	Waktu yang dibutuhkan <i>website</i> untuk menampilkan konten pertama dalam bentuk gambar dan teks.	<ul style="list-style-type: none"> • Cepat, indikator hijau (0-1,8 detik) • Sedang, indikator oranye (1,8-3 detik) • Lambat, indikator merah (> 3 detik)
2.	<i>Largest Contentful Paint</i> (LCP)	Mengukur waktu yang dibutuhkan <i>website</i> untuk menampilkan konten resolusi tertinggi pada <i>viewport</i> saat <i>rendering</i> .	<ul style="list-style-type: none"> • Cepat, indikator hijau (0-2,5 detik) • Sedang, indikator oranye (2,5-4 detik) • Lambat, indikator merah (> 4 detik)
3.	<i>Total Blocking Time</i> (TBT)	Total waktu yang dibutuhkan <i>website</i> untuk memblokir semua respon <i>input</i> pengguna selama pemuatan halaman, yaitu dari awal pemuatan sampai fitur interaktif berjalan.	<ul style="list-style-type: none"> • Cepat, indikator hijau (0-200 milidetik) • Sedang, indikator oranye (200-600 milidetik) • Lambat, indikator merah (> 600 milidetik)
4.	<i>Cumulative Layout Shift</i> (CLS)	Skor yang dihitung berdasarkan perubahan tata letak elemen-elemen <i>website</i> pada layar.	<ul style="list-style-type: none"> • Baik, indikator hijau (0-0,1) • Perlu perbaikan, indikator oranye (0,1-0,25) • Buruk, indikator merah (> 0,25)
5.	<i>Speed Index</i> (SI)	Mengukur kecepatan <i>website</i> dalam menampilkan konten secara visual selama pemuatan halaman.	<ul style="list-style-type: none"> • Cepat, indikator hijau (0-3,4 detik) • Sedang, indikator oranye (3,4-5,8 detik) • Lambat, indikator merah (> 5,8 detik)

Tabel 4.2 Hasil Pengujian Keseluruhan Metrik Lighthouse

Metrik	Blade	ReactJS	VueJS
Kinerja (<i>Performance</i>)	✔ 60	46	57
Aksesibilitas (<i>Accessibility</i>)	✔ 76	74	75
Praktik Terbaik (<i>Best Practices</i>)	52	✔ 56	✔ 56
SEO	83	✔ 100	✔ 100

Tabel 4.3 Hasil Pengujian Metrik Kinerja pada Lighthouse

Metrik	Blade	ReactJS	VueJS
<i>First Contentful Paint</i> (FCP)	✔ 1.9 detik	2.1 detik	3.5 detik
<i>Largest Contentful Paint</i> (LCP)	4.3 detik	✔ 4.0 detik	9.0 detik
<i>Total Blocking Time</i> (TBT)	90 milidetik	440 milidetik	✔ 0 milidetik
<i>Cumulative Layout Shift</i> (CLS)	0.061	✔ 0	0.001
<i>Speed Index</i> (SI)	10.7 detik	✔ 2.7 detik	3.5 detik
Skor Lighthouse (Kinerja)	✔ 60	46	57

Tabel 4.4 Hasil Pengujian Chrome DevTools *Performance* tab

Metrik	Blade	ReactJS	VueJS
<i>Largest Contentful Paint</i> (LCP)	9.83 detik	1.42 detik	✔ 1.27 detik
<i>Cumulative Layout Shift</i> (CLS)	0.08	0.07	✔ 0.00
<i>Interaction to Next Paint</i> (INP)	48 milidetik	56 milidetik	✔ 40 milidetik

a. Hasil Pengujian Metrik Utama Lighthouse

Tabel 4.2 merangkum hasil pengujian metrik utama Lighthouse untuk ketiga teknologi. Blade *Templating Engine* menunjukkan performa terbaik dalam skor kinerja Lighthouse dengan nilai 60, diikuti oleh VueJS (57) dan ReactJS (46). Hal ini dapat dikaitkan dengan arsitektur Blade yang terintegrasi erat dengan Laravel, memungkinkan optimasi yang lebih baik antara *front-end* dan *back-end*. Blade juga menggunakan pendekatan *server-side rendering* (SSR) yang dapat meningkatkan kecepatan pemuatan awal dan SEO. Namun, pendekatan SSR juga dapat menyebabkan beban server yang lebih tinggi dan waktu respons yang lebih lambat untuk interaksi pengguna yang kompleks, seperti yang dijelaskan dalam penelitian Chastro & Darmawan (2020).

Di sisi lain, dalam aspek aksesibilitas, Blade sedikit unggul dengan skor 76, dibandingkan dengan VueJS (75) dan ReactJS (74). Hal ini menunjukkan bahwa Blade memiliki keunggulan dalam menyediakan konten yang dapat diakses oleh berbagai jenis pengguna. Untuk praktik terbaik, ReactJS dan VueJS berbagi skor tertinggi yaitu 56, sementara Blade mencapai skor 52. Ini mengindikasikan bahwa ReactJS dan VueJS telah mengikuti standar dan praktik terbaik dalam pengembangan web dengan lebih baik dibandingkan Blade. ReactJS dan VueJS juga menunjukkan performa optimal dalam aspek SEO dengan skor sempurna 100, sedangkan Blade mencapai skor 83. Hal ini menunjukkan bahwa ReactJS dan VueJS telah dioptimalkan dengan baik untuk mesin pencari, memungkinkan peningkatan visibilitas dan peringkat dalam hasil pencarian.

b. Hasil Pengujian Metrik Kinerja Spesifik

Tabel 4.3 menyajikan hasil pengujian metrik kinerja spesifik untuk ketiga teknologi. Blade unggul dalam *First Contentful Paint* (FCP) dengan waktu 1,9 detik, yang menunjukkan bahwa Blade dapat memberikan respons visual pertama kepada pengguna dengan sangat cepat. Hal ini berbanding lurus dengan temuan Chastro & Darmawan (2020) bahwa Blade lebih sesuai untuk aplikasi sederhana yang membutuhkan *rendering* awal yang cepat.

ReactJS menunjukkan keunggulan dalam *Largest Contentful Paint* (LCP) dengan waktu 4,0 detik dan *Speed Index* (SI) 2,7 detik. Ini mengindikasikan bahwa ReactJS mampu memuat konten utama dengan cepat dan memberikan pengalaman pengguna yang responsif secara keseluruhan. Hal ini dapat dikaitkan dengan penggunaan Virtual DOM oleh ReactJS, yang memungkinkan pembaruan UI yang efisien tanpa *re-render* ulang seluruh halaman (Uriawan et al., 2024). ReactJS juga memiliki ekosistem yang kaya dan dukungan komunitas yang luas, yang berkontribusi pada optimasi performa dan penyelesaian masalah yang lebih cepat. Selain itu, dalam hal *Cumulative Layout Shift* (CLS), ReactJS mencapai skor sempurna 0, menunjukkan stabilitas tata letak yang sangat baik. Ini mengindikasikan bahwa ReactJS dapat menjaga konsistensi tampilan selama interaksi pengguna, mengurangi pergeseran konten yang tidak diinginkan.

VueJS unggul dalam *Total Blocking Time* (TBT) dengan 0 milidetik yang menunjukkan waktu minimal dalam memblokir *thread* utama dan memberikan interaktivitas yang lebih lancar. Keunggulan ini dapat dikaitkan dengan reaktivitas bawaan VueJS dan penerapan *lazy loading* yang baik (Hadi, 2024). VueJS juga memiliki ukuran *bundle* yang lebih kecil dibandingkan dengan React, yang berkontribusi pada waktu

pemuatan yang lebih cepat. Hal ini sejalan dengan temuan Sofi'ie & Qoiriah (2023) yang menyatakan bahwa VueJS menunjukkan performa yang lebih baik daripada React dalam sebagian besar skenario pengujian, mampu *me-render* komponen dengan lebih cepat dan responsif, serta mengoptimalkan penggunaan sumber daya yang lebih hemat.

c. Hasil Pengujian Chrome DevTools *Performance Tab*

Tabel 4.4 merangkum hasil pengujian menggunakan Chrome DevTools *Performance tab*, memberikan wawasan lebih mendalam tentang perilaku ketiga teknologi dalam kondisi *real-time*. VueJS menunjukkan performa terbaik dalam *Largest Contentful Paint* (LCP) dengan waktu 1,27 detik, mengungguli ReactJS (1,42 detik) dan Blade (9,83 detik). Hal ini menunjukkan bahwa VueJS sangat baik dalam memuat konten utama, memberikan pengalaman pengguna yang cepat dan responsif dalam kondisi *real-time*. Temuan ini memperkuat keunggulan VueJS dalam aplikasi yang membutuhkan pembaruan konten yang sering (*frequent*) dan halus (*smooth*). Dalam hal stabilitas tata letak, VueJS mencapai skor sempurna pada *Cumulative Layout Shift* (CLS) dengan nilai 0, mengungguli ReactJS (0,07) dan Blade (0,08). Hal ini menunjukkan kemampuan VueJS dalam menjaga konsistensi tampilan selama proses pemuatan dan interaksi pengguna, mencegah pergeseran konten yang dapat mengganggu pengalaman pengguna.

Keunggulan VueJS juga terlihat dari skor *Interaction to Next Paint* (INP) sebesar 40 milidetik, lebih baik dibandingkan Blade (48 milidetik) dan ReactJS (56 milidetik). Performa superior ini merupakan hasil dari optimasi sistem *event handling* dan mekanisme pembaruan DOM yang baik, memungkinkan VueJS memberikan respons yang sangat cepat terhadap interaksi pengguna. Meskipun Blade menunjukkan performa INP yang cukup baik untuk interaksi dasar, namun memiliki keterbatasan dalam menangani interaksi yang lebih kompleks. Sementara itu, meski memiliki waktu INP yang paling tinggi, ReactJS tetap mampu memberikan performa yang baik dalam menangani kompleksitas interaksi pada aplikasi dengan fitur yang dinamis.

Berdasarkan serangkaian pengujian yang komprehensif menggunakan Google Lighthouse dan Chrome DevTools *Performance tab* terhadap tiga teknologi *front-end* (Blade, ReactJS, dan VueJS), ditemukan bahwa setiap teknologi memiliki karakteristik dan keunggulan yang berbeda. VueJS menunjukkan performa terbaik secara keseluruhan dengan keunggulan yang konsisten dalam responsivitas, interaktivitas, dan kinerja *real-time*. Hal ini terlihat dari skor *Total Blocking Time* 0 milidetik dan *Interaction to Next Paint* terendah (40 milidetik), menjadikannya pilihan yang sesuai untuk aplikasi dengan antarmuka yang

memerlukan pembaruan *frequent* dan *smooth*, seperti *platform e-commerce* (Alokai, 2023), atau aplikasi media sosial yang membutuhkan pembaruan data secara *real-time* (Oshungboye, 2024) dan transisi antarmuka yang lancar.

Blade *Templating Engine* memperlihatkan keunggulan dalam kecepatan *rendering* awal dengan *First Contentful Paint* 1,9 detik dan skor aksesibilitas tertinggi (76), menjadikannya pilihan ideal untuk aplikasi berbasis konten statis yang membutuhkan kecepatan tampilan awal. Sementara itu, ReactJS, meskipun memiliki skor kinerja yang lebih rendah, menunjukkan keandalan dalam stabilitas tata letak dengan *Cumulative Layout Shift* bernilai 0 dan performa SEO yang sempurna (100), membuatnya cocok untuk aplikasi yang membutuhkan komponen dinamis yang kompleks (Prasatya, 2024).

Hasil pengujian tersebut menunjukkan bahwa pemilihan teknologi *front-end* dalam pengembangan aplikasi web sebaiknya disesuaikan dengan karakteristik dan kebutuhan spesifik proyek. Untuk aplikasi yang membutuhkan performa optimal dalam semua aspek, pendekatan *hybrid* yang menggabungkan kekuatan masing-masing teknologi dapat menjadi solusi. Misalnya, menggunakan Blade untuk halaman statis yang membutuhkan SEO optimal, sementara memanfaatkan VueJS atau ReactJS untuk bagian aplikasi yang memerlukan interaktivitas tinggi. Terlepas dari teknologi yang dipilih, optimasi berkelanjutan tetap menjadi kunci untuk mencapai performa maksimal. Hal ini mencakup implementasi *best practices* seperti *code splitting*, *lazy loading* berupa optimasi *asset*, *tree shaking*, dan *minification* (Akbarrizky et al., 2023), serta *monitoring* performa secara reguler menggunakan *tools* seperti Lighthouse dan Chrome DevTools. Dengan pemahaman mendalam tentang karakteristik dan *use case* yang sesuai untuk masing-masing teknologi, pengembang dapat membuat keputusan yang lebih informatif dalam memilih teknologi yang paling sesuai dengan kebutuhan proyek-proyek mereka.

Di sisi lain, keterbatasan dalam pengujian ini terletak pada ruang lingkup yang terfokus pada proyek *Talent Pool Platform*, sebuah ekosistem terintegrasi yang terdiri dari beberapa modul dengan fungsi dan tampilan berbeda. Untuk penelitian mendatang, disarankan melakukan pengujian pada aplikasi yang lebih setara dan kompleks guna memahami pengaruh kompleksitas dan ukuran aplikasi terhadap performa masing-masing *framework*, serta mengeksplorasi aspek integrasi teknologi *front-end* dalam konteks pengembangan aplikasi *mobile-first* atau *progressive web applications*.

Temuan ini membawa implikasi penting bagi pengembang dan organisasi dalam pengembangan aplikasi web modern. Kesuksesan pengembangan tidak hanya bergantung pada

pemilihan teknologi berdasarkan metrik kinerja, tetapi juga pada pemahaman mendalam tentang konteks penggunaan dan kebutuhan pengguna akhir. Faktor-faktor seperti kualitas kode, optimasi *database*, dan infrastruktur *server* juga memainkan peran krusial. Oleh karena itu, sebagaimana dikemukakan oleh Anggraeni et al. (2024), pemilihan teknologi harus dilakukan secara holistik dengan mempertimbangkan tidak hanya kinerja teknis tetapi juga kebutuhan spesifik proyek, keahlian tim, skalabilitas, dan kemudahan pemeliharaan.

Pendekatan *Hybrid* dalam Pengembangan Aplikasi Web

Berdasarkan serangkaian pengujian yang telah dilakukan pada tiga proyek yang masing-masing menggunakan teknologi *front-end* berbeda namun semuanya didukung oleh Laravel sebagai *back-end*, hasil analisis menunjukkan bahwa setiap teknologi memiliki karakteristik dan keunggulan yang berbeda dalam implementasinya. VueJS menunjukkan keunggulan signifikan dalam responsivitas dan interaktivitas, terutama dalam kecepatan pemrosesan dan respons terhadap interaksi pengguna. Blade *Templating Engine* unggul dalam kecepatan *rendering* awal dan aksesibilitas, membuatnya ideal untuk konten statis. Sementara ReactJS, meskipun memiliki skor kinerja yang lebih rendah, menunjukkan keandalan luar biasa dalam stabilitas tata letak dan performa SEO yang optimal. Perbedaan karakteristik ini, yang semuanya berhasil terintegrasi dengan Laravel sebagai *back-end*, menjadi dasar pertimbangan dalam pengembangan pendekatan *hybrid*.

Menurut Jamilatulain (2024), sistem *hybrid* adalah pendekatan yang menggabungkan dua atau lebih metode, teknologi, atau sistem untuk memanfaatkan keunggulan masing-masing dan mengatasi kekurangan dari pendekatan individu. Dalam konteks pengembangan web, pendekatan *hybrid* memungkinkan penggunaan *multiple framework* secara strategis berdasarkan kebutuhan spesifik proyek. Implementasi ini bukan berarti menggabungkan semua *framework* secara acak, melainkan memanfaatkan kelebihan masing-masing *framework* untuk mengoptimalkan performa aplikasi secara keseluruhan.

a. Implementasi Pendekatan *Hybrid* dalam Proyek

Berdasarkan hasil pengujian, implementasi pendekatan *hybrid* dapat dilakukan dengan menggunakan Blade untuk halaman statis yang membutuhkan SEO optimal dan *rendering* awal yang cepat. VueJS dapat dimanfaatkan untuk komponen yang memerlukan interaksi *real-time* dan pembaruan yang *frequent*, mengingat keunggulannya dalam pemrosesan dan responsivitas. Sementara ReactJS cocok digunakan untuk komponen kompleks yang membutuhkan stabilitas tata letak.

Implementasi pendekatan *hybrid* ini telah diterapkan secara nyata oleh PT Javan Cipta Solusi dalam pengembangan beberapa proyeknya termasuk pada proyek G. Proyek G merupakan Platform Ujian – Pelatihan *Coding* yang bertujuan memberikan fasilitas tambahan dalam pelaksanaan pelatihan yang bertema *coding*, sekaligus menyediakan media praktik bagi peserta untuk berlatih *coding* dasar tanpa memerlukan instalasi di perangkat mereka. Proyek ini merupakan contoh konkret penggunaan *multiple framework* secara strategis, mengombinasikan ReactJS, VueJS, dan Laravel sebagai *back-end* dalam satu aplikasi terintegrasi.

Penggunaan teknologi pada proyek G dipilih secara cermat berdasarkan kebutuhan spesifik setiap komponen. **ReactJS** diimplementasikan untuk antarmuka utama yang menampilkan informasi penting, seperti daftar pelatihan, *credentials* yang digunakan, dan informasi lainnya sebelum memulai praktik pelatihan secara langsung. Pemilihan ReactJS pada bagian ini didasarkan pada kebutuhannya akan tampilan yang menarik, dinamis, dan tata letak yang stabil. **VueJS** digunakan untuk fitur praktik *coding* yang membutuhkan respons *real-time*, seperti editor kode *online* dengan *timer* yang memungkinkan pengguna menulis dan menguji kode secara langsung dalam browser. Sementara itu, **Laravel** menangani komponen yang membutuhkan optimasi SEO. Pendekatan ini dikembangkan melalui proses riset dan diskusi mendalam untuk memenuhi kebutuhan arsitektur *microservice* yang diinginkan klien.

Arsitektur *hybrid* disusun dengan memisahkan tanggung jawab antara *back-end* dan *front-end* secara jelas. Laravel sebagai *back-end*, dapat menangani API *endpoints*, *database management*, *authentication*, dan *business logic*. Pada sisi *front-end*, ReactJS digunakan untuk komponen utama dan *state management* kompleks (Redux/Context API), dan VueJS untuk fitur interaktif dan *real-time*. Integrasi antar *framework* diatur melalui API Gateway yang berfungsi sebagai jembatan komunikasi, mengelola seluruh *request* dari ReactJS dan VueJS ke Laravel. Dalam implementasinya, struktur proyek diorganisir dengan memisahkan komponen-komponen berdasarkan *framework* untuk memudahkan pengembangan dan pemeliharaan. *Deployment* dilakukan secara terpisah untuk setiap *framework* dengan konfigurasi *environment* yang berbeda, memungkinkan fleksibilitas dalam pengelolaan versi dan pembaruan. Dengan arsitektur terdistribusi ini, tim pengembang dapat bekerja secara paralel pada komponen yang berbeda tanpa saling mengganggu, sekaligus mempertahankan integrasi yang mulus antar komponen melalui API yang telah didefinisikan dengan jelas.

b. Pertimbangan dan Optimasi

Implementasi pendekatan *hybrid* perlu mempertimbangkan beberapa aspek penting, termasuk kompleksitas integrasi antar *framework*, keahlian tim pengembang, *overhead maintenance*, dan kejelasan arsitektur. *Monitoring* dan evaluasi performa secara berkala menggunakan *tools* seperti Lighthouse dan Chrome DevTools menjadi kunci untuk mengukur dan mengevaluasi keberhasilan implementasi. Optimasi berkelanjutan juga diperlukan untuk tetap mempertahankan dan meningkatkan performa aplikasi seiring dengan perkembangannya.

Pemahaman karakteristik dan keunggulan masing-masing teknologi berdasarkan hasil pengujian memungkinkan pengembang membuat keputusan yang lebih informatif dalam mengimplementasikan pendekatan *hybrid*. Pemilihan teknologi yang tepat untuk setiap komponen aplikasi, didukung dengan implementasi *best practices* dan *monitoring* berkelanjutan, akan menghasilkan aplikasi web yang optimal dari segi performa, *user experience*, dan *maintainability*. Pendekatan *hybrid* menawarkan penyelesaian yang tepat untuk mengatasi tantangan teknis dalam pengembangan. Pendekatan ini juga memberikan fleksibilitas dalam menghadapi kompleksitas dan mengakomodasi kebutuhan yang beragam dalam pengembangan aplikasi web modern.

4.1.2 Dokumentasi proyek

Pengembangan sebuah proyek memerlukan standar dokumentasi yang jelas agar proses pengembangan perangkat lunak berjalan efektif. Untuk memenuhi standar ini, Synko & Peleshchyshyn (2020) mengidentifikasi beberapa jenis dokumentasi dasar yang penting dalam pengembangan perangkat lunak, seperti dokumentasi persyaratan (*requirements*), dokumentasi arsitektur/desain, dokumentasi teknis, dokumentasi untuk pengguna (*user-friendly documentation*), dan dokumentasi pemasaran.

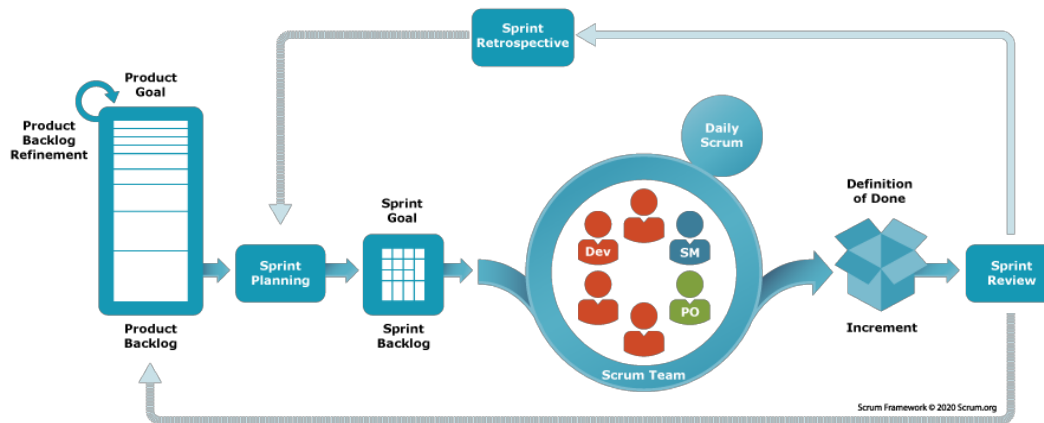
PT Javan Cipta Solusi telah menerapkan sebagian besar dari jenis dokumentasi tersebut, terutama dalam hal dokumentasi teknis. Menurut Synko & Peleshchyshyn (2020), dokumentasi teknis berhubungan dengan kode, algoritma, dan antarmuka program. Biasanya digunakan oleh pengembang untuk menjelaskan bagaimana API bekerja, bagaimana struktur data dibentuk, dan bagaimana algoritma dijalankan. Namun, di PT Javan Cipta Solusi, dokumentasi teknis tidak hanya mencakup hal-hal tersebut. Ruang lingkupnya jauh lebih luas dan mendalam, mencakup sembilan jenis dokumen teknis yang lebih lengkap, yaitu:

1. Dokumen arsitektur sistem, yang berisi diagram arsitektur sistem, ERD, skema basis data dan tabel, serta aliran data dan proses.
2. Dokumen spesifikasi fungsional, yang mencakup daftar fitur dan fungsi utama, deskripsi detail setiap fitur, *use case diagram* dengan langkah-langkah jelas, dan *user screen flow diagram*.
3. Dokumen konfigurasi, yang menjelaskan persyaratan perangkat keras dan perangkat lunak, pengaturan lingkungan pengembangan, pengujian, dan produksi, serta konfigurasi server dan sumber daya jaringan.
4. Dokumen instalasi, yang memuat langkah-langkah instalasi sistem dan kebutuhan khusus atau perubahan yang harus dilakukan pada lingkungan (*environment*).
5. Dokumen API, yang berisi daftar *endpoint* API, detail *request* dan *response* untuk setiap *endpoint*, serta contoh *output* setiap API.
6. Dokumen Keamanan, yang merinci langkah-langkah keamanan yang diimplementasikan, izin akses dan pengelolaan otorisasi, prosedur penanganan data sensitif, serta *report stress test*, *performance*, dan VA terbaru.
7. Dokumen pemeliharaan, yang meliputi panduan pemeliharaan rutin, penjadwalan tugas pemeliharaan, serta prosedur penanganan dan pemecahan masalah.
8. Dokumen *User Manual*, yang menyediakan panduan penggunaan sistem, tutorial langkah demi langkah untuk tugas umum, serta penjelasan antarmuka pengguna.
9. Dokumen *Source Code*, yang memuat struktur folder dan modul, serta keterangan singkat tentang setiap fungsi dan potongan kode.

Selama magang di PT Javan Cipta Solusi, penulis mengamati bahwa perusahaan ini telah menerapkan praktik dokumentasi yang baik. Penulis sendiri terlibat langsung dalam penyusunan dokumentasi *source code* untuk beberapa proyek. Dokumen tersebut dilengkapi dengan komentar dan penjelasan terkait struktur *file*, sehingga membantu memudahkan pemahaman tim dalam proses pengembangan. Dokumentasi ini sangat penting untuk mempermudah pemeliharaan dan pengembangan perangkat lunak di masa mendatang. Meskipun demikian, penulis juga mengamati bahwa dalam beberapa kasus, penyusunan dokumentasi *source code* sering kali terkendala oleh keterbatasan waktu. Hal ini dapat menyebabkan dokumentasi yang dihasilkan kurang lengkap atau tidak selalu *up-to-date*, yang pada akhirnya dapat menghambat tim dalam proses pengembangan di kemudian hari. Oleh karena itu, sangat penting untuk mengalokasikan waktu yang cukup dan memastikan seluruh anggota tim terlibat dalam proses dokumentasi. Selain itu, penulis juga mengamati bahwa

perhatian lebih perlu diberikan terhadap pengelolaan *timeline* dalam pembuatan dokumentasi untuk mencegah beban kerja yang berlebihan bahkan *overload*, sehingga penyusunan dokumentasi tidak mengganggu proses pengembangan perangkat lunak.

4.1.3 Metode Scrum



Gambar 4.13 Tahapan aktivitas dalam *scrum framework*

Sumber: (Scrum.org, 2020)

Gambar 4.13 memberikan gambaran menyeluruh tentang siklus hidup sebuah *sprint* dalam metode *Scrum*. Proses dimulai dengan perencanaan *sprint* (*sprint planning*) di mana tim menentukan tujuan dan tugas yang akan diselesaikan, dilanjutkan dengan pelaksanaan harian (*daily scrum*) untuk memantau progres, hingga akhirnya dilakukan evaluasi (*sprint review*) dan refleksi (*sprint retrospective*). Menurut Schwaber & Sutherland (2020), *sprint retrospective* adalah momen penting bagi tim *Scrum* untuk merefleksikan kinerja mereka selama satu *sprint*. Tujuan dari *sprint retrospective* adalah merencanakan cara untuk meningkatkan kualitas dan efektivitas. Dalam sesi ini, tim akan membahas apa yang berjalan dengan baik, masalah yang dihadapi, dan bagaimana masalah tersebut diselesaikan. Frekuensi *sprint retrospective* yang ideal adalah sekali setelah setiap *sprint*, memungkinkan tim untuk secara teratur melakukan penyesuaian dan perbaikan secara teratur. Melalui *retrospective* yang konsisten, tim dapat meningkatkan performa mereka dan menjaga proyek tetap sesuai rencana. Dengan demikian, tim *Scrum* dapat terus belajar, berkembang, dan mencapai tujuan mereka.

Pada praktiknya, *Scrum* menawarkan kerangka kerja yang fleksibel untuk tim pengembangan perangkat lunak. Namun, implementasinya dapat bervariasi di setiap organisasi. Sebagai contoh, di PT Javan Cipta Solusi, frekuensi pelaksanaan *sprint retrospective* tidak selalu mengikuti pedoman *Scrum* secara ketat. Berdasarkan pengamatan penulis selama magang, *sprint retrospective* di Javan sering kali dilakukan berdasarkan

kebutuhan, terutama ketika tim menghadapi tantangan atau masalah yang signifikan selama *sprint*. Hal ini berbeda dengan praktik Scrum pada umumnya, di mana *sprint retrospective* idealnya diadakan setelah setiap *sprint* untuk terus meningkatkan proses kerja tim secara berkesinambungan.

Perbedaan pendekatan ini dapat disebabkan oleh beberapa faktor, seperti ukuran tim, kompleksitas proyek, atau budaya organisasi. Meskipun demikian, penting untuk diingat bahwa tujuan utama dari *sprint retrospective* adalah perbaikan terus-menerus (Cohn et al., 2024). Terlepas dari frekuensinya, *sprint retrospective* tetap menjadi bagian penting dalam siklus pengembangan perangkat lunak menggunakan metode Scrum. *Sprint retrospective* merupakan alat yang berharga untuk mengidentifikasi masalah dan menemukan solusi (Schwaber & Sutherland, 2020). Oleh karena itu, penting bagi Javan untuk mempertimbangkan pelaksanaan *sprint retrospective* secara lebih teratur, meskipun dengan durasi yang lebih singkat atau format yang berbeda.

Retrospektif memberikan kesempatan bagi seluruh anggota tim untuk berbicara, termasuk mereka yang sering kali kurang berani mengemukakan pendapat atau masalah yang dialami selama *sprint*. Menurut Cohn et al. (2024), durasi ideal untuk *sprint retrospective* adalah 45 menit hingga 3 jam, tergantung pada durasi *sprint*. Misalnya, untuk *sprint* dua minggu, retrospektif dapat berlangsung 1,5 jam, sedangkan untuk *sprint* empat minggu, dapat berlangsung hingga tiga jam. Lokasi seharusnya tidak menjadi penghalang untuk pelaksanaan retrospektif, karena dengan banyaknya alat kolaborasi digital yang tersedia, sehingga tim *scrum* dapat berpartisipasi di mana pun mereka berada. Namun, Di PT Javan Cipta Solusi, *sprint retrospective* biasanya berlangsung selama 40 menit hingga 2 jam, menggunakan platform Zoom jika ada anggota tim yang bekerja dari rumah (WFH), atau dilakukan pertemuan langsung di luar kantor di tempat yang nyaman, seperti kafe dan tempat lainnya. Meskipun format dan durasi yang diterapkan sudah sesuai dengan kebutuhan tim, frekuensi retrospektif yang tidak selalu diadakan setelah setiap *sprint* dapat menghambat perbaikan berkelanjutan. Oleh karena itu, disarankan agar tim di Javan mengadakan *sprint retrospective* lebih teratur untuk mendukung perbaikan berkelanjutan dan memastikan proyek tetap sesuai jalur.

4.1.4 Tim Pengembangan Sistem

Menurut Kholin & Bitkina (2023), berbagai metode manajemen proyek sering diterapkan dalam pengembangan perangkat lunak, termasuk *Scrum*, *Kanban*, dan metodologi *Agile* lainnya. Di antara metode tersebut, *Scrum* menjadi salah satu pendekatan yang populer karena

bersifat iteratif dan inkremental. Metode ini dirancang untuk membantu tim mengelola proyek secara efektif dengan membagi pengembangan ke dalam *sprint*, sehingga tim dapat mengevaluasi hasil setiap iterasi secara berkala. Sebagai kerangka kerja yang terstruktur namun fleksibel, *Scrum* memberikan panduan bagi tim dalam menyelesaikan pekerjaan yang kompleks dan memungkinkan penyesuaian prioritas sesuai kebutuhan proyek.

Praktik penerapan metode *Scrum* melibatkan pembagian peran yang jelas untuk memastikan setiap anggota tim memahami tanggung jawab masing-masing. Secara umum, *Scrum* memiliki peran utama yang terdiri dari *Project Manager (PM)*, *Product Manager*, *Business Analyst (BA)*, *Backend Developers*, *Frontend Developers*, *Mobile Developers*, *UI/UX Designer*, *Tech Lead*, dan *Quality Assurance (QA) Engineers*. Selain itu, dalam beberapa kasus, terdapat pula peran opsional yang bisa ditambahkan sesuai kebutuhan proyek, seperti *Solution Architect*, *DevOps Engineer*, *VR Designer*, dan *3D Modeler* (Kholin & Bitkina, 2023). Keberadaan peran-peran ini membantu tim dalam memastikan kualitas serta efektivitas hasil akhir pengembangan.

PT Javan Cipta Solusi telah memiliki seluruh peran utama, termasuk *DevOps Engineer* sebagai peran opsional. Namun, terdapat satu peran utama yang tidak tersedia, yaitu *UI/UX Designer*. Berdasarkan informasi dari salah satu *Tech Lead* di PT Javan Cipta Solusi, ketidakadaan peran ini berkaitan dengan salah satu *library* yang dikembangkan oleh Javan, yaitu *Alurkerja*. *Alurkerja* merupakan sebuah aplikasi *library workflow* yang memudahkan untuk terintegrasi dengan Camunda ke dalam Laravel dan Springboot. Selain itu, *Alurkerja* menyediakan komponen-komponen yang telah terstandarisasi untuk digunakan oleh *front-end*. Oleh karena itu, peran *UI/UX Designer* tidak dianggap terlalu berkontribusi besar dalam pemodelan aplikasi, mengingat bahwa komponen yang diperlukan sudah tersedia melalui *Alurkerja*. Ketika terdapat kebutuhan tambahan terkait desain atau hal serupa, biasanya *wireframe* atau *mock-up* akan disediakan oleh *user* atau dibuat oleh *product owner* dan analis. Dalam hal ini, desain tersebut tetap mengikuti *design system* yang telah ditetapkan, termasuk warna dan komponen yang digunakan.

4.1.5 Evaluasi Supervisor

Evaluasi magang oleh supervisor dilakukan menggunakan Google Form yang terdiri dari dua kategori utama yaitu Kompetensi Teknis dan *Soft Skills*. *Form* ini menggunakan empat tingkat penilaian (Sangat Baik, Baik, Cukup, dan Perlu Perbaikan) serta dilengkapi kolom komentar untuk memberikan *feedback* spesifik pada setiap aspek. Evaluasi ini dilakukan oleh

dua supervisor yang memegang peran berbeda dalam struktur organisasi, yaitu *Operational Lead* dan *Tribe Lead* yang juga merangkap sebagai *Product Owner*. Evaluasi dari dua perspektif ini memberikan pandangan komprehensif mengenai kinerja penulis selama periode magang, mencakup aspek teknis dan *soft skills*.

Evaluasi oleh *Operational Lead*

Tabel 4.5 di bawah ini menunjukkan hasil evaluasi kinerja magang oleh Bapak Dicky Puja Pratama selaku *Operational Lead*. Penilaian ini mencakup aspek kompetensi teknis dan *soft skills* yang diamati selama masa magang.

Tabel 4.5 Hasil Evaluasi Magang oleh *Operational Lead*

Aspek Penilaian	Sangat Baik	Baik	Cukup	Perlu Perbaikan
Kompetensi Teknis				
Kemampuan teknis sesuai bidang kerja		✓		
Kualitas hasil pekerjaan		✓		
Pemahaman terhadap tugas		✓		
Soft Skills				
Kemampuan bekerja dalam tim		✓		
Komunikasi dan interpersonal		✓		
Inisiatif dan kemandirian		✓		
Kedisiplinan dan tanggung jawab		✓		

Tabel 4.5 menunjukkan bahwa *Operational Lead* memberikan penilaian yang konsisten dengan nilai “Baik” di seluruh aspek penilaian. *Operational Lead* memberikan nilai positif pada kemampuan adaptasi dan pembelajaran penulis untuk menyesuaikan diri dengan kebutuhan proyek dan kecepatan dalam mempelajari hal baru. Dalam evaluasinya, *Operational Lead* juga mengapresiasi kemampuan manajerial penulis, terutama dalam hal memprioritaskan pekerjaan dan pola kerja yang terstruktur. Kontribusi penulis dinilai cukup signifikan dalam membantu perkembangan proyek, termasuk kesediaan untuk mempelajari hal-hal baru yang relevan dengan proyek.

Catatan pengembangan yang diberikan mencakup peningkatan inisiatif dalam bertanya dan pemahaman mengenai *roadmap* pembelajaran. Pengembangan ini diharapkan mendukung penulis dalam meningkatkan kemampuan *knowledge sharing* kepada tim dan mendorong pengembangan tim secara keseluruhan.

Evaluasi oleh *Tribe Lead* (dan *Product Owner*)

Berikut adalah Tabel 4.6 yang menunjukkan hasil evaluasi kinerja magang oleh Mba Novia Endah Larasati selaku *Tribe Lead* yang juga berperan sebagai *Product Owner*. Posisi ganda ini memberikan penilaian yang komprehensif, baik dari sisi kepemimpinan tim maupun pengelolaan produk.

Tabel 4.6 Hasil Evaluasi Magang oleh *Tribe Lead*

Aspek Penilaian	Sangat Baik	Baik	Cukup	Perlu Perbaikan
Kompetensi Teknis				
Kemampuan teknis sesuai bidang kerja		✓		
Kualitas hasil pekerjaan	✓			
Pemahaman terhadap tugas	✓			
Soft Skills				
Kemampuan bekerja dalam tim	✓			
Komunikasi dan interpersonal		✓		
Inisiatif dan kemandirian	✓			
Kedisiplinan dan tanggung jawab	✓			

Tabel 4.6 menunjukkan bahwa evaluasi *Tribe Lead*, yang juga berperan sebagai *Product Owner*, memberikan nilai “Sangat Baik” dan “Baik” di beberapa aspek utama. Penulis dinilai berhasil memenuhi ekspektasi dengan baik, terutama dalam pengerjaan *task-task front-end* dan integrasi *front-end* dengan *back-end* pada proyek aplikasi. Kontribusi ini membantu percepatan proyek yang memiliki *deadline* ketat. Kualitas pekerjaan penulis secara keseluruhan sangat memuaskan, dengan hasil yang minim *bugs* dan selalu sesuai dengan *acceptance criteria*. Dalam pengerjaan proyek, *acceptance criteria* yang diberikan dapat dipenuhi dan diselesaikan tepat waktu sesuai dengan *mockup*, *mapping field*, dan *requirement* yang ditentukan. Kemampuan teknis di bidang *front-end development* menunjukkan hasil yang memuaskan, namun terdapat catatan untuk peningkatan kemampuan pada sisi *back-end* agar penulis memiliki fleksibilitas dalam mengerjakan tugas *back-end* jika diperlukan.

Dari segi *soft skills*, *Tribe Lead* memberikan apresiasi tinggi pada kedisiplinan dan tanggung jawab penulis yang konsisten selama pelaksanaan magang. Namun, ada catatan pengembangan dalam hal proaktivitas pada saat diskusi serta keberanian menyampaikan opini. *Tribe Lead* menyarankan agar penulis tidak perlu ragu atau khawatir dalam menyampaikan

pendapat dan ide pengembangan, terutama pada aspek-aspek yang berhubungan dengan *front-end* yang menjadi fokus utama penulis.

Penilaian Keseluruhan

Secara keseluruhan, kedua supervisor memberikan penilaian “Memuaskan” terhadap kinerja penulis selama periode magang, dengan catatan pengembangan yang berfokus pada peningkatan proaktivitas dan kemampuan *knowledge sharing* dengan tim. Evaluasi ini mencerminkan kinerja penulis yang solid dalam memenuhi target proyek serta menunjukkan area-area yang dapat dikembangkan lebih lanjut untuk mendukung kesuksesan karier di masa depan. Supervisor berharap komitmen yang ditunjukkan akan berdampak positif pada perkembangan karier dan mendukung pertumbuhan penulis di bidang yang ingin ditekuni.

4.2 Manfaat Magang Bagi Mahasiswa

Selama sekitar enam bulan menjalani magang di PT Javan Cipta Solusi, penulis mendapatkan banyak manfaat yang berharga. Manfaat utama dari program magang ini adalah kesempatan untuk merasakan pengalaman kerja langsung di lingkungan profesional, memberikan gambaran nyata tentang dunia kerja serta tantangan yang dihadapi. Pengalaman ini tentu menjadi bekal penting ketika penulis memasuki dunia kerja yang sesungguhnya. Selain itu, magang ini juga memberikan manfaat lain, seperti peningkatan keterampilan teknis, pengembangan *soft skills*, dan pemahaman yang lebih mendalam mengenai proses kerja dalam tim. Semua manfaat tersebut akan dijelaskan lebih rinci pada subbab berikutnya.

4.2.1 Magang di Perusahaan *Software House*

Magang di perusahaan *software house* seperti PT Javan Cipta Solusi memberikan pengalaman yang sangat beragam dan menantang. Penulis terlibat langsung dalam pengembangan perangkat lunak dengan tim yang menggunakan metodologi *Scrum*, yang memungkinkan penulis untuk memahami alur kerja pengembangan aplikasi secara lebih mendalam. Proses pengembangan yang dinamis dan sering kali menuntut adaptasi cepat terhadap perubahan dan memberikan wawasan baru tentang manajemen proyek dalam dunia perangkat lunak. Selama enam bulan masa magang, penulis terlibat dalam delapan proyek berbeda. Dari delapan proyek tersebut, lima di antaranya melibatkan pengembangan langsung, sementara tiga lainnya lebih berfokus pada pembuatan dokumentasi *source code*. Salah satu tantangan terbesar yang dihadapi adalah penulis sering kali dipindah-pindahkan dari satu

proyek ke proyek lain, biasanya setiap satu atau dua bulan. Awalnya, hal ini menimbulkan kebingungan dan kesulitan untuk menyesuaikan diri, tetapi seiring waktu, penulis menyadari bahwa ini merupakan bagian dari tuntutan untuk dapat beradaptasi dengan cepat terhadap kebutuhan perusahaan dan proyek yang sedang berjalan.

Pengalaman ini menuntut penulis untuk tidak hanya mengembangkan keterampilan teknis, tetapi juga kemampuan belajar yang cepat dan adaptasi terhadap teknologi yang berbeda. Pada awal magang, penulis dihadapkan dengan proyek yang menggunakan *tech stack* Laravel dan Blade, yang berbeda dengan latar belakang penulis sebagai *frontend* (React) *programmer*. Hal ini menambah tantangan, karena penulis belum pernah menggunakan Laravel dan Blade sebelumnya. Dalam kondisi tersebut, penulis harus secara mandiri mencari tahu, mempelajari teknologi baru, serta berkonsultasi dengan mentor untuk memahami proyek yang sedang dihadapi. Selain itu, pengalaman mengerjakan dokumentasi *source code* di tengah magang juga memberikan perspektif baru. Pada awalnya, hal ini membuat penulis merasa kurang optimal karena tidak terlibat langsung dalam proses pengembangan atau *coding*. Penulis dihadapkan dengan tugas administratif dan tidak sesuai ekspektasi awal yang lebih berorientasi pada pengembangan kode. Namun, hal ini menjadi pembelajaran penting bahwa kontribusi di dalam proyek tidak selalu berbentuk pengembangan langsung, tetapi juga melalui dokumentasi yang merupakan bagian esensial dari pengembangan proyek, untuk memastikan keberlanjutan dan mempermudah pemahaman tim pengembang di masa depan.

4.2.2 Pengenalan Lingkungan Perusahaan

Selama magang, penulis banyak belajar mengenai peran-peran (*roles*) dalam tim operasional di PT Javan Cipta Solusi. Penulis baru mengetahui bahwa ternyata ada sistem *holacracy*, suatu manajemen sistem yang menggunakan model struktur *self-management organization*, yang menjelaskan struktur organisasi dengan bentuk lingkaran besar/*circle*. Penulis tergabung dalam *circle* tim operasional, yang terdiri dari beberapa *tribe lead*. Setiap *tribe lead* menangani klien dan proyek yang berbeda-beda, dengan tim pengembangannya masing-masing. Dalam satu *tribe*, terdapat tanggung jawab untuk mengelola beberapa proyek, di mana setiap tim proyek terdiri dari *product owner* (PO), analis, *programmers*, dan *quality assurance* (QA).

Interaksi harian dengan berbagai divisi, seperti tim *programmer*, *analyst*, dan QA memberikan pelajaran berharga tentang pentingnya komunikasi dan kolaborasi dalam menyelesaikan proyek. Meskipun penulis pernah merasa kewalahan dengan dinamika

perpindahan proyek serta tanggung jawab yang tidak sesuai ekspektasi, pengalaman ini justru memberikan pelajaran penting tentang fleksibilitas, ketahanan, dan kemampuan untuk cepat mempelajari hal baru serta beradaptasi. Keterampilan ini sangat dibutuhkan di dunia kerja, terutama di industri teknologi yang terus berkembang pesat. Penulis juga belajar untuk bekerja di luar zona nyaman, menghadapi tantangan yang tidak hanya teknis tetapi juga terkait dengan manajemen waktu dan peran yang terus berubah. Pengalaman ini telah mempersiapkan penulis untuk menghadapi tantangan serupa di masa mendatang, di mana ketidakpastian dan perubahan sering kali menjadi hal yang tak terhindarkan dalam dunia teknologi.

Selain itu, penulis juga memperoleh pengetahuan baru terkait berbagai *tools* dan teknologi yang digunakan dalam pengembangan proyek. Berbagai teknologi seperti Taiga dan Active Collab untuk manajemen proyek, Metabase untuk *monitoring*, GitLab, DBeaver, Postman, Apidog, dan teknologi lainnya, memberikan kemudahan dalam menyelesaikan pekerjaan. Pengetahuan ini akan sangat berguna di masa depan, ketika penulis harus mengelola proyek dengan efisiensi tinggi.

4.2.3 Mempelajari dasar-dasar BPMN

Selama magang, penulis mempelajari dasar-dasar *Business Process Model and Notation* (BPMN) yang memberikan pemahaman mendasar tentang cara mengoptimalkan proses bisnis. Salah satu *tool* yang digunakan untuk membuat dan memvisualisasikan BPMN adalah Camunda, yang juga dapat di-*deploy* dan diintegrasikan dengan sistem untuk meningkatkan efisiensi. Memahami BPMN menjadi penting karena PT Javan Cipta Solusi merupakan *Business Process Optimization Partner* yang secara intensif menggunakan BPMN dan Camunda dalam setiap proyek pengembangan proses bisnis. Dengan mempelajari ini, penulis memperoleh kompetensi yang merupakan standar utama di perusahaan.

4.2.4 Mengembangkan Proyek

Pengalaman mengembangkan proyek menggunakan metode Scrum sangat terasa selama magang di PT Javan Cipta Solusi. Jika sebelumnya penulis hanya mengenal Scrum secara teori dari perkuliahan, di sini penulis dapat menerapkannya secara langsung. Setiap tahapan dalam Scrum, mulai dari *sprint planning*, *sprint backlog*, *daily scrum*, hingga *sprint retrospective* dijalankan dengan disiplin. Selain itu, setelah *sprint planning* dan *sprint backlog* selesai, tim operasional juga harus mematuhi skema pemberian *task*. Setiap pengembang harus dipastikan tidak kekurangan *task* dan memiliki target harian yang jelas. Pemberian *task* pun memiliki

aturan tertentu, di mana *task* didistribusikan oleh analis melalui Taiga, dengan *due date* minimal dua hari setelah tugas diberikan. Pengalaman menjalani proyek dengan alur yang terstruktur ini memberikan penulis pemahaman yang mendalam bahwa pengembangan proyek tidak hanya membutuhkan keterampilan teknis, tetapi juga manajemen waktu dan ketepatan dalam eksekusi.

Selain itu, aspek pengujian juga menjadi bagian penting dalam pengembangan proyek. Beberapa pengujian yang dilakukan termasuk *unit test* menggunakan Sonar untuk analisis kualitas kode dan *functional test* menggunakan Apidog. Sonar membantu memastikan kualitas kode tetap terjaga, sementara Apidog digunakan untuk menguji fungsionalitas API secara otomatis dan berkala, tanpa perlu keterlibatan manusia. Kedua jenis pengujian ini dilakukan oleh *programmers* yang tergabung dalam masing-masing proyek. Setiap tim bertanggung jawab penuh dalam menjaga performa dan kualitas kode proyek yang sedang mereka kembangkan secara konsisten.

4.2.5 *Meetings*

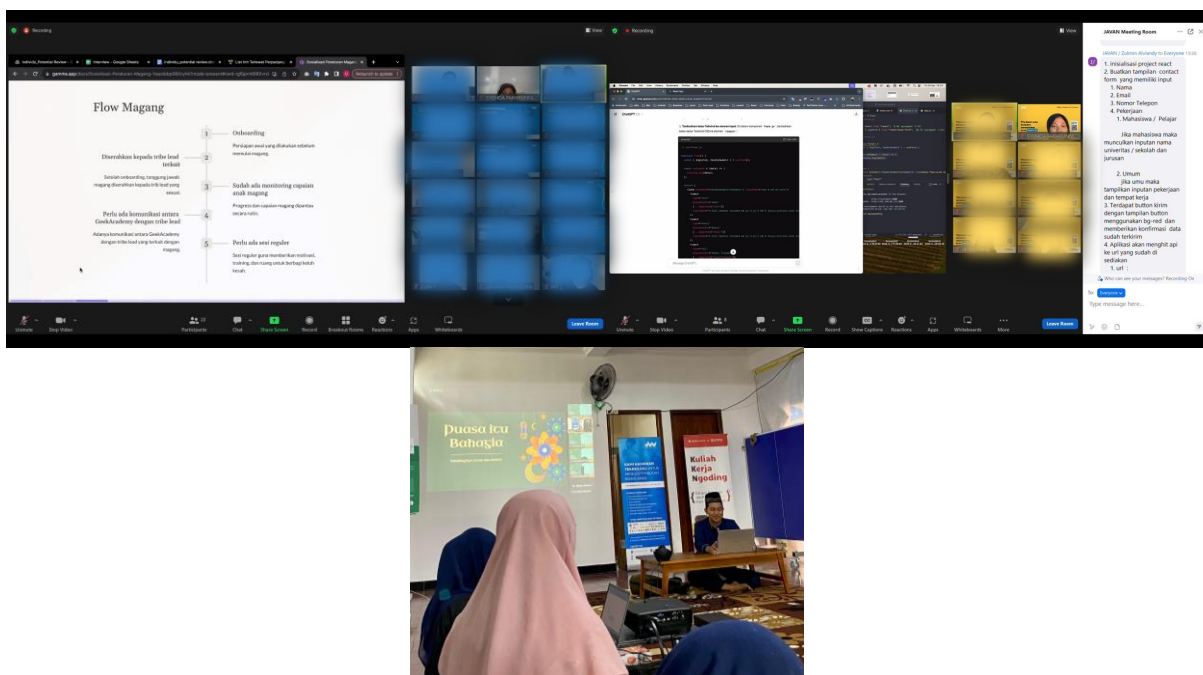
Selama magang, penulis mengikuti berbagai jenis meeting dalam proses pengembangan proyek, mulai dari *governance meeting*, *tactical meeting*, hingga meetings dalam metode Scrum. *Governance meeting* diadakan lebih jarang dibanding *tactical meeting*. Sesi ini digunakan untuk membahas hal-hal yang struktural dalam *circle*, seperti perubahan tanggung jawab atau penambahan anggota baru dalam tim. Contohnya, ketika penulis bergabung dalam tim proyek, *governance meeting* diadakan untuk memperkenalkan penulis sebagai anggota baru di tim tersebut. Sementara itu, *tactical meeting* diadakan secara rutin oleh *tribe lead* untuk membahas hal-hal terkait kendala operasional, perencanaan proyek, dan tindakan yang perlu diambil. Hasil dari *tactical meeting* ini dapat berupa tugas jangka pendek ataupun proyek jangka panjang, tergantung kebutuhan dan prioritas tim.

Pada metode *Scrum*, berbagai jenis *meeting* juga diterapkan, seperti perencanaan *sprint* (*sprint planning*), pelaksanaan harian (*daily scrum*), evaluasi (*sprint review*), dan refleksi (*sprint retrospective*). Pada *sprint planning*, tim menentukan tujuan dan tugas yang harus diselesaikan dalam periode *sprint*. Setiap hari, tim melakukan *daily scrum* untuk memantau kemajuan tugas, dan di akhir *sprint* diadakan *sprint review* untuk mengevaluasi hasil, diikuti dengan *sprint retrospective*. Sesi *retrospective* ini bertujuan untuk merefleksikan apa yang telah berjalan dengan baik, tantangan yang dihadapi, serta langkah-langkah perbaikan ke depannya. Setiap anggota tim biasanya diberikan kesempatan untuk bercerita apa pun dan

menyampaikan pandangan maupun kendala yang dihadapi selama *sprint*. Sesi ini umumnya dipimpin oleh *tribe lead* dan wajib dihadiri oleh seluruh anggota tim proyek yang terlibat. Selain itu, apabila diperlukan, pihak manajemen atau petinggi perusahaan juga dapat turut hadir untuk memberikan masukan dan perspektif tambahan terkait hasil *sprint*.

4.2.6 Hari Berkualitas

PT Javan Cipta Solusi juga memiliki tradisi mingguan yang disebut ‘Hari Berkualitas’, yang diadakan setiap Jumat. Hari ini diisi dengan berbagai kegiatan yang bermanfaat seperti *meeting*, *workshop*, pelatihan, *town hall*, ataupun kegiatan lain yang bertujuan meningkatkan kompetensi karyawan. Kegiatan ini juga membantu mempererat hubungan antar-karyawan serta memberikan kesempatan untuk pengembangan *soft skills* dan *hard skills*. Gambar 4.14 dan Gambar 4.15 menunjukkan contoh kegiatan yang diadakan pada ‘Hari Berkualitas’. Beberapa kegiatan yang terdokumentasi dalam Gambar 4.14 meliputi pelatihan *coding* dengan AI, sosialisasi magang, dan kegiatan di bulan Ramadan yang diisi oleh narasumber ahli di bidangnya. Pada sesi sosialisasi magang, yang wajib dihadiri oleh seluruh peserta magang, dibahas berbagai hal terkait program magang seperti *flow* magang, sistem insentif, dan poin-poin penting lainnya. Sementara itu, pelatihan *coding* yang dipimpin oleh *Tech Lead* dibagi ke beberapa *room* sesuai dengan *tech lead* masing-masing dan setiap *tech lead* akan menjelaskan alurnya dan diakhiri dengan praktik *coding with AI* yang hasilnya wajib dikumpulkan.



Gambar 4.14 Kegiatan-kegiatan yang diadakan pada ‘Hari Berkualitas’

Gambar 4.15 menunjukkan contoh *task* yang diberikan melalui ActiveCollab untuk pelatihan BPMN. Setiap karyawan diwajibkan mengikuti tes ini untuk mempelajari secara lebih mendalam terkait BPMN dan akan diberikan *due date* pengerjaannya.

The screenshot shows a task in ActiveCollab with the following details:

- Title:** Tersedia update progress self-paced learning BPMN - Syuhda Fakhrunnisa
- Description:** Dear tim Merindaklanjuti kegiatan training BPMN kemarin, dimohon untuk semua peserta Programmer menonton video BPMN UdemY dan menyelesaikan exercise yang telah diberikan.
- Details:** Detail jadwal, link video, akan untuk akses video, dan ketentuan lain ada di spreadsheet berikut: https://docs.google.com/spreadsheets/d/1uEplRUK319gy_o0l8aF3H2HJULClckam8D7KcNpAaYQ/edit?usp=sharing
- Subtasks:**
 - Berhasil mengerjakan exercise
 - Berhasil menonton video udemy
- Discussion:**
 - Syuhda F. Jul 23, 05:33 - #5: Berhasil mengerjakan seluruh exercise dan telah diunggah ke GDrive (with screenshot of a Google Drive folder containing 'TUGAS 5 BPMN - Exercise 5')
 - Nadia A. Jul 22, 14:51 - #4: okee mbas nanti malam aku lanjut kerjain yaa
 - Nadia A. Jul 22, 13:17 - #3: Halo @Syuhda Fakhrunnisa Minta tolong untuk segera diselesaikan tugas exercise nya ya Maksimal hari ini
 - Nadia A. Jul 19, 08:10 - #2: sip mantap monggo dilanjut mengerjakan exercise ya @Syuhda Fakhrunnisa
 - Syuhda F. Jul 19, 07:25 - #1: Berhasil menonton video UdemY (with screenshots of Udemy course progress)
- Task Management (Right Sidebar):**
 - Assignee: Syuhda Fakhrunnisa
 - Start and Due Date: Jul 18 - Jul 20
 - Time Estimation: 0:00
 - Expense Tracking: 0.00
 - Task is billable:
 - Subscribers: Nadia Alkatsiri, Syuhda Fakhrunnisa
 - Reminders: Add...

Gambar 4.15 Pelatihan BPMN yang diadakan pada ‘Hari Berkualitas’

BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan hasil magang di PT Javan Cipta Solusi dan analisis performa berbagai *framework* dalam pengembangan aplikasi web, disimpulkan bahwa pemilihan teknologi *framework (front-end)*, seperti ReactJS, VueJS, dan Blade *Templating Engine*, berperan signifikan dalam mendukung performa aplikasi yang optimal. Hasil pengujian menggunakan alat analisis berupa Google Lighthouse dan Chrome DevTools menunjukkan bahwa setiap *framework* memiliki karakteristik dan kelebihan dalam aspek tertentu. ReactJS, yang sering dipilih oleh PT Javan Cipta Solusi karena dukungan *library* internal seperti Alurkerja, unggul dalam stabilitas tata letak dan performa SEO, sementara VueJS menonjol dalam hal responsivitas serta fleksibilitas. Di sisi lain, Blade *Templating Engine* memberikan keunggulan dalam pemuatan konten pertama, sehingga lebih cocok untuk aplikasi dengan kebutuhan akses cepat pada konten awal. Hasil analisis juga menunjukkan bahwa pendekatan *hybrid* yang menggabungkan kelebihan dari berbagai *framework* dalam satu proyek dapat menjadi solusi yang optimal, terutama untuk aplikasi yang membutuhkan performa tinggi pada komponen-komponen yang berbeda.

Kesimpulan utama yang diperoleh adalah pemilihan teknologi harus disesuaikan dengan kebutuhan dan karakteristik proyek tertentu, baik dalam hal kinerja maupun kemudahan pengelolaan. Namun, pemilihan tetap harus dilakukan secara holistik dengan memerhatikan faktor-faktor lain yang akan memengaruhi aplikasi, agar dapat mendukung kebutuhan klien serta arsitektur aplikasi yang diinginkan. Selama masa magang, penulis juga memperoleh pengalaman langsung dalam penyusunan dokumentasi, implementasi metode *Scrum*, serta penggunaan berbagai alat pengujian dan pengembangan yang dapat menjadi faktor pendukung dalam peningkatan kualitas kerja dalam tim proyek.

5.2 Saran

Berdasarkan pengalaman magang dan hasil analisis yang diperoleh, terdapat beberapa aspek yang dapat dioptimalkan dalam proses pengembangan di perusahaan, terutama terkait pemilihan teknologi dan pengelolaan dokumentasi. Untuk meningkatkan kualitas pengembangan aplikasi di masa mendatang, berikut adalah beberapa saran yang diharapkan dapat mendukung kinerja perusahaan:

a. Melakukan Evaluasi Berkala Terhadap Teknologi

Evaluasi rutin terhadap teknologi *framework* dan alat bantu pengembangan menjadi sangat penting mengingat pesatnya perkembangan teknologi saat ini. Langkah ini diperlukan untuk memastikan bahwa teknologi yang digunakan dalam pengembangan proyek tetap relevan dan optimal. Berdasarkan hasil pengujian, setiap *framework* memiliki karakteristik dan keunggulan masing-masing dalam aspek tertentu. Dengan demikian, pemilihan *framework* dan teknologi perlu mempertimbangkan berbagai aspek, baik dari segi skalabilitas maupun kompleksitas aplikasi, termasuk kemungkinan penerapan pendekatan *hybrid* yang menggabungkan kelebihan beberapa *framework* dalam satu proyek sesuai kebutuhan spesifik setiap komponennya. Melalui evaluasi berkala ini, perusahaan dapat memastikan bahwa teknologi yang digunakan tetap sesuai dengan kebutuhan pengembangan dan standar industri terkini.

b. Pengembangan dan Pelatihan Dokumentasi Proyek

Dokumentasi yang lengkap dan terstruktur merupakan komponen penting dalam pengembangan perangkat lunak. Untuk itu, diperlukan peningkatan dalam hal pelatihan dan pengembangan keterampilan dokumentasi bagi tim, sehingga memastikan seluruh dokumentasi tetap *up-to-date* dan mudah dipahami oleh anggota tim yang baru bergabung.

c. Optimasi Proses Pengujian

Untuk meningkatkan kualitas pengembangan, disarankan memperluas penerapan alat pengujian otomatis dan mendorong praktik pengujian berkala untuk setiap fase pengembangan, serta pengujian yang komprehensif terhadap integrasi antar *framework*. Langkah ini akan membantu menjaga kualitas dan performa aplikasi secara keseluruhan serta mengurangi potensi kesalahan yang dapat muncul di tahap produksi.

d. Pengelolaan *Task* yang Lebih Efektif

Untuk menghindari beban kerja yang berlebihan pada anggota tim dan memastikan distribusi pekerjaan yang lebih seimbang, perusahaan disarankan menerapkan sistem pengelolaan *task* yang lebih baik. Hal ini dapat berupa pengaturan prioritas *task*, pembagian beban kerja yang adil, dan alokasi waktu yang memadai untuk penyelesaian dokumentasi. Pengelolaan yang lebih terstruktur akan membantu tim menjaga kualitas pekerjaan tanpa mengorbankan produktivitas.

Penerapan rekomendasi ini diharapkan dapat mendukung perusahaan dalam meningkatkan pengelolaan proyek dan kualitas produk yang dihasilkan, sehingga perusahaan tetap kompetitif dan mampu bersaing di tengah tuntutan industri yang semakin tinggi.

DAFTAR PUSTAKA

- Adhitya, W. R., Teviana, T., Sienny, S., Hidayat, A., & Khaira, I. (2024). Implementasi Digital Marketing Menggunakan Platform E-Commerce dan Media Sosial Terhadap Masyarakat Dalam Melakukan Pembelian. *TIN: Terapan Informatika Nusantara*, 5(1), 63–72. <https://doi.org/10.47065/tin.v5i1.5293>
- Adli, A., Wihayanti, T., & Witarsyah, D. (2023). *Top 5 E-Commerce Performance Analysis Using Google Lighthouse Matrix* (Issue Icoeins). Atlantis Press International BV. https://doi.org/10.2991/978-94-6463-340-5_21
- Akbarrizky, A., Daman Huri, A. R., Wisnuadhi, B., & Firdaus, L. H. (2023). Optimasi Micro Frontend Website untuk Meningkatkan Load Times: Teknik, Tantangan, dan Best Practice. *Jurnal CoSciTech (Computer Science and Information Technology)*, 4(2), 366–375. <https://doi.org/10.37859/coscitech.v4i2.5205>
- Alokai. (2023). *6 tips for ensuring fast Vue.js performance: The essential guide to Vue*. <https://alokai.com/blog/good-performance-with-vue-js>
- Anggraeni, O. S. I., Sugiarto, L., & Agustin, T. (2024). *Studi Komparatif Performa Framework Javascript Modern dalam Pengembangan Aplikasi Web*. 2(4), 162–177. <https://doi.org/https://doi.org/10.62951/modem.v2i4.239>
- Antoro, D., Anwar, N., Bahrul Ulum, M., Mulyo Widodo, A., & Erzed, N. (2022). Rancang Bangun Sistem Penggajian Karyawan Menggunakan Metode Rapid Application Development (RAD). *Ikraith-Informatika*, 7(1), 76–81. <https://doi.org/10.37817/ikraith-informatika.v7i1.2238>
- Chastro, C., & Darmawan, E. (2020). Perbandingan Pengembangan Front End Menggunakan Blade Template dan Vue Js. *Jurnal Strategi*, 2(2), 302–313.
- Cohn, M., Dabrytski, P., & Watts, G. (2024). *The Scrum Events and How They Work Together*. Scrum Alliance Inc. <https://resources.scrumalliance.org/Article/scrum-events>
- CoreUI. (2022). *How to create a single page application using React.js*. <https://coreui.io/blog/how-to-create-a-single-page-application-using-reactjs/>
- Developers, C. for. (2019a). *Lighthouse Accessibility Score*. <https://developer.chrome.com/docs/lighthouse/accessibility/scoring>
- Developers, C. for. (2019b). *Lighthouse Performance Scoring*. <https://developer.chrome.com/docs/lighthouse/performance/performance-scoring>
- Endra, R. Y., Aprilinda, Y., Dharmawan, Y. Y., & Ramadhan, W. (2021). Analisis

- Perbandingan Bahasa Pemrograman PHP Laravel dengan PHP Native pada Pengembangan Website. *EXPERT: Jurnal Manajemen Sistem Informasi Dan Teknologi*, 11(1), 48. <https://doi.org/10.36448/expert.v11i1.2012>
- Erolin, J. (2023). *React Single Page Application*. <https://www.bairesdev.com/blog/react-spa-single-page-application/>
- GoogleChrome. (2024a). *Lighthouse*. <https://github.com/GoogleChrome/lighthouse/blob/main/core/config/default-config.js>
- GoogleChrome. (2024b). *Lighthouse Scores*. <https://github.com/GoogleChrome/lighthouse/blob/main/docs/scoring.md>
- Hadi, A. P. (2024). *Mengenal Frontend Development* (Rudjiono (ed.)). Yayasan Prima Agus Teknik.
- Jamilatulain. (2024). *Pengertian Sistem Hybrid*. RedaSamudera.Id. <https://redasamudera.id/pengertian-sistem-hybrid/>
- Kaleev, N. (2024). *How to Use Chrome DevTools for Performance Audits*. <https://nitropack.io/blog/post/chrome-devtools-performance-tab>
- Kamal, N. 'Allam. (2022). Manajemen Sistem Organisasi Holacracy Perusahaan Software House Menggunakan Aplikasi Grassfrog. *Automata*, 3(2).
- Karić, A., & Durmić, N. (2024). Comparison of JavaScript Frontend Frameworks – Angular, React, and Vue. *International Journal of Innovative Science and Research Technology (IJISRT)*, 9(6), 1383–1390. <https://doi.org/10.38124/ijisrt/ijisrt24jun600>
- Khoirurrisal, M. F., Hidayat, C. R., & Ruuhwan, R. (2024). Analisis Perbandingan Framework Front-End Javascript Solidjs Dan Vuejs Pada Pengembangan Website Interaktif. *Jurnal Informatika Dan Teknik Elektro Terapan*, 12(2). <https://doi.org/10.23960/jitet.v12i2.4106>
- Kholin, S., & Bitkina, A. (2023). *Ideal Software Development Team Size & Roles for Your Project*. Onix Systems. <https://onix-systems.com/blog/software-development-team-size-and-roles>
- Kravchenko, Y., Leshchenko, O., Trush, O., Dakhno, N., & Krasnopyorov, P. (2023). Optimizing and Improvement a Web Application Using Open Source Tools. *CEUR Workshop Proceedings*, 3624, 368–379.
- Kurniawan, H., Syafa'at, F., Budihartono, E., Lorosae, T. A., Apriana, D., Marisa, M., Carudin, C., Adhichandra, I., Syaddad, H. N., & Ikhsan, M. (2023). *Belajar Web Programming : Referensi Pengenalan Dasar Tahapan Belajar Pemrograman Web Untuk Pemula* (Issue August). <https://books.google.co.id/books?id=gs3OEAAAQBAJ>

- Long, M. (2023). *How to Create a Single Page Application Using ReactJS*. <https://groovetechnology.com/blog/technologies/how-to-create-a-single-page-application-using-reactjs/>
- Luhukay, P. K., & Mailoa, E. (2024). *PENGEMBANGAN SISTEM INFORMASI PENJUALAN MENGGUNAKAN FRAMEWORK LARAVEL (STUDI KASUS : BUTIK FANTHY MODESTE)*. 7(2), 234–240.
- Metabase. (2024). *Metabase documentation*. <https://www.metabase.com/docs/latest/>
- Muna, S. S., Nurdin, N., & Taufiq, T. (2022). Tokopedia and Shopee Marketplace Performance Analysis Using Metrix Google Light-house. *International Journal of Engineering, Science and Information Technology*, 2(3), 106–110. <https://doi.org/10.52088/ijesty.v2i3.312>
- Oktavanusa, N. H. (2022). *Pengembangan Aplikasi Balelabs Billing Dan Globalplanet News Di Pt Bale Lab Indonesia*. Universitas Islam Indonesia.
- Oshungboye, O. (2024). *How to Build a Real-time Chat App with Laravel, Vue.js, and Pusher*. Twilio. <https://www.twilio.com/en-us/blog/build-real-time-chat-app-laravel-vuejs-pusher>
- Pardosi, V. B. A., Nooriansyah, S., Nadziroh, F., Kusmiati, H., Hendrawan, L., Kurniati, Puspitarini, E. W., Bakti, I., Anwar, & Firdaus, M. (2024). *Merintis Dunia Digital dengan Esensi Desain Web*. CV Rey Media Grafika. https://books.google.co.id/books?hl=en&lr=&id=ZpYREQAAQBAJ&oi=fnd&pg=PA1&dq=MERINTIS+DUNIA+DIGITAL+DENGAN+ESENSI+DESAIN+WEB&ots=4HECqi5xD-&sig=ih6KaoWf3SE9IUypZtCUxmJCM2Y&redir_esc=y#v=onepage&q=MERINTIS DUNIA DIGITAL DENGAN ESENSI DESAIN WEB&f=false
- Pasaribu, J. S. (2017). Penerapan Framework Yii Pada Pembangunan Sistem Ppdb Smp Bppi Baleendah Kabupaten Bandung. *Jurnal Ilmiah Teknologi Infomasi Terapan*, 3(2), 154–163. <https://doi.org/10.33197/jitter.vol3.iss2.2017.132>
- Prasatya. (2024). *Pengertian, Fungsi, Contoh, dan Install React Js Adalah*. Codepolitan. <https://www.codepolitan.com/blog/pengertian-fungsi-contoh-dan-install-react-js-adalah/>
- Prasetyo, S. M., Nugroho, M. I. P., Putri, R. L., & Fauzi, O. (2022). Pembahasan Mengenai Front-End Web Developer dalam Ruang Lingkup Web Development. *Jurnal Multidisiplin Ilmu*, 1(6), 1015–1020. <https://journal.mediapublikasi.id/index.php/bullet>
- Roslani, E., Budiman, T., & Hidayat, S. (2023). Design And Build A Laundry Service

- Ordering Information System Using Framework Spring Boot And Vue Js (Case Study Qu Cuci Garut). *JURNAL RISTEC : Research in Information Systems and Technology*, 4(2).
- Sa'ad, M. I. (2020). *Otodidak Web Programming: Membuat Website Edutainment*. PT Elex Media Komputindo.
https://books.google.co.id/books?hl=en&lr=&id=I73NDwAAQBAJ&oi=fnd&pg=PP1&dq=Otodidak+Web+Programming:+Membuat+Website+Edutainment&ots=1U1-LlPyNh&sig=WOJVrElrWzeVfLGxgISrf3q67zw&redir_esc=y#v=onepage&q=Otodidak+Web+Programming%3A+Membuat+Website+Edutainment
- Schwaber, K., & Sutherland, J. (2020). The Scrum Guide. In *The Definitive Guide to Scrum: The Rules of the Game* (Issue November). Scrum.org.
<https://creativecommons.org/licenses/by-sa/4.0/legalcode>
- Scrum.org. (2020). *What is Scrum?* <https://www.scrum.org/resources/what-scrum-module>
- Siahaan, M., & Vianto, V. O. (2022). Comparative Analysis Study of Front-End JavaScript Frameworks Performance Using Lighthouse Tool. *Jurnal Mantik*, 6(3), 2685–4236.
- Sinlae, F., Steno Birama, P., Ardian, D., Siregar, N., Safriadi, W., & Tawakal, H. (2024). Design dan Implementasi Sistem Informasi Pembelian Properti Berbasis Web Menggunakan Framework Laravel. *Jurnal Siber Multi Disiplin (JSMD)*, 2(2), 152–157.
<https://creativecommons.org/licenses/by/4.0/>
- Sofi'ie, F. A. F., & Qoiriah, A. (2023). Analisis Perbandingan Framework Front-End Javascript React dan Vue Pada Pengembangan Website. *Journal of Informatics and Computer Science (JINACS)*, 5(02), 157–164. <https://doi.org/10.26740/jinacs.v5n02.p157-164>
- Sulistiyanto, S., Rahmi, L., Kunio, N. I. H., Asoka, E., & Wati, A. S. (2024). Implementasi Web Development Life Cycle dalam Pembuatan Website Company Profile Ening Production. *Jurnal Penelitian Inovatif*, 4(2), 387–394. <https://doi.org/10.54082/jupin.324>
- Synko, A., & Peleshchyshyn, A. (2020). Software development documenting – documentation types and standards. *Scientific Journal of the Ternopil National Technical University*, 98(2), 120–128. https://doi.org/10.33108/visnyk_tntu2020.02.120
- Tang, L. (2019). *Building websites with Laravel and VueJS* (Issue October).
<https://www.theseus.fi/handle/10024/261248>
- TJS. (2023). *There is such thing as the perfect website*. <https://www.tjs.co.uk/web-design-news-lincolnshire/the-perfect-google-lighthouse-score-by-tjs>
- Touch, Q. (2024). *What Is QA Touch?* <https://www.qatouch.com/pricing/>
- Uriawan, W., Putra, R. D., Siregar, R. I., Gunawan, S. N., Adriansyah, S., & Nurrohman, W.

- (2024). *BrainNest: Implementation of TypeScript and MERN Stack to Improve Scalability of Interactive and Personalized E-learning*. <https://doi.org/10.20944/preprints202407.0051.v1>
- Wali, M., & Ahmad, L. (2018). Perancangan Access Open Journal System (AOJS) dengan menggunakan Framework Codeigniter dan ReactJs. *Jurnal JTik (Jurnal Teknologi Informasi Dan Komunikasi)*, 2(1), 48. <https://doi.org/10.35870/jtik.v2i1.53>
- Web.dev. (2023). *Cumulative Layout Shift (CLS)*. <https://web.dev/articles/cls>
- Wijayanto, D., & Firmansyah, F. (2022). Laravel Blade Templating Engine Implementation (Case Study: Muhammadiyah Lamongan Hospital Job Management System). *Conference SENATIK STT Adisutjipto Yogyakarta*, 7, 201–210. <https://doi.org/10.28989/senatik.v7i0.447>
- Yusran, N. (2023). *Pengimplementasian Reactjs Dan Firebase Dalam Pengembangan Platform Wahda*. <https://dspace.uui.ac.id/handle/123456789/47912>

LAMPIRAN