

IMPLEMENTASI ALGORITMA GREEDY MATCHING PADA SISTEM REKOMENDASI JASA TUKANG BANGUNAN

Rosidin^{1*)}, Misbahus Sudur²⁾, Raditya Danar Dana³⁾, Ratna Dewi Lestiyorini⁴⁾, Sukarsa⁵⁾

^{1,3,4}STMIK IKMI Cirebon

^{2,5}Fakultas Ilmu Komputer, Universitas Nahdlatul Ulama Cirebon

email : ¹rosidin.crb@gmail.com*, ²misbahussudur2103@gmail.com, ³radith_danar@yahoo.com,

⁴ratna.dewilestiyorini@gmail.com, ⁵sukarsa@unucirebon.ac.id

*Corresponding Author

Abstract

The small-scale construction industry in Warnasari District, Brebes Regency, still faces inefficiencies in finding daily construction workers because the recruitment process is commonly carried out through gathering points and word-of-mouth information. This condition creates income uncertainty for workers and makes it difficult for employers to assess workers' skills, location, and reputation. This study aims to develop a web-based recommendation system for construction worker services by implementing the Greedy Matching algorithm. The main contribution of this research is the application of weighted matching based on skill, location distance, and performance rating criteria to generate fast and measurable worker recommendations. The system was developed using the Rapid Application Development (RAD) method, while the recommendation process was carried out by calculating the heuristic value of each candidate and selecting the candidate with the highest score as a local optimal solution. The evaluation was conducted through Black Box testing to assess the validity of system functions and a Greedy Matching calculation simulation to evaluate the candidate ranking output. The test results showed that all system features functioned validly with a success rate of 100%. The calculation simulation results showed that the candidate with a score of 97.0 was placed in the first position compared to another candidate with a score of 74.0. These results indicate that the system is able to generate measurable construction worker service recommendations based on the applied weighted criteria.

Keywords: Greedy Matching, Recommendation System, Construction Worker Services, Web.

Abstrak

Industri konstruksi skala kecil di Kecamatan Warnasari, Kabupaten Brebes, masih menghadapi inefisiensi dalam pencarian tenaga kerja harian karena proses rekrutmen umumnya dilakukan melalui titik kumpul dan informasi dari mulut ke mulut. Kondisi ini menyebabkan ketidakpastian pendapatan bagi tukang serta menyulitkan pemberi kerja dalam menilai keahlian, lokasi, dan reputasi pekerja. Penelitian ini bertujuan membangun sistem rekomendasi jasa tukang bangunan berbasis web dengan menerapkan algoritma Greedy Matching. Kontribusi utama penelitian ini adalah penerapan pencocokan berbobot berdasarkan kriteria keahlian, jarak lokasi, dan rating kinerja untuk menghasilkan rekomendasi tukang secara cepat dan terukur. Sistem dikembangkan menggunakan metode Rapid Application Development (RAD), sedangkan proses rekomendasi dilakukan dengan menghitung nilai heuristik setiap kandidat dan memilih kandidat dengan skor tertinggi sebagai solusi optimal lokal. Evaluasi dilakukan melalui pengujian Black Box untuk menilai validitas fungsi sistem dan simulasi perhitungan Greedy Matching untuk menilai keluaran pemeringkatan kandidat. Hasil pengujian menunjukkan seluruh fitur sistem berjalan valid dengan tingkat keberhasilan 100%. Hasil simulasi perhitungan menunjukkan bahwa kandidat dengan skor 97,0 ditempatkan pada urutan pertama dibandingkan kandidat lain dengan skor 74,0. Hasil tersebut menunjukkan bahwa sistem mampu menghasilkan rekomendasi jasa tukang secara terukur berdasarkan bobot kriteria yang digunakan.

Keywords: Greedy Matching, Sistem Rekomendasi, Jasa Tukang Bangunan, Web

PENDAHULUAN

Perkembangan teknologi informasi telah mendorong transformasi layanan pada berbagai sektor, termasuk sektor jasa pertukangan dan konstruksi. Pada sektor konstruksi skala kecil, kebutuhan terhadap jasa tukang bangunan terus meningkat seiring bertambahnya pembangunan perumahan, renovasi bangunan, dan infrastruktur lokal. Namun, proses pencarian tukang bangunan di Kecamatan Warnasari, Kabupaten Brebes, masih banyak dilakukan secara konvensional melalui titik kumpul, relasi personal, atau informasi dari mulut ke mulut. Pola tersebut menimbulkan inefisiensi pasar (*market inefficiency*) karena informasi antara pemberi kerja dan penyedia jasa tidak tersebar secara merata. Pemberi kerja sulit menilai keahlian, jarak lokasi, dan reputasi tukang secara objektif, sedangkan tukang bangunan menghadapi ketidakpastian peluang kerja akibat ketergantungan pada jaringan informal.

Sejumlah penelitian sebelumnya telah menerapkan berbagai metode untuk mendukung proses pencarian, rekomendasi, dan pemilihan tenaga kerja. Sistem rekomendasi pencarian kerja telah dikembangkan menggunakan metode *Simple Additive Weighting* (SAW), sementara pendekatan *freelance matching* telah diterapkan menggunakan analisis data dan *machine learning* (Ali Fikri et al., 2025; Sahnoun & Elhadjamor, 2024). Model pendukung keputusan juga telah diterapkan untuk pemilihan kontraktor, sedangkan pendekatan *Greedy Ensemble Selection* digunakan untuk menghasilkan rekomendasi *Top-N* (Mergawy et al., 2023; Vente et al., 2022). Penelitian tersebut menunjukkan penggunaan metode pembobotan, *machine learning*, dan algoritma *greedy* dalam proses pencocokan atau pemilihan alternatif. Namun, penelitian ini secara khusus menerapkan *Greedy Matching* berbobot dengan menggabungkan kriteria keahlian, jarak lokasi, dan rating kinerja untuk menghasilkan pemeringkatan jasa tukang bangunan secara cepat dan terukur.

Berdasarkan penelitian terdahulu tersebut, masih terdapat celah penelitian (*research gap*) pada mekanisme pemeringkatan tukang yang mempertimbangkan beberapa kriteria secara simultan, yaitu keahlian, jarak lokasi, dan rating kinerja. Penelitian sebelumnya umumnya telah menyediakan fasilitas pencarian, pemesanan, atau rekomendasi, tetapi belum menekankan proses pencocokan berbobot yang dapat menjelaskan alasan seorang tukang ditempatkan pada urutan rekomendasi teratas (Komen et al., 2021). Dengan demikian, permasalahan yang perlu dijawab dalam penelitian ini bukan hanya bagaimana membangun aplikasi pencarian tukang, melainkan bagaimana sistem dapat menghasilkan rekomendasi tukang secara cepat, terukur, dan transparan berdasarkan

kombinasi kriteria yang relevan dengan kebutuhan pengguna. Pengembangan sistem berbasis web juga dapat mendukung penyediaan layanan secara lebih terstruktur (Titasari Br Bangun et al., 2026).

Beberapa metode dapat digunakan untuk menyelesaikan masalah pemeringkatan atau pengambilan keputusan multikriteria, seperti AHP, TOPSIS, dan Machine Learning. AHP memiliki keunggulan dalam menentukan bobot kriteria melalui perbandingan berpasangan, sedangkan TOPSIS dapat digunakan untuk menentukan alternatif terbaik berdasarkan jarak terhadap solusi ideal. Namun, kedua metode tersebut umumnya membutuhkan proses pembobotan, normalisasi, dan penilaian alternatif yang lebih kompleks. Machine Learning juga dapat digunakan untuk sistem rekomendasi karena mampu mengenali pola dari data historis dan menghasilkan prediksi yang lebih adaptif (Hatmara Sukma et al., 2026). Akan tetapi, pendekatan ini membutuhkan data latih yang memadai, data historis pengguna, serta evaluasi menggunakan metrik seperti akurasi, presisi, *recall*, atau *F1-score*. Pada konteks pencarian tukang bangunan di wilayah lokal, ketersediaan data historis dan data berlabel masih terbatas, sehingga penggunaan Machine Learning belum menjadi pilihan utama pada tahap pengembangan awal sistem.

Algoritma Greedy Matching dipilih karena memiliki keunggulan dalam menghasilkan rekomendasi secara cepat dengan proses perhitungan yang sederhana dan efisien (Saktia Purnama et al., 2024). Pada penelitian ini, sistem hanya memerlukan proses pemeringkatan kandidat tukang berdasarkan beberapa kriteria yang telah ditentukan, yaitu keahlian, jarak lokasi, dan rating kinerja (Rahmawan et al., 2022). Pendekatan *Greedy* telah diterapkan pada berbagai permasalahan pemilihan alternatif dan optimasi (Angul et al., 2025; Fulber-Garcia et al., 2022; Gracia Sembiring et al., 2025). Dibandingkan dengan metode AHP yang membutuhkan proses perbandingan berpasangan antar kriteria dan alternatif, serta TOPSIS yang memerlukan tahapan normalisasi dan perhitungan kedekatan terhadap solusi ideal, Greedy Matching mampu menghasilkan keputusan rekomendasi dengan tahapan yang lebih ringkas dan waktu komputasi yang lebih cepat (Sunanto & Abadi Faural, 2021). Selain itu, dibandingkan pendekatan Machine Learning yang memerlukan data historis dalam jumlah besar untuk proses pelatihan model, Greedy Matching dapat langsung diterapkan tanpa proses training sehingga lebih sesuai untuk lingkungan sistem yang memiliki jumlah data terbatas dan kebutuhan rekomendasi secara real-time.

Kebaruan (*novelty*) penelitian ini terletak pada penerapan mekanisme Greedy Matching berbobot

dalam sistem rekomendasi jasa tukang bangunan berbasis web untuk konteks pasar tenaga kerja konstruksi lokal. Berbeda dari penelitian sebelumnya yang menekankan pencarian, pemesanan, lokasi, atau kecocokan layanan, penelitian ini menggabungkan keahlian, jarak lokasi, dan rating kinerja ke dalam satu skor rekomendasi. Penelitian ini bertujuan membangun sistem rekomendasi jasa tukang bangunan berbasis web serta mengevaluasi algoritma Greedy Matching dalam menghasilkan pemeringkatan kandidat tukang yang relevan secara cepat dan terukur di Kecamatan Warnasari, Kabupaten Brebes.

METODOLOGI PENELITIAN

Penelitian ini menggunakan metode *Rapid Application Development (RAD)* dalam proses pengembangan sistem. *RAD* dipilih karena sesuai untuk pengembangan aplikasi layanan jasa yang membutuhkan proses pembuatan purwarupa secara cepat, pengujian bertahap, dan penyesuaian terhadap kebutuhan pengguna. Tahapan *RAD* yang digunakan dalam penelitian ini meliputi *Requirements Planning*, *User Design*, *Construction*, dan *Cutover*.

Pada tahap *Requirements Planning*, dilakukan identifikasi kebutuhan sistem berdasarkan permasalahan pencarian tukang bangunan di Kecamatan Warnasari. Kebutuhan utama sistem meliputi pengelolaan data pengguna, data pekerja, kategori keahlian, proyek, penugasan pekerja, pembayaran, dan ulasan pekerja. Selain itu, ditentukan tiga kriteria utama yang digunakan dalam proses rekomendasi, yaitu kesesuaian keahlian, kedekatan lokasi, dan rating kinerja.

Pada tahap *User Design*, dilakukan perancangan alur sistem, basis data, antarmuka pengguna, serta proses rekomendasi tukang. Perancangan ini menghasilkan alur kerja sistem yang dimulai dari pengguna memasukkan kebutuhan pekerjaan, sistem mengambil data tukang tersedia, sistem menghitung skor kecocokan, kemudian menampilkan daftar rekomendasi tukang berdasarkan skor tertinggi yang telah diurutkan sesuai nilai setiap kandidat.

Pada tahap *Construction*, sistem dikembangkan menggunakan PHP Native dengan pendekatan OOP dan MVC. Basis data menggunakan MySQL/MariaDB serta antarmuka menggunakan HTML, CSS, JavaScript, dan Bootstrap. Pada tahap *Cutover*, dilakukan pengujian fungsi sistem menggunakan Black Box Testing dan simulasi Greedy Matching untuk mengevaluasi pemeringkatan tukang. Pengujian dilakukan pada seluruh fitur utama aplikasi, meliputi proses registrasi, login, pencarian tukang, pengelolaan proyek, penugasan pekerja, pembayaran, serta pemberian ulasan untuk memastikan sistem berjalan sesuai kebutuhan pengguna.

Aplikasi dikembangkan menggunakan arsitektur Model View Controller (MVC). MVC

digunakan untuk memisahkan logika pengolahan data, logika pengendali sistem, dan tampilan antarmuka pengguna. Pemisahan ini membuat struktur aplikasi lebih terorganisasi dan memudahkan proses pemeliharaan sistem serta pengembangan berkelanjutan.

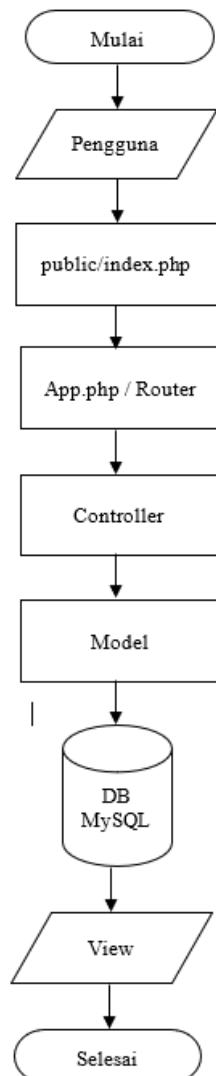
Arsitektur sistem terdiri atas tiga komponen utama. Model berfungsi mengelola akses data ke basis data, seperti data pekerja, proyek, dan pengguna. View berfungsi menampilkan antarmuka sistem kepada pengguna, seperti halaman utama, halaman login, halaman admin, daftar pekerja, dan daftar proyek. Controller berfungsi menerima permintaan dari pengguna, memanggil model yang dibutuhkan, memproses logika sistem, kemudian mengirimkan data ke view secara terstruktur dan efisien.

Tabel 1. Struktur MVC pada aplikasi

Komponen MVC	File/Folder	Fungsi
Front Controller	public/index.php	Titik masuk utama aplikasi dan pemanggil router.
Core Router	app/Core/App.php	Membaca URL, menentukan controller, method, dan parameter.
Core Controller	app/Core/Controller.php	Memuat model dan view yang diperlukan.
Core Model	app/Core/Model.php	Kelas induk untuk model.
Database	app/Core/Database.php	Mengelola koneksi dan query database menggunakan PDO.
Controller	app/Controllers	Mengatur alur proses aplikasi.
Model	app/Models	Mengelola operasi data pada database.
View	app/Views	Menampilkan halaman antarmuka kepada pengguna.

Aplikasi menerapkan konsep Object Oriented Programming (OOP) melalui penggunaan kelas seperti App, Controller, Database, Model, Pekerja, Proyek, User, HomeController, AuthController, dan AdminController. Konsep OOP digunakan untuk membungkus data dan fungsi ke dalam kelas sehingga pengelolaan kode menjadi lebih modular, terstruktur, dan mudah dikembangkan. Buku Algoritma dan Pemrograman menjelaskan bahwa OOP membungkus data dan instruksi ke dalam kelas, sedangkan instansiasi kelas saat program dijalankan disebut objek.

Penerapan OOP pada aplikasi ini mendukung prinsip pemisahan tanggung jawab antar kelas (separation of concerns), sehingga setiap kelas memiliki fungsi sesuai perannya. Kelas Model mengelola akses data, sedangkan kelas Controller menangani logika proses dan interaksi dengan tampilan. Pendekatan ini memudahkan pengembangan, pengujian, dan pemeliharaan sistem karena perubahan pada satu bagian aplikasi tidak memengaruhi struktur program secara signifikan. Gambar 1 menunjukkan alur kerja arsitektur aplikasi.



Gambar 1. Alur Kerja Arsitektur

Pada Gambar 1, permintaan dari pengguna pertama kali diterima oleh public/index.php sebagai front controller, kemudian diteruskan ke router pada kelas App untuk menentukan controller dan method yang sesuai berdasarkan URL. Controller selanjutnya memanggil model untuk mengambil atau mengolah data pada basis data. Setelah proses pengolahan data selesai, controller meneruskan data

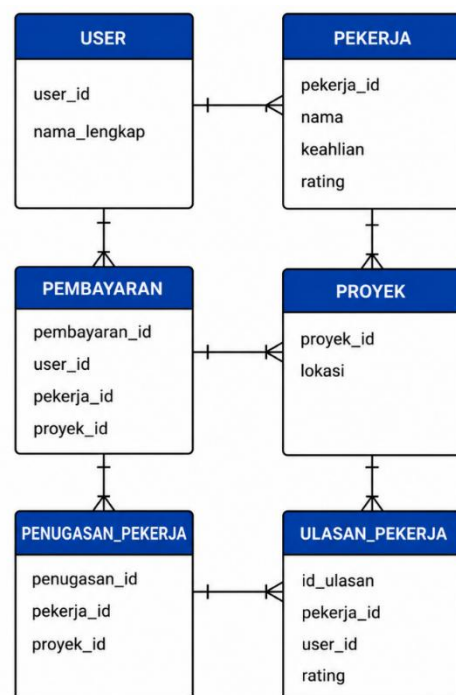
ke view untuk ditampilkan kembali kepada pengguna.

Basis data sistem dirancang untuk mendukung pengelolaan data pengguna, pekerja, proyek, penugasan, ulasan, dan pembayaran. Seperti ditunjukkan pada Gambar 2, basis data terdiri atas enam tabel utama, yaitu user, pekerja, proyek, penugasan_pekerja, ulasan_pekerja, dan pembayaran.

Tabel user digunakan untuk menyimpan data pengguna sistem, sedangkan tabel pekerja digunakan untuk menyimpan data tukang yang menjadi kandidat dalam proses rekomendasi. Tabel proyek digunakan untuk menyimpan informasi pekerjaan yang dibuat oleh pengguna. Selanjutnya, tabel penugasan_pekerja berfungsi menghubungkan data pekerja dengan proyek yang dikerjakan, sehingga hubungan antara pekerja dan proyek dapat dikelola dengan baik.

Tabel ulasan_pekerja digunakan untuk menyimpan penilaian pengguna terhadap pekerja dalam bentuk rating, sedangkan tabel pembayaran digunakan untuk mencatat transaksi yang berkaitan dengan pengguna, pekerja, dan proyek. Relasi antar tabel memungkinkan sistem mengelola proses pencarian tukang, penugasan proyek, pemberian ulasan, serta pembayaran secara terintegrasi.

Data pekerja digunakan sebagai kandidat dalam proses rekomendasi Greedy Matching. Sistem menghitung nilai kecocokan berdasarkan kriteria keahlian, jarak lokasi, dan rating kinerja, kemudian mengurutkan kandidat berdasarkan skor tertinggi untuk menghasilkan rekomendasi tukang yang relevan secara cepat dan terukur.



Gambar 2. ERD Aplikasi Ahli Tukang

Algoritma Greedy Matching digunakan untuk melakukan pemeringkatan kandidat tukang berdasarkan nilai kecocokan tertinggi. Pendekatan yang digunakan adalah score-based greedy ranking, yaitu sistem menghitung skor setiap kandidat berdasarkan kriteria yang telah ditentukan, kemudian mengurutkan kandidat dari skor tertinggi ke skor terendah.

Tabel 2. Tiga kriteria yang digunakan dalam proses rekomendasi adalah sebagai berikut:

Kriteria	Simbol	Bobot	Keterangan
Kesesuaian keahlian	K	40%	Mengukur kesesuaian antara jenis pekerjaan yang dicari pengguna dan keahlian tukang.
Kedekatan lokasi	J	30%	Mengukur kedekatan lokasi tukang terhadap lokasi pekerjaan.
Rating kinerja	R	30%	Mengukur reputasi tukang berdasarkan nilai rating pengguna.

Bobot 40%, 30%, dan 30% ditentukan berdasarkan prioritas kebutuhan sistem. Kesesuaian keahlian diberi bobot tertinggi karena menjadi syarat utama dalam pemilihan tukang. Kedekatan lokasi dan rating kinerja diberi bobot seimbang karena keduanya menjadi faktor pendukung dalam menentukan rekomendasi.

Rumus Perhitungan Skor

Nilai akhir setiap kandidat dihitung menggunakan rumus:

$$V_i = (0,40 \times K_i) + (0,30 \times J_i) + (0,30 \times R_i) \quad (1)$$

Keterangan:

V_i = Nilai akhir kandidat tukang ke- i

K_i = Skor kesesuaian keahlian kandidat tukang ke- i

J_i = Skor kedekatan lokasi kandidat tukang ke- i

R_i = Skor rating kinerja kandidat tukang ke- i

Seluruh nilai kriteria dinormalisasi ke dalam skala 0–100 agar dapat dihitung dalam satu skala yang sama.

Tabel 3. Skor kesesuaian keahlian

Kondisi Kesesuaian	Skor
Keahlian sesuai dengan kebutuhan pekerjaan	100
Keahlian sebagian sesuai	50
Keahlian tidak sesuai	0

Tabel 4. Skor kedekatan lokasi ditentukan sebagai berikut:

Jarak	Skor
≤ 2 km	100
> 2 km sampai 5 km	70
> 5 km sampai 10 km	33,33
> 10 km	0

Skor rating dihitung menggunakan rumus:

$$R_i = \frac{rating_i}{5} \times 100 \quad (2)$$

Dengan rumus tersebut, Rating 4,5 menghasilkan skor 90, sedangkan rating 4,0 menghasilkan skor 80.

Pseudo-code algoritma *Greedy Matching* disajikan sebagai berikut.

Pseudocode Greedy Matching Tukang

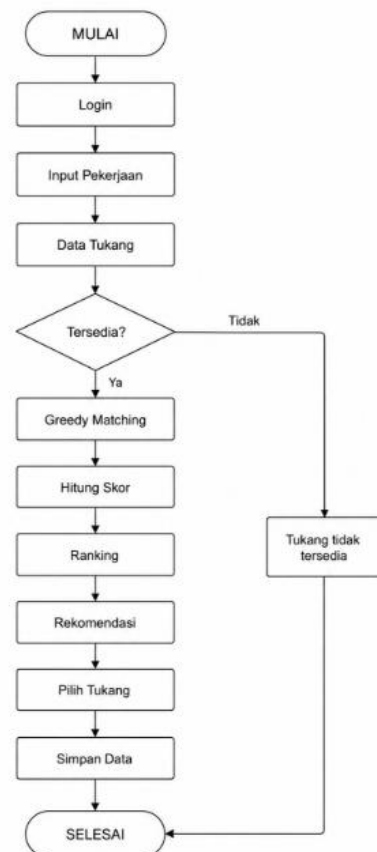
Mulai

1. Pengguna melakukan login
2. Input jenis dan lokasi pekerjaan
3. Ambil data tukang
4. Periksa ketersediaan tukang
5. Jika tukang tersedia, lakukan Greedy Matching:
 - a. Hitung skor keahlian (K)
 - b. Hitung skor jarak (J)
 - c. Hitung skor rating (R)
 - d. Hitung skor akhir:

$$V = (0,40 \times K) + (0,30 \times J) + (0,30 \times R)$$
6. Urutkan tukang berdasarkan skor tertinggi
7. Tampilkan rekomendasi tukang
8. Pengguna memilih tukang
9. Simpan data pemilihan

Selesai

Alur algoritma *Greedy Matching* adalah sebagai berikut:



Gambar 3. Alur Kerja Algoritma Greedy Matching

Pada Gambar 3, proses dimulai dari input kebutuhan pengguna berupa jenis pekerjaan dan lokasi pekerjaan. Sistem mengambil data tukang

yang tersedia dari database, kemudian menghitung skor setiap kandidat berdasarkan kesesuaian keahlian, kedekatan lokasi, dan rating kinerja. Setelah nilai akhir dihitung, sistem mengurutkan kandidat berdasarkan skor tertinggi dan menampilkan daftar rekomendasi kepada pengguna.

Kompleksitas Algoritma
Jika jumlah kandidat tukang yang tersedia adalah n , maka proses perhitungan skor untuk seluruh kandidat membutuhkan waktu sebesar $O(n)$. Setelah skor dihitung, sistem melakukan pengurutan kandidat berdasarkan nilai akhir. Proses pengurutan memiliki kompleksitas $O(n \log n)$. Dengan demikian, kompleksitas waktu keseluruhan algoritma adalah:

$$O(n) + O(n \log n) = O(n \log n) \quad (3)$$

Kompleksitas ruang algoritma adalah $O(n)$, karena sistem menyimpan daftar kandidat dan skor hasil perhitungan sebelum ditampilkan sebagai rekomendasi. Jika sistem hanya membutuhkan satu kandidat terbaik tanpa menampilkan daftar peringkat, proses pengurutan dapat diganti dengan pencarian nilai maksimum sehingga kompleksitas waktu menjadi $O(n)$. Namun, pada penelitian ini proses pengurutan tetap digunakan karena sistem menampilkan daftar rekomendasi tukang secara berurutan.

Algoritma Greedy Matching memiliki keterbatasan karena pemilihan kandidat berdasarkan nilai tertinggi pada saat perhitungan dilakukan. Oleh karena itu, hasil rekomendasi bersifat optimal lokal dan tidak selalu menjamin solusi optimal global. Hasil rekomendasi dapat dipengaruhi oleh bobot kriteria yang digunakan. Misalnya, kandidat dengan keahlian sesuai dapat memperoleh skor tinggi meskipun jaraknya lebih jauh, atau kandidat dengan rating tinggi dapat berada pada posisi atas meskipun tidak selalu menjadi pilihan terbaik dalam semua kondisi.

Untuk mengurangi bias, penelitian ini menggunakan normalisasi skor dan pembobotan eksplisit pada setiap kriteria. Namun, validasi bobot menggunakan pakar, metode AHP, TOPSIS, atau pengujian berbasis data historis dapat dilakukan pada penelitian selanjutnya.

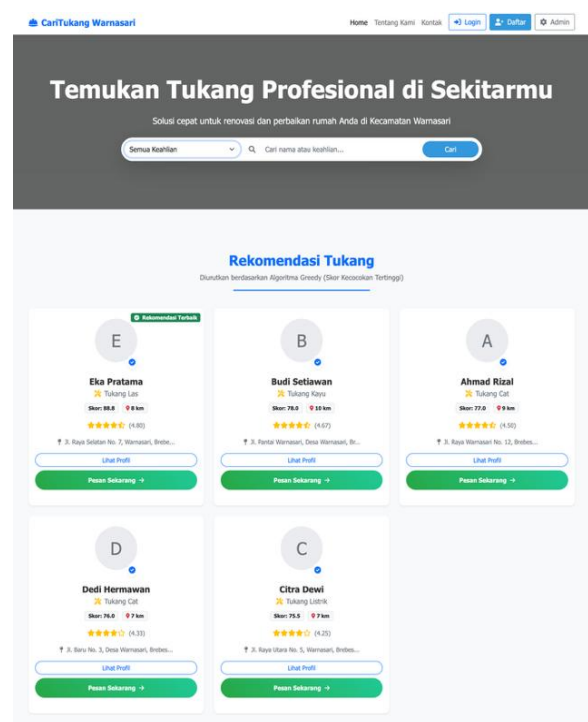
Pengujian sistem dilakukan menggunakan dua pendekatan. Pertama, *Black Box Testing* digunakan untuk menguji validitas fungsi sistem, meliputi proses registrasi, login, pencarian tukang, pengelolaan data pekerja, pengelolaan proyek, penugasan pekerja, pembayaran, dan tampilan detail tukang. Pengujian ini bertujuan memastikan bahwa setiap fitur menghasilkan keluaran sesuai dengan rancangan.

Kedua, dilakukan simulasi perhitungan Greedy Matching untuk mengevaluasi keluaran pemeringkatan kandidat. Simulasi dilakukan dengan membandingkan skor beberapa kandidat

berdasarkan kriteria keahlian, jarak, dan rating. Kandidat dengan nilai akhir tertinggi ditempatkan pada urutan pertama sebagai rekomendasi utama. Dengan demikian, pengujian tidak hanya menilai validitas fungsi sistem, tetapi juga menilai apakah proses pemeringkatan kandidat telah berjalan sesuai dengan rumus Greedy Matching.

HASIL DAN PEMBAHASAN

Hasil dari tahap Construction pada metode RAD adalah aplikasi pencarian jasa tukang berbasis web yang terintegrasi dengan algoritma Greedy Matching. Halaman utama menyediakan fitur pencarian berdasarkan keahlian dan menampilkan daftar rekomendasi tukang yang telah diurutkan berdasarkan skor kecocokan.



Gambar 4. Hasil Pencarian dengan Algoritma Greedy

Berdasarkan Gambar 4, sistem menampilkan beberapa kandidat tukang yang telah diurutkan berdasarkan skor hasil perhitungan algoritma Greedy Matching. Proses pemeringkatan dilakukan menggunakan kombinasi kriteria keahlian, jarak lokasi, dan rating kinerja. Tukang dengan skor tertinggi ditempatkan pada urutan pertama dan diberikan penanda "Rekomendasi Terbaik" sebagai indikator bahwa kandidat tersebut memiliki tingkat kecocokan paling tinggi terhadap kebutuhan pengguna.

Pada skenario pengujian ini, tukang bernama Eka Pratama menempati peringkat pertama dengan skor 88,5, lebih tinggi dibandingkan kandidat lain yang memperoleh skor 78,0 dan 75,0. Hasil tersebut menunjukkan bahwa algoritma tidak hanya

mempertimbangkan satu kriteria, tetapi menggabungkan seluruh kriteria yang digunakan dalam proses perhitungan skor. Dengan demikian, rekomendasi yang dihasilkan menjadi lebih objektif dibandingkan proses pemilihan secara manual.

Selain menghasilkan pemeringkatan kandidat, sistem juga mampu menampilkan rekomendasi secara dinamis ketika data pekerja, rating, atau lokasi berubah. Hal ini menunjukkan bahwa algoritma Greedy Matching dapat beradaptasi terhadap perubahan data dan tetap menghasilkan urutan rekomendasi berdasarkan skor kecocokan terbaru. Dari sisi implementasi, proses rekomendasi dapat ditampilkan secara langsung pada halaman utama sehingga memudahkan pengguna dalam memilih tukang yang sesuai dengan kebutuhan pekerjaan.

Untuk memperoleh evaluasi yang lebih komprehensif, pengujian tidak hanya dilakukan pada dua kandidat pekerja, tetapi diperluas menggunakan dataset penelitian berukuran 250, 500, dan 1000 kandidat. Dataset dibangun berdasarkan karakteristik data aktual sistem yang mencakup variasi keahlian, lokasi, rating kinerja, dan status ketersediaan pekerja. Pengujian dilakukan melalui 30 simulasi pencarian dengan kebutuhan keahlian Tukang Cat, Tukang Batu, Tukang Kayu, Tukang Keramik, dan Tukang Listrik pada wilayah Warnasari.

Tabel 5. Perhitungan Skor Greedy Matching

Ukuran Dataset	Kandidat Tersedia	Skor Maksimum	Skor Rata-rata	Waktu Eksekusi (ms)
250 Kandidat	202	80,00	43,84	0,44
500 Kandidat	399	80,00	41,91	1,07
1000 Kandidat	798	80,00	41,78	2,31

Berdasarkan Tabel 5, algoritma mampu menghasilkan rekomendasi yang konsisten pada berbagai ukuran dataset. Skor maksimum tetap berada pada rentang optimal sebesar 80,00 pada seluruh skenario pengujian, menunjukkan bahwa sistem mampu menemukan kandidat dengan tingkat kecocokan tertinggi meskipun jumlah pekerja meningkat secara signifikan. Selain itu, jumlah kandidat yang dievaluasi bertambah dari 202 pekerja tersedia pada dataset 250 kandidat menjadi 798 pekerja tersedia pada dataset 1000 kandidat.

Dari sisi performa, peningkatan jumlah kandidat menyebabkan waktu eksekusi bertambah dari 0,44 ms menjadi 2,31 ms. Meskipun demikian, waktu respons tersebut masih berada pada kategori sangat cepat dan layak digunakan untuk proses rekomendasi secara real-time. Hasil ini menunjukkan bahwa algoritma Greedy Matching memiliki kemampuan skalabilitas yang baik dalam

melakukan pemeringkatan kandidat berdasarkan kriteria keahlian, jarak lokasi, dan rating kinerja.

Berbeda dengan pengujian sebelumnya yang hanya menggunakan dua kandidat, pengujian pada tiga skala dataset memberikan gambaran yang lebih representatif mengenai perilaku algoritma dalam menangani jumlah data yang lebih besar. Namun, penelitian ini belum melakukan perbandingan langsung dengan metode pengambilan keputusan lain seperti AHP atau TOPSIS. Oleh karena itu, evaluasi yang dilakukan difokuskan pada konsistensi hasil rekomendasi dan kinerja komputasi algoritma Greedy Matching pada berbagai ukuran dataset.

Pengujian Sistem Pengujian dilakukan menggunakan metode *Black Box Testing* dengan menggunakan beberapa pendekatan. Pertama, Black Box Testing digunakan untuk memverifikasi kesesuaian fungsi sistem terhadap kebutuhan yang telah dirancang, meliputi proses registrasi, login, pencarian tukang, tampilan detail tukang, pengelolaan proyek, pembayaran, dan pemberian ulasan. Kedua, User Testing dilakukan untuk memperoleh umpan balik pengguna terhadap kemudahan penggunaan sistem dan relevansi rekomendasi yang dihasilkan. Ketiga, pengujian akurasi rekomendasi dilakukan dengan membandingkan hasil pemeringkatan yang dihasilkan algoritma Greedy Matching terhadap kriteria keahlian, jarak lokasi, dan rating kinerja yang digunakan dalam proses seleksi. Selain itu, dilakukan pengukuran response time untuk mengetahui waktu yang dibutuhkan sistem dalam menghasilkan rekomendasi sejak pengguna memasukkan kebutuhan pekerjaan hingga rekomendasi ditampilkan pada layar.

Tabel 6. Black Box Testing

No	Skenario Pengujian	Hasil yang Diharapkan	Hasil
1	Registrasi Pengguna	Data pengguna tersimpan	Valid
2	Login Sistem	Pengguna berhasil masuk	Valid
3	Pencarian Tukang	Sistem menampilkan daftar tukang	Valid
4	Detail Tukang	Informasi tukang ditampilkan	Valid
5	Pengelolaan Proyek	Data proyek tersimpan	Valid
6	Penugasan Pekerja	Data penugasan tersimpan	Valid
7	Pembayaran	Data pembayaran tersimpan	Valid
8	Pemberian Ulasan	Data ulasan tersimpan	Valid

Berdasarkan hasil Black Box Testing, seluruh fungsi utama sistem berjalan sesuai kebutuhan yang dirancang. Pengujian ini menunjukkan sistem mampu menerima input, memproses data, dan menghasilkan keluaran yang sesuai dengan spesifikasi fungsional.

Tabel 7. User Testing

No	Aspek Penilaian	Rata-rata Skor
1	Kemudahan Penggunaan	4,4
2	Kemudahan Navigasi	4,3
3	Kejelasan Informasi	4,5
4	Relevansi Rekomendasi	4,2
5	Kepuasan Pengguna	4,4

Skor menggunakan skala Likert 1–5. Hasil menunjukkan bahwa pengguna menilai sistem mudah digunakan dan rekomendasi yang dihasilkan cukup relevan dengan kebutuhan pekerjaan yang dimasukkan.

Tabel 8. Pengujian Response Time

No	Skenario	Waktu Respon (detik)
1	Menampilkan Data Tukang	0,42
2	Menghitung Skor Greedy Matching	0,31
3	Menampilkan Rekomendasi	0,48
4	Menyimpan Penugasan	0,27

Rata-rata waktu respon sistem sebesar 0,37 detik. Hasil ini menunjukkan bahwa sistem mampu menghasilkan rekomendasi dan memproses data dengan waktu respon yang relatif cepat.

SIMPULAN DAN SARAN

Berdasarkan hasil penelitian yang telah dilakukan, dapat disimpulkan bahwa sistem rekomendasi jasa tukang bangunan berbasis web berhasil dibangun menggunakan metode Rapid Application Development (RAD) dan mampu mendukung proses pencarian, pemilihan, serta pengelolaan data pekerja secara terintegrasi. Penerapan algoritma Greedy Matching pada sistem mampu menghasilkan pemeringkatan kandidat tukang berdasarkan beberapa kriteria, yaitu keahlian, jarak lokasi, dan rating kinerja. Berdasarkan hasil pengujian menggunakan dataset yang terdiri atas 250, 500, dan 1000 kandidat, algoritma mampu mempertahankan konsistensi hasil rekomendasi dengan skor maksimum sebesar 80,00 pada seluruh skenario pengujian. Dari sisi kinerja, waktu eksekusi mengalami peningkatan seiring dengan bertambahnya jumlah kandidat, yaitu dari 0,44 ms pada dataset 250 kandidat menjadi 2,31 ms pada dataset 1000 kandidat. Meskipun demikian, hasil tersebut menunjukkan bahwa sistem tetap mampu menghasilkan rekomendasi secara cepat meskipun jumlah kandidat yang diproses semakin besar.

Berdasarkan keterbatasan yang ditemukan dalam penelitian ini, pengembangan selanjutnya disarankan untuk mengintegrasikan layanan peta digital berbasis API agar perhitungan jarak antara pengguna dan tukang dapat dilakukan secara lebih akurat dan real-time. Selain itu, penelitian selanjutnya dapat melakukan perbandingan

algoritma Greedy Matching dengan metode pengambilan keputusan lainnya, seperti Analytical Hierarchy Process (AHP) atau Technique for Order Preference by Similarity to Ideal Solution (TOPSIS), sehingga evaluasi terhadap performa dan kualitas rekomendasi dapat dilakukan secara lebih komprehensif. Pengembangan sistem dalam bentuk aplikasi berbasis mobile juga dapat dilakukan untuk meningkatkan kemudahan dan fleksibilitas penggunaan. Selain itu, pemanfaatan data historis pengguna, seperti riwayat pencarian, pemilihan jasa, dan penilaian terhadap tukang, dapat dikembangkan untuk meningkatkan personalisasi serta kualitas rekomendasi yang dihasilkan oleh sistem.

DAFTAR PUSTAKA

- Ali Fikri, M., Lubis Ghazali, A., & Aldi Wibowo, I. (2025). *Sistem Rekomendasi Pencarian Kerja dengan Metode Simple Additive Weighting Berbasis Website: Studi Kasus SMKN 1 Terisi*. 11(3).
<https://doi.org/10.31884/jtt.v11i3.1142>
- Angul, A., Fallo, D., Tanggo, K. V., Belo Alves Nazario, I., & Hoar, F. (2025). Implementasi Algoritma Dijkstra dan Greedy dalam Penyelesaian Masalah Rute Terpendek. *Jurnal Kridatama Sains Dan Teknologi*, 7(1), 489–496.
- Fulber-Garcia, V., Camargo, F. E., & Duarte, E. P. (2022). Sk-Greedy: A Heuristic Scheduling Algorithm for Wireless Networks under the SINR Model. *ACM International Conference Proceeding Series*, 143–148.
<https://doi.org/10.1145/3569902.3570190>
- Gracia Sembiring, F., Putri Syaifullah Nst, S., Shaleh Lbn Gaol, A., & Harliana, P. (2025). IMPLEMENTASI ALGORITMA GREEDY DALAM PENENTUAN JALUR STRATEGIS DARI SIMPANG POS KE LAPANGAN MERDEKA KOTA MEDAN. In *Jurnal Mahasiswa Teknik Informatika* (Vol. 9, Number 1).
- Hatmara Sukma, S., Suarna, N., Bahtiar, A., Pramudya Marta, P., & Anam, K. (2026). *MODEL PREDIKSI PENERIMA BANTUAN SOSIAL BERBASIS ALGORITMA RANDOM FOREST*. 6(1), 45–49.
- Komen, J. J., Belitser, S. V., Wyss, R., Schneeweiss, S., Taams, A. C., Pajouheshnia, R., Forslund, T., & Klungel, O. H. (2021). Greedy caliper propensity score matching can yield variable estimates of the treatment-outcome association—A simulation study. *Pharmacoepidemiology and Drug Safety*, 30(7), 934–951.
<https://doi.org/10.1002/pds.5232>
- Mergawy, R. T., Hosny, H. E., & Abdelazeem, A. S. (2023). Decision Support Model for Contractor Selection. *The Open Civil Engineering Journal*, 17(1).

- <https://doi.org/10.2174/18741495-v17-e230215-2022-51>
- Rahmawan, A., Ramdhani, L., & Baha'udin. (2022). IMPLEMENTASI ALGORITMA GREEDY UNTUK PENENTUAN PRIORITAS LOKASI WISATA BERDASARKAN JARAK TERDEKAT. *INFOTIKA: Jurnal Pendidikan Informatika*, 1(1). <https://jurnal.habi.ac.id/index.php/Info>
- Sahnoun, I., & Elhadjamor, E. A. (2024). Enhanced Freelance Matching: Integrated Data Analysis and Machine Learning Techniques. *Journal of Computing Theories and Applications*, 1(4), 507–517. <https://doi.org/10.62411/jcta.10152>
- Saktia Purnama, R. D., Nisa, F., Tundo, T., Nurohman, K., Fakhurrofi, F., Nugrahaini, L., & Dalail, D. (2024). IMPLEMENTASI PENGGUNAAN ALGORITMA GREEDY BEST FIRST SEARCH UNTUK MENENTUKAN RUTE TERPENDEK DARI CILACAP KE YOGYAKARTA. *Jurnal Informatika Dan Teknik Elektro Terapan*, 12(2). <https://doi.org/10.23960/jitet.v12i2.4068>
- Sunanto, & Abadi Faural. (2021). OPTIMASI PENGATURAN STEAM UAP MENGGUNAKAN ALGORITMA GREEDY UNTUK MENDUKUNG PROSES PEREBUSAN KELAPA SAWIT. *Journal of Software Engineering and Information System (SEIS)*, 1(1), 30–35. <https://doi.org/10.37859/seis.v1i1.2815>
- Titasari Br Bangun, E., Arvia Luthfi, F., Sukesni, R., Arlen, Mr., Asral, Mz., Mikhraj, U., & Aulia, V. (2026). PERANCANGAN SISTEM INFORMASI LAYANAN SERVICE AC PADA PT.TEKNINDO ABADI PRATAMA BERBASIS WEB. 6(1). <https://ejurnal.umri.ac.id/index.php/SEIS/index>
- Vente, T., Mehta, Z., Wegmeth, L., & Beel, J. (2022). Greedy Ensemble Selection for Top-N Recommendations. 3924, 12–16.