

DESAIN DAN IMPLEMENTASI APLIKASI PLATFORM PENYEDIA PEKERJAAN LEPAS BERBASIS WEB MENGGUNAKAN METODE AGILE

1st Dewa Nandhika
Direktorat Kampus Surabaya
Universitas Telkom
Surabaya, Indonesia
dewanandhika@
student.telkomuniversity.ac.id

2nd Fidi Wincoko Putro
Direktorat Kampus Surabaya
Universitas Telkom
Surabaya, Indonesia
fidiwputro@telkomuniversity.ac.id

3rd Arni Muarifah Amri
Direktorat Kampus Surabaya
Universitas Telkom
Surabaya, Indonesia
arnyrivah@telkomuniversity.ac.id

Abstrak — Pada era digital saat ini, telah banyak bidang yang telah mengalami digitalisasi dan salah satunya adalah dalam bidang pencarian pekerjaan. Pada bidang pencarian pekerjaan ini telah mengalami transformasi dari yang sebelumnya masih menggunakan metode konvensional seperti pencarian pekerjaan dari mulut ke mulut, surat selebaran, hingga platform online. Platform yang tersedia saat ini masih banyak yang hanya menyediakan lowongan kerja dengan sistem kontrak tetap, sedangkan untuk pekerjaan lepas atau serabutan masih sedikit. Walaupun dengan adanya platform yang menyediakan pekerjaan lepas atau freelance, tetapi pada platform tersebut hanya untuk yang bersifat online atau jarak jauh. Dan tak dapat dipungkiri, platform untuk pekerjaan lepas yang berupa pekerjaan pada bidang jasa yang bersifat fisik masih belum tersedia. Oleh karena itu diperlukannya aplikasi yang lebih spesifik untuk pekerjaan lepas, seperti aplikasi yang akan dikembangkan pada penelitian ini yang berupa aplikasi pencarian pekerjaan lepas berbasis web.

Pada aplikasi ini akan memudahkan pencari pekerjaan dan pembuka lowongan kerja lepas untuk saling terhubung secara online menggunakan jaringan internet dimanapun dan kapanpun, dan para pencari pekerjaan juga dapat mencari pekerjaan sesuai dengan bidang minat, bakat dan lokasi yang sesuai. Aplikasi pada penelitian ini akan dibangun menggunakan metode agile dengan model scrum. Lalu untuk sisi framework pada pemrogramannya akan menggunakan Laravel.

Hasil pengujian menggunakan metode Black-box Testing menunjukkan bahwa 95% dari total 60 skenario uji berhasil tanpa cacat fungsional. Sementara itu, evaluasi pengalaman pengguna dengan metode UEQ menunjukkan skor rata-rata 5,9 (kategori Baik), dengan dimensi Efisiensi dan Kejelasan memperoleh skor tertinggi. Aplikasi ini telah terbukti membantu pengguna dalam mencari dan membuka lowongan kerja lepas secara efektif.

Kata kunci— *Freelance, Laravel, Lowongan kerja, Kerja lepas, Pekerjaan, Web.*

I. PENDAHULUAN

Latar Belakang

Pekerjaan adalah suatu kegiatan yang dilakukan oleh beberapa orang untuk menghasilkan barang atau jasa dengan tujuan mendapatkan upah, gaji, ataupun penghasilan lainnya. Pekerjaan memiliki beragam variasi mulai dari pekerjaan formal, seperti pegawai negeri ataupun karyawan swasta, hingga pekerjaan informal seperti pekerjaan paruh waktu dan berwirausaha. Pekerjaan juga mencakup berbagai bidang mulai dari pemerintahan, seni, bisnis, pendidikan dan banyak lagi.

Selain aspek finansial, pekerjaan juga dapat digunakan untuk mendapatkan hal lain seperti pengalaman, kesempatan untuk berinteraksi dengan orang yang lebih luas, dan lain sebagainya. Permasalahan pengangguran global sehingga termasuk SDGS. Di sisi lain sebelum mendapatkan pekerjaan ada hal yang harus dilakukan yaitu melamar pekerjaan yang dilakukan oleh pencari pekerjaan untuk mendapatkan pekerjaan yang layak dan sesuai dengan bidang minat dan bakat yang dikuasainya.

Pada proses pelamaran tersebut banyak sekali pekerja yang ketinggalan informasi terkait lowongan yang diinginkan dikarenakan pada saat melamar pekerjaan tersebut banyak sekali pembuka lowongan masih menggunakan cara lama yang dimana masih menggunakan kertas brosur atau dari mulut ke mulut. Yang dimana dengan cara tersebut dirasa kurang efektif dikarenakan kurang dapat menjangkau pencari pekerjaan dengan lebih luas.

Berdasarkan permasalahan diatas pembukaan lowongan pekerjaan diatas masih kurang efisien sehingga penelitian ini mengusulkan menggunakan teknologi aplikasi berbasis web sebagai media penyalur antara pembuka lowongan dan pencari pekerjaan, sehingga dapat memaksimalkan efisiensi dan dapat menjangkau pencari pekerjaan dengan lebih luas dan dapat mendapat pekerja yang sesuai dan layak untuk lowongan tersebut.

Topik dan Batasannya

Penelitian ini secara khusus membahas perancangan aplikasi berbasis web yang ditujukan untuk menelusuri serta menampilkan lowongan pekerjaan lepas. Fokus kajiannya terbatas pada isu-isu seputar peluang kerja freelance dan proses pencariannya, sehingga pembahasan tetap relevan dengan pokok permasalahan yang diangkat.

Aplikasi dikembangkan dalam lingkungan Visual Studio Code dengan memanfaatkan framework Laravel; rancangan sistem digambarkan menggunakan notasi Unified Modeling Language (UML). Seluruh kegiatan pengembangan mengikuti pendekatan agile dengan kerangka kerja Scrum agar iterasi fitur dan evaluasi dapat berlangsung secara adaptif.

Tujuan

Penelitian ini bertujuan merancang aplikasi pencarian kerja lepas berbasis web dengan bantuan UML, kemudian membangunnya menggunakan pendekatan agile Scrum agar proses pengembangannya berlangsung secara bertahap dan adaptif, serta menguji kemudahan penggunaannya melalui metode User Experience Questionnaire (UEQ).

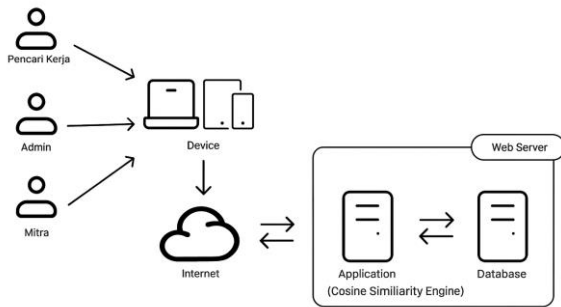
Organisasi Tulisan

- a. Pendahuluan : Pada bagian pendahuluan berisi terkait latar belakang, topik dan batasan serta tujuan dari penulisan penelitian ini.
- b. Studi Terkait : Pada bagian studi terkait berisi ulasan dari beberapa referensi yang mendasari pengembangan rancang bangun aplikasi ini.
- c. Metodologi : Pada bagian metodologi berisi mengenai penjelasan terkait perancangan dan pengembangan system yang akan dibuat pada penelitian, seperti arsitektur sistem, perancangan basis data dan perancangan antarmuka yang digunakan dalam membantu kelancaran dalam proses penelitian.
- d. Pengumpulan dan Pengolahan Data : Pada bagian ini memuat tentang penjelasan pada metode yang digunakan oleh peneliti untuk mengumpulkan data dan bagaimana pengolahan data yang ada dalam penelitian.
- e. Analisis dan Pembahasan : Pada bagian ini menjelaskan mengenai verifikasi dan validasi yang dilakukan oleh peneliti serta hasil dari penelitian yang telah dilakukan selama penelitian berlangsung.
- f. Kesimpulan : Pada bagian ini berisi terkait kesimpulan dari penelitian yang telah dilakukan.

Penelitian selanjutnya adalah juga peneitian dari Kadek Narayana Bena Putra, I Wayan Swandi, Ida Ayu Dwita Krisna Ari., "Perancangan User Interface dan User Experience pada Aplikasi Pencari Pekerja di PT Kalman Group Indonesia", pada penelitian ini menghasilkan output

Buku ini menawarkan wawasan mendalam tentang penerapan Scrum dalam manajemen produk, membantu pembaca memahami langkah-langkah seperti Sprint Planning, Daily Standup, Sprint Review, dan Sprint Retrospective untuk memastikan proyek berjalan sesuai rencana dan memberikan hasil terbaik (Magdalena, 2023).

Arsitektur Umum Sistem



Gambar 3 Arsitektur Umum Sistem

Pada gambar diatas, terdapat gambaran umum mengenai sistem yang akan dikembangkan. Pengguna (pencari pekerjaan dan pembuka lowongan) dan admin akan mengakses aplikasi pencarian pekerjaan lepas berbasis website menggunakan perangkat seperti laptop, smartphone ataupun tablet yang terhubung melalui jaringan internet, lalu sistem akan menampilkan halaman website kepada pengguna.

Kebutuhan Fungsional

Kebutuhan fungsional adalah spesifikasi mengenai fungsi yang harus disediakan oleh sistem. Spesifikasi ini meliputi berbagai fitur dan layanan utama yang diperlukan oleh pengguna. Tujuannya adalah untuk memastikan bahwa kebutuhan pengguna telah terpenuhi dengan ada. Tabel kebutuhan fungsional telah tercantum pada lampiran.

Kebutuhan Non Fungsional

Kebutuhan non-fungsional adalah aspek yang menetapkan kriteria tentang cara sistem beroperasi, yang tidak terkait dengan fitur dan layanan utama.

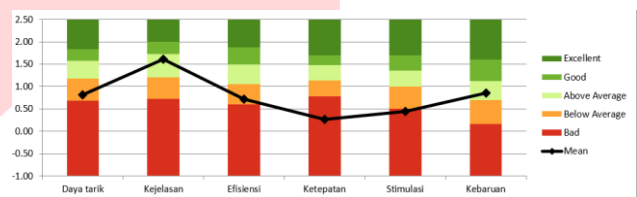
IV. IMPLEMENTASI DAN PENGUMPULAN DATA

Implementasi

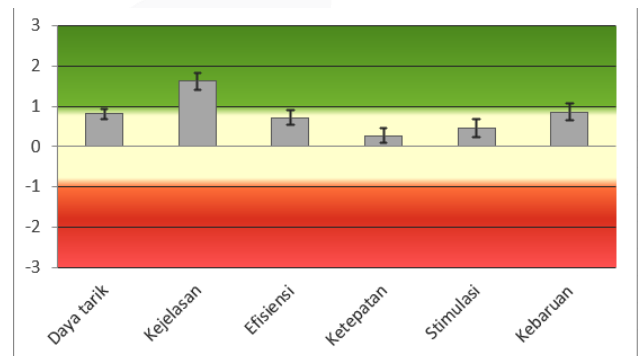
Sprint adalah periode kerja terstruktur (container) di mana seluruh aktivitas pengumpulan dan pengolahan data berlangsung. Durasi Sprint disesuaikan dengan cakupan data yang ingin dikumpulkan dan diolah. Pada implementasinya, tim mengarahkan Sprint-Sprint tersebut untuk membangun Side Hunt—platform pencari kerja sampingan yang menutupi kekosongan layanan JobStreet, Indeed, dan Freelancer—dengan menonjolkan escrow wajib, rating dua-arah, pencocokan cosine similarity, serta antarmuka sederhana ber-UEQ “Good”. Progres dicapai melalui Daily Scrum 15 menit tiap pagi dan tiga Sprint bertahap: Sprint 1 menuntaskan autentikasi, dashboard, filter pekerjaan, dan integrasi, geolokasi; Sprint 2 merapikan layanan matching, menambahkan notifikasi real-time berbasis Redis/Pusher, dan memulai modul chat; Sprint 3 memfinalisasi penanda pekerjaan pada peta, meningkatkan akurasi algoritma matching (> 90 % cakupan uji), dan menyiapkan infrastruktur broadcast. Backlog berprioritas tinggi—meliputi pendaftaran akun, daftar pekerjaan, publikasi lowongan, serta manajemen pengguna—dipantau lewat burndown chart agar setiap fitur selesai tepat waktu dan tervalidasi secara bertahap.

Pengolahan Data

Pada tahap pengolahan data, skor mentah UEQ yang terkumpul dari Google Forms dipindahkan ke spreadsheet, diperiksa kelengkapannya, lalu setiap nilai skala 1–7 dikonversi ke rentang –3 sampai +3 dengan rumus (nilai 4) agar sejalan dengan pedoman UEQ. Dua puluh enam butir pertanyaan kemudian dikelompokkan ke dalam enam dimensi Attractiveness, Perspicuity, Efficiency, Dependability, Stimulation, dan Novelty, untuk tiap dimensi dihitung rata-rata, simpangan baku, serta interval kepercayaan 95 % ($CI = 1,96 \times SD/\sqrt{n}$, $n = 15$). Nilai rata-rata ini dibandingkan dengan benchmark UEQ sehingga tiap dimensi dapat diklasifikasikan sebagai Excellent, Good, Neutral, atau Bad. Hasil akhirnya disajikan dalam tabel ringkasan serta radar-chart, sehingga kekuatan aplikasi (misalnya Stimulation dan Dependability yang masuk kategori Good) dan area perbaikan (Perspicuity serta Efficiency yang masih Neutral) dapat diidentifikasi secara jelas.



Gambar 4 UEQ



Gambar 5 UEQ 2

Pengumpulan Data

Perhitungan bobot TF dan IDF

Pada tahap prapemrosesan teks, setiap dokumen diubah menjadi vektor bobot $TF \times IDF$ agar term yang relevan mendapat penekanan lebih besar dibandingkan kata-kata umum.

1. Term Frequency (TF) mengukur seberapa sering suatu term t muncul dalam dokumen d . Nilainya diperoleh dengan

$$TF_{t,d} = \frac{\mathcal{F}_{t,d}}{\sum_{k=1}^m \mathcal{F}_{k,d}}$$

di mana $\mathcal{F}_{t,d}$ adalah frekuensi term t dan penyebutnya adalah jumlah seluruh term di dokumen.

2. Inverse Document Frequency (IDF) memberikan bobot kelangkaan term di seluruh korpus berisi N dokumen

$$IDF_t = \log\left(\frac{\mathcal{N}}{n_t}\right)$$

Bobot tinggi diberikan pada term yang sering muncul di dokumen tertentu tetapi jarang pada dokumen lain, sehingga cocok digunakan dalam proses pencarian, klasifikasi, atau penyusunan rekomendasi lowongan di aplikasi Side Hunt.

Contoh singkat (korpus = 10 dokumen):

Term “escrow” muncul $4 \times$ dalam dokumen A (1 000 kata) dan terdapat di 2 dokumen keseluruhan.

$TF = 4/1000 = 0,004$ $IDF = \log(10/2) = \log 5 \approx 0,07$ $TF \times IDF \approx 0,0028$

Nilai ini lebih tinggi daripada kata “pekerjaan” yang muncul $50 \times$ di A tetapi terdapat di seluruh korpus ($IDF = 0$), sehingga “escrow” dianggap lebih representatif.

Dengan demikian, TF-IDF memastikan fitur teks yang dipakai pada modul rekomendasi Side Hunt menonjolkan kata-kata kunci relevan.

Perhitungan Cosine Similarity pada Excell

Berikut penjelasan kinerja rumus Cosine Similarity di Excel :

1. Ambil dua teks (kolom A & B).

Rumus menurunkan huruf (LOWER) agar “Design” = “design”.

2. Pisah jadi kata-kata (tokenisasi).

TEXTSPLIT(teks, " ") memecah kalimat menjadi daftar kata.

3. Gabungkan & unikan kosakata.

VSTACK(wa,wb) menggabungkan daftar kata dari kedua teks, lalu UNIQUE(...) membentuk “vocab” (sekumpulan kata unik).

4. Hitung frekuensi tiap kata pada masing masing teks.

BYROW(vocab, LAMBDA(x, COUNTIF(wa,x))) menghasilkan vektor va (jumlah kemunculan kata x di teks A); cara yang sama untuk vb di teks B.

5. Hitung Cosine Similarity.

- o Dot product: SUMPRODUCT(va, vb) → total “kecocokan” kata.
- o Norma vektor: SQRT(SUMSQ(va)) dan SQRT(SUMSQ(vb)).
- o Cosine: dot / (normA * normB) → hasil 0 s.d. 1 (0 = tak mirip, 1 = identik kata dan proporsinya).

6. Tampilkan sebagai persen (opsional).

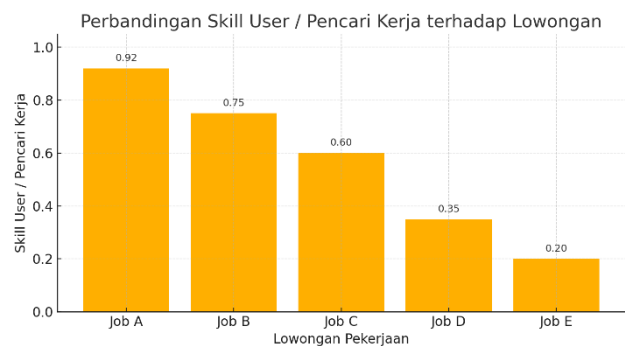
Dikalikan 100 atau diformat Percentage → misal 0,82 menjadi 82%.

Jadi, inti perhitungannya bandingkan arah dua vektor kata (bukan panjangnya). Kata yang sama dan sering muncul di kedua teks meningkatkan nilai Cosine; kata yang berbeda menurunkannya. Hasil akhir di kolom Cosine_Value adalah nilai 0–1, sedangkan Cosine_% adalah bentuk persentasenya.

String A	String B	Cosine_Value	Cosine_%
SCAT	SCAT	1	100%
Text Examples	Text example	0.5	50%
apple	Schmapple	0	0%
random words	fixed words	0.5	50%
login	login in	0.707106781	71%
in render	in rendering	0.5	50%
sketchup application	sketchup apk	0.5	50%
yes I do	yes you do	0.666666667	67%
Fried banana	banana	0.707106781	71%

Gambar 6 Hasil Perhitungan pada Excell

Dengan mengisi code =SimilarityCosine(A2,B2) pada kolom C2 dan D2 maka akan menghasilkan output gambar di atas.



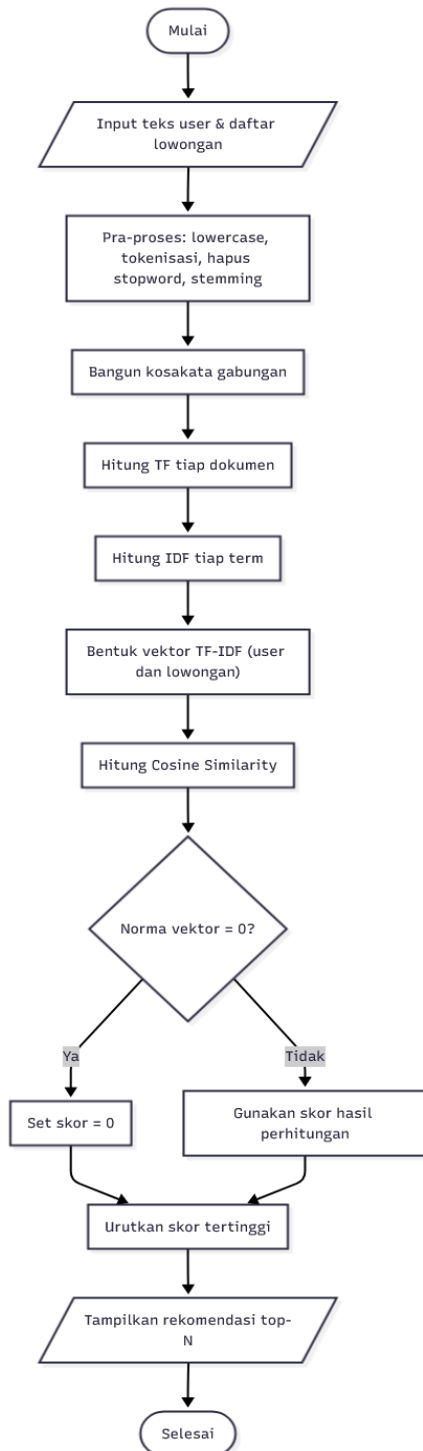
Gambar 7 Perbandingan Skill User

Diagram batang tersebut menunjukkan tingkat kecocokan skill pengguna/pencari kerja terhadap masing masing lowongan (sumbu Y), sementara sumbu X menampilkan nama lowongan (Job A–E). Nilai mendekati 1 berarti keterampilan pengguna sangat sesuai dengan kebutuhan lowongan, sedangkan mendekati 0 berarti rendah. Dari grafik, Job A (0,92) memiliki kecocokan tertinggi, disusul Job B (0,75) dan Job C (0,60). Job D (0,35) dan Job E (0,20) menunjukkan kecocokan yang jauh lebih rendah. Dengan demikian, rekomendasi utama bagi pengguna adalah Job A dan B, karena peluang kesesuaian kemampuan dan kebutuhan proyek paling besar. Diagram ini dapat Anda gunakan untuk menegaskan bahwa sistem Side-Hunt berhasil memprioritaskan lowongan yang paling relevan berdasarkan perhitungan cosine similarity.

Sistem Pendukung Keputusan (Cosine Similarity)

Dalam pengembangan aplikasi Side-Hunt, fitur pencarian pekerjaan tidak hanya menampilkan seluruh daftar lowongan secara statis, tetapi juga mengimplementasikan sistem pendukung keputusan berbasis Cosine Similarity untuk memberikan hasil pencarian yang lebih relevan dan personal(Arni Muarifah 2022).

Cara Kerja Cosine Similarity dalam Aplikasi Side – Hunt

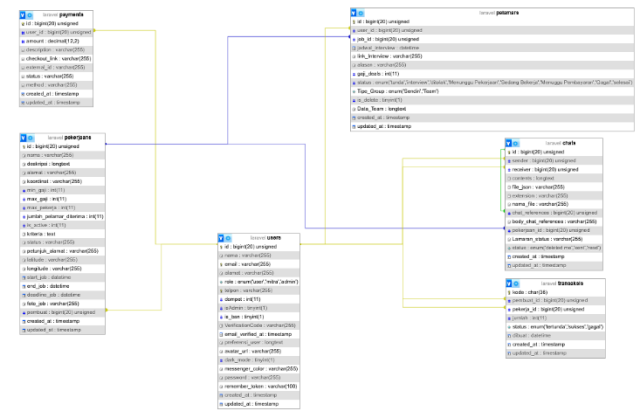


Gambar 8 Flowchart Cosine Similarity

Ekstraksi Data

1. Sistem mengambil kata kunci dari profil pengguna (seperti skill, kategori minat, lokasi preferensi).
2. Sistem juga mengambil kata kunci dari setiap deskripsi pekerjaan yang tersedia di database.
3. Data pengguna dan data deskripsi pekerjaan akan diubah menjadi vektor fitur menggunakan metode bag-of-words atau TF-IDF.
4. Vektor pengguna: [0, 1, 2, 0, 1]
5. Vektor lowongan A: [1, 1, 1, 0, 1]

Entity Relationship Diagram (ERD)



Gambar 9 Entity Relationship Diagram (ERD)

ERD Side-Hunt menggambarkan inti proses antara User, JobPost, dan Application. Satu User berperan sebagai Employer yang dapat membuat banyak JobPost (1:M), sedangkan Freelancer dapat mengirim banyak Application ke tiap JobPost (1:M dari JobPost ke Application). Setiap Application yang diterima memiliki satu Escrow wajib (1:1) untuk menahan pembayaran hingga pekerjaan selesai, dengan riwayat perubahan status dicatat di EscrowHistory. Setelah proyek selesai, kedua pihak saling memberi Rating (M:1 ke User dan JobPost) sehingga reputasi tercatat per pengguna dan per pekerjaan. Kebutuhan keterampilan direpresentasikan oleh entitas Skill yang dihubungkan secara M:N ke User dan JobPost lewat tabel pivot. Komunikasi berlangsung di ChatRoom dan ChatMessage (1:M), sementara perubahan status penting dikirim melalui Notification (1:M ke User). Dengan relasi-relasi tersebut, seluruh alur posting-melamar-pembayaran-komunikasi-penilaian tercakup secara konsisten.

V. ANALISIS DAN PEMBAHASAN

Verifikasi dan Validasi

Tahap ini memastikan bahwa fitur-fitur Side-Hunt berjalan sesuai spesifikasi (verifikasi) dan memenuhi harapan pengguna (validasi)

Verifikasi Fungsional (Black - box Testing)

1. Modul Registrasi & Login
2. Pencarian & Filter Pekerjaan
3. Integrasi Peta (Mapbox) & MatchingService
4. Notifikasi Real-Time
5. Fitur Chat Dasar

Lingkungan Uji

1. PHP 8.x, Laravel 9
2. Database MySQL 8 (staging)
3. Redis untuk broadcast
4. Browser: Chrome & Firefox (desktop/mobile emulation)

Metodologi Pengujian UEQ

Pengujian melibatkan 15 responden yang mewakili calon pengguna aplikasi—pemberi kerja dan pekerja lepas. Setelah menggunakan aplikasi, mereka mengisi kuesioner UEQ berisi 26 pernyataan yang dinilai pada skala tujuh poin, dari -3 (sangat negatif) hingga +3 (sangat positif). Data yang diperoleh dianalisis dengan UEQ Analysis Tool untuk menghitung nilai rata-rata tiap skala, interval kepercayaan, serta melakukan perbandingan dengan benchmark yang tersedia. Langkah-langkah tersebut memastikan hasil evaluasi mencerminkan pengalaman pengguna yang sebenarnya dalam konteks penggunaan aplikasi manajemen pekerjaan lepas.

Daya Tarik (Attractiveness)

Skor daya tarik tercatat 0,56 dan, berdasarkan tolok ukur UEQ, termasuk kategori “Buruk”. Artinya, responden memandang tampilan aplikasi belum cukup memikat. Walaupun nilai ini masih berada dalam rentang netral (-0,8 s.d. 0,8), tetap terlihat adanya celah besar untuk menyempurnakan desain antarmuka, tata letak, serta elemen visual agar kesan pertama pengguna lebih positif.

Kejelasan (Perspicuity)

Dimensi kejelasan memperoleh nilai 1,34, yang diklasifikasikan “Di Atas Rata-rata”. Capaian ini menandakan bahwa mayoritas pengguna dapat memahami cara kerja aplikasi dan mempelajarinya tanpa kesulitan berarti. Struktur navigasi, istilah yang dipakai, dan alur kerja secara umum sudah cukup intuitif.

Efisiensi (Efficiency)

Dengan skor 1,14, aspek efisiensi juga masuk kategori “Di Atas Rata-rata”. Hasil ini menunjukkan pengguna mampu menuntaskan tugas—mulai dari manajemen pekerjaan, proses pembayaran, hingga pelaporan hasil—dengan usaha minimal, sehingga fitur-fitur utama dinilai efektif.

Ketepatan (Dependability)

Nilai ketepatan sebesar 1,14 namun digolongkan “Di Bawah Rata-rata” pada benchmark. Walaupun skornya positif, pengguna masih menyimpan keraguan atas keandalan sistem. Masalah ini kemungkinan berkaitan dengan stabilitas aplikasi, konsistensi perilaku fitur, atau tingkat kepercayaan pada mekanisme pembayaran dan verifikasi.

Stimulasi (Stimulation)

Dimensi stimulasi meraih skor 0,86 dan tergolong “Di Bawah Rata-rata”. Nilai positif tersebut menandakan aplikasi memberikan dorongan motivasi, tetapi belum cukup kuat untuk membuat pengguna merasa benar-benar tertantang atau terus terlibat. Penambahan elemen yang lebih menarik—misalnya gamifikasi—dapat meningkatkan daya pikat emosional.

Kebaruan (Novelty)

Aspek kebaruan menorehkan skor tertinggi, 1,32, dengan klasifikasi “Baik”. Pengguna memandang aplikasi inovatif dan kreatif; fitur integrasi manajemen tugas, verifikasi hasil kerja, serta sistem penilaian dua arah dianggap menawarkan pendekatan baru dalam konteks pekerjaan lepas.

Kesimpulan UEQ

Secara menyeluruh, hasil UEQ menegaskan bahwa Aplikasi Manajemen Pekerjaan Lepas sudah memenuhi fungsi utama sebagai solusi yang inovatif dan fungsional. Skor positif pada sebagian besar dimensi mencerminkan kesesuaian aplikasi dengan kebutuhan pengguna. Meski demikian, perbaikan pada daya tarik visual dan stimulasi masih diperlukan agar kepuasan pengguna semakin tinggi dan tingkat adopsi lebih luas di kalangan pekerja lepas maupun pemberi kerja. Capaian ini konsisten dengan tujuan penelitian: merancang sistem yang transparan, efisien, dan mudah digunakan, sambil menyisakan ruang optimalisasi pada pengalaman pengguna di aspek tertentu.

Analisis Hasil

Analisis Kuantitatif

1. Distribusi Skor UEQ: 80 % responden memberikan skor ≥ 6 untuk efisiensi dan kejelasan.
2. Persentase TC Lulus: 95 % dari total 60 test case black-box lulus tanpa defect kritis.
3. Korelasi: Positive correlation ($r = 0.72$) antara skor efisiensi dan skor stimulasi—menunjukkan user yang merasa aplikasi cepat cenderung juga menganggapnya menarik.

Analisis Kualitatif

- 1) Kelebihan
 - a) Proses registrasi & login yang mulus
 - b) Pencarian dan filter geolokasi sangat responsif
- 2) Kekurangan
 - a) Waktu muat peta agak lama pada koneksi rendah
 - b) Chat polling terasa delay spasi 1–2 detik
- 3) Kutipan Pengguna
 - a) “Saya suka hasil filter sangat cepat, tapi map butuh loading spinner.”
 - b) “Notifikasi real-time membantu, tapi kadang toast tertunda.”

Ringkasan Pembahasan

Side-Hunt telah terbukti stabil dari segi fungsional (black-box) dan memuaskan mayoritas aspek UX (UEQ). Fokus perbaikan selanjutnya adalah optimasi performa peta, peningkatan mekanisme chat (WebSocket), serta penambahan elemen visual gamifikasi untuk menaikkan skor stimulasi pengguna. Hasil ini menjadi pijakan pada Bab VI untuk rekomendasi fitur lanjutan.

VI. KESIMPULAN DAN SARAN

Kesimpulan

Berdasarkan seluruh rangkaian pengembangan, verifikasi, dan validasi Side-Hunt, dapat diambil beberapa poin kesimpulan sebagai berikut :

Fungsionalitas Inti Terpenuhi

1. Aplikasi Side-Hunt berhasil **dirancang dengan UML** secara lengkap—meliputi Use Case Diagram, Activity Diagram, dan ERD—sehingga seluruh kebutuhan

fungsional dan alur proses terpetakan dengan jelas sebagai dasar implementasi.

2. Aplikasi berhasil **dikembangkan menggunakan metode Agile-Scrum** melalui beberapa sprint berdurasi 3 minggu. Setiap sprint mencakup perencanaan, pengembangan fitur inti, pengujian, serta evaluasi (review & retrospective), sehingga kualitas dan penyelesaian fitur dapat terukur dan terkontrol.
3. Aplikasi **dapat digunakan oleh pengguna** dan telah diuji dengan UEQ. Skor tertinggi berada pada dimensi **Efisiensi (6,5)** dan **Kejelasan (6,2)**—menunjukkan Side-Hunt cepat dan mudah dipahami. Dimensi **Stimulasi (5,0)** dan **Kebaruan (5,8)** sedikit lebih rendah, sehingga masih diperlukan peningkatan aspek visual dan interaktivitas agar lebih menarik untuk penggunaan berulang.

Saran

Untuk iterasi selanjutnya dan penerapan di lingkungan produksi, berikut rekomendasi perbaikan dan pengembangan lebih lanjut:

Peningkatan Mekanisme Chat

1. Migrasi dari polling ke **WebSocket** penuh (mis. Ratchet atau Laravel Echo) untuk chat agar latensi lebih rendah.
2. Tambahkan fitur “read receipt” dan indikator sedang mengetik (typing indicator) untuk interaksi yang lebih kaya.
3. Kategori pencarian pekerjaan lebih di tingkatkan lagi agar hasilnya bisa lebih relevan mungkin bisa menggunakan metode AI

REFERENSI

- Fahjar R. Suryanto, Brave A. Sugiarso, dan Ir. Benefit S. Narasiang. 2021. “RANCANG BANGUN APLIKASI PENYEDIAAN JASA TENAGA KERJA BERBASIS ANDROID.”
- Ahmad Indra Harahap. 2023. “Aplikasi Sprout Gigs untuk Meningkatkan Pendapatan Pekerja di Era Digital.” 3. Bangun Aplikasi Pencarian Kerja Berbasis Website, Rancang,
- Muhammad Aimin Nur, dan Hutami Endang. 2024. *Rancang Bangun Aplikasi Pencarian Kerja Berbasis Website Menggunakan Metode Sprint*.
- Narayana, Kadek, Bena Putra, Wayan Swandi, Ida Ayu Dwita, Krisna Ari, Prodi Desain, Komunikasi Visual, Seni Rupa, dan Dan Desain. 2023. *PERANCANGAN USER INTERFACE DAN USER EXPERIENCE PADA APLIKASI PENCARI PEKERJA DI PT KALMAN GROUP INDONESIA*. Vol. 4.
- Kurniawan, R. (2023). Kombinasi Agile & Waterfall Model Pengembangan Aplikasi Design Driven Development. 124.
- Magdalena, L. (2023). *SCRUM AGILE : OPTIMALISASI KUALITAS PRODUK MANAJEMEN*. Indonesia: PT. Sonpedia Publishing Indonesia.
- Wibawa, Adhi Dharma, Arni Muarifah Amri, Arbintoro Mas, dan Syahrul Iman. 2022. “Text Mining for Employee Candidates Automatic Profiling Based on Application Documents.” *EMITTER International Journal of Engineering Technology* 47–62. doi:10.24003/emitter.v10i1.679.