

Implementasi Algoritma *Best First Search* untuk Pencarian Rute Terpendek pada Aplikasi Cerdas Pendaftaran Santri Baru

Herfandi^{1*}, Ulfatus Soleha², Agung Susilo Yuda Irawan², Kiki Ahmad Baihaqi³,
Reza Maulana⁴

^{1,2}Program Studi Teknik Informatika, Universitas Teknologi Sumbawa

³Program Studi Teknik Informatika, Universitas Singaperbangsa Karawang

⁴Program Studi Teknik Informatika, Sekolah Tinggi Teknologi Terpadu Nurul Fikri

Email: *herfandi@uts.ac.id

Abstrak. Pesantren di Indonesia merupakan lembaga pendidikan islam tradisional yang mampu eksis sampai saat ini. Proses pendaftaran yang kurang terstruktur memungkinkan kesalahan serta keterlambatan dalam proses pendaftaran santri baru akan mudah terjadi. Pondok Pesantren Manbaul Ulum dalam menjalankan proses administrasi pendidikan masih menggunakan cara konvensional hal ini menyebabkan sistem pelayanan yang digunakan belum efisien serta masalah keterbatasan asrama maka dibuatlah regulasi pendaftaran harus sesuai dengan zona jarak tempat tinggal yang telah ditentukan. Karenanya penelitian ini melakukan implementasi algoritma *best first search* untuk pencarian rute terpendek pada aplikasi cerdas pendaftaran santri baru. Penelitian ini menghasilkan aplikasi cerdas pendaftaran santri baru dengan menerapkan algoritma *best first search*. Metode pengembangan menggunakan *waterfall*. Pengujian perangkat lunak dengan metode *black box testing* dengan teknik *test case equivalence partitioning* mendapatkan kesimpulan berhasil dari berbagai jenis pengujian. Dengan fitur untuk santri yaitu data estimasi, *home*, petunjuk, *contact us*, mendaftar, *login*, mengisi formulir pendaftaran, lihat verifikasi, cetak bukti lulus, dan melihat rute terpendek, untuk *admin setting* tahun ajaran, lihat pendaftaran, lihat pendaftaran diterima dan ditolak serta melihat laporan. Sedangkan untuk *superadmin* yaitu CRUD data *admin* dan lihat laporan. Aplikasi ini diharapkan dapat membantu administrasi pesantren dan calon santri melihat rute serta mendukung pendidikan di era Industri 4.0.

Kata kunci: *Algoritma Best First Search, Aplikasi Cerdas, Test Case Equivalence Partitioning-Black Box Testing, Pesantren, Waterfall.*

1 Pendahuluan

Pesantren di Indonesia merupakan Lembaga Pendidikan Islam Tradisional yang mampu eksis dan tidak hilang ditelan zaman [1]. Masyarakat Indonesia masih menggunakan pesantren sebagai pilihan dalam menempuh pendidikan dikarenakan materi dan sistem pendidikan masih mempertahankan, mendalami dan menyebarkan ajaran agama Islam [2]. Pesantren berperan juga sebagai identitas pendidikan Islam di Indonesia dan memiliki makna keaslian

(*indigenous*) bagi Indonesia [3]. Situasi tersebut menjadikan pesantren sebagai lembaga pendidikan Islam tradisional harus mampu mengikuti perkembangan zaman agar bisa bertahan di masyarakat modern saat ini. Setiap lembaga pendidikan berlomba-lomba untuk menarik minat masyarakat dengan menawarkan materi, sistem pendidikan, dan kualitas pelayanan yang baik dalam hal penerapan teknologi, yang tentunya ini bertujuan untuk merespon pendidikan di era 4.0 [4]. Peningkatan persaingan tersebut secara tidak langsung mengharuskan lembaga pendidikan pesantren memiliki sistem pelayanan yang tepat dalam melakukan proses administrasi agar lebih efisien dan mampu mengurangi beban biaya yang otomatis berdampak pada penyesuaian terhadap revolusi industri keempat sehingga mampu mempertahankan keberadaannya di tengah-tengah masyarakat modern. Seiring dengan perkembangan teknologi informasi dalam kehidupan sehari-hari [5], maka menjadi hal yang mudah dalam berbagi informasi yang ada, serta dapat memudahkan pengelolaan data. Salah satunya ialah pada pendaftaran santri baru yang merupakan suatu proses awal dari para calon santri untuk menimba ilmu dalam naungan pondok pesantren [6]. Proses pendaftaran ini merupakan salah satu kegiatan yang menunjang kegiatan belajar mengajar nantinya. Jika pada proses pendaftaran kurang terstruktur, maka kemungkinan terdapat kesalahan dalam pendataan pendaftaran santri baru dan keterlambatan dalam proses pendaftaran santri baru [7].

Berdasarkan hasil pengamatan pada Lembaga Pendidikan Pondok Pesantren Manbaul Ulum pada daerah Kabupaten Bondowoso, dalam menjalankan proses administrasi pendidikan masih menggunakan cara konvensional. Pondok Pesantren inilah yang dijadikan sebagai studi kasus dalam penelitian ini. Dalam menjalankan proses pendaftaran santri baru masih memiliki beberapa kendala dikarenakan sistem pelayanan yang digunakan belum efisien. Dengan sistem yang berjalan saat ini para calon santri kesulitan untuk mendapatkan informasi terkait pendaftaran. Selama ini pada proses pendaftaran para calon santri diminta untuk mengisi lembar formulir pendaftaran dengan tulis tangan yang disediakan oleh panitia sehingga membutuhkan waktu lama dan pada tahap seleksi mengakibatkan keterlambatan dalam proses pengumuman. Persyaratan lain seperti berkas yang telah ditentukan sebagai salah satu persyaratan masih disimpan pada map masing-masing calon santri sehingga dapat menimbulkan kesalahan dalam tahap seleksi dikarenakan harus mengecek satu persatu berkas yang telah dikumpulkan. Bukan hanya pada proses pendaftaran, akan tetapi banyak para calon santri tidak mengetahui rute pada lokasi pesantren yang berada di pedesaan. serta mengingat keterbatasan asrama untuk tempat tinggal santri maka dibuatlah batasan jarak untuk santri yang akan menetap sehingga banyak yang mendaftar tidak sesuai dengan zona jarak yang telah ditentukan. Maka karenanya tujuan dari penelitian ini melakukan implementasi algoritma *best first search* untuk pencarian rute terpendek pada aplikasi cerdas pendaftaran santri baru.

Penelitian terdahulu pada bidang *artificial intelligence*, aplikasi berbasis pengetahuan (sistem aplikasi cerdas) bermanfaat bagi lembaga pendidikan dalam membantu administrasi kesiswaan sekolah dan aplikasi bisa digunakan dimana dan kapan saja [8] serta untuk memperoleh informasi yang dikehendaki dan mendukung pendidikan di era Industri 4.0 [9]. Teknik dalam *searching* (pencarian) adalah dengan mendefinisikan ruang masalah yang digambarkan dalam himpunan keadaan (*state*) kemudian diubah ke dalam state lainnya sampai menemukan solusi [10]. Terdapat dua jenis metode dalam teknik *searching* yaitu *blind* atau *un-informed search* dan *heuristic* atau *informed search* [10]. Pencarian *heuristic* akan menghitung biaya estimasi dari suatu *state* tertentu untuk menuju ke *state* tujuan sehingga tidak membutuhkan memori yang besar dari sebuah masalah yang sederhana [11]. Algoritma *best first search* termasuk ke dalam teknik pencarian *heuristic* yang akan memilih simpul baru berdasarkan biaya terkecil dari semua *lead nodes* yang pernah di bangkitkan, hal ini mampu menghasilkan hasil yang optimal [12]. *Best first search* merupakan algoritma pencarian yang efektif dan optimal untuk pencarian rute terpendek dari masalah *searching* [13]. Karenanya algoritma yang digunakan dalam pencarian rute terpendek pada penelitian ini menggunakan *best first search*. Bagi para calon santri akan sangat membantu mengetahui rute pada lokasi pesantren yang berada di pedesaan, mengingat regulasi pendaftaran harus sesuai dengan zona jarak tempat tinggal yang telah ditentukan.

Penelitian sebelumnya yang dilakukan oleh peneliti lainnya terkait dengan yang diteliti saat ini adalah sebagai berikut, E. Nofarita (2021) mendeskripsikan tentang implementasi algoritma *best first search* secara umum. Artikel terdahulu ini hanya menjelaskan tentang ide dasar proses penggunaan algoritma *best first search* sehingga mendapatkan hasil yang optimal pada kasus *Traveling Salesman Problem* [12]. Penelitian terdahulu ini tidak menerapkan algoritma *best first search* pada program aplikasi, tidak sebagaimana penelitian dalam artikel ini adalah menerapkan algoritma *best first search* pada program aplikasi cerdas pendaftaran santri baru untuk pencarian rute terpendek berbasis web. N. Wibowo, C. E. Widodo, and K. Adi (2020) mencari titik rute terpendek di kota semarang untuk menghindari kemacetan sebagai studi kasus. Penelitian terdahulu ini memperkirakan jarak tempuh atau estimasi perjalanan antara titik awal sampai ke titik tujuan [13]. Berbeda dengan penelitian pada artikel ini mencari rute terpendek dengan algoritma *best first search*. Selain itu artikel ini membangun program aplikasi cerdas yang menjadi keunggulan dibandingkan dengan penelitian sebelumnya. Y. Afero et al. (2021) memperkenalkan tentang algoritma *best first search* untuk menentukan jalur terpendek. Artikel terdahulu ini hanya menjelaskan ide dasar *graph* pada algoritma *best first search* [14]. sedangkan artikel pada penelitian ini membangun aplikasi cerdas dengan algoritma *best first search* guna membantu para calon santri mengetahui rute pada lokasi pesantren. R. Apriandi et al. (2018) menerapkan metode *best first search* untuk mencari

lokasi SPBU terdekat. Perbedaannya dengan artikel ini adalah segi implementasi algoritma *best first search* serta sistem yang dibangun, pada artikel terdahulu ini menggunakan perangkat arduino berbasis *android* [15] dan sistem aplikasi yang dibangun tidak berbasis *website* sedangkan pada artikel ini berbasis *website* yang dibangun dengan *framework bootstrap* untuk tampilan *website* yang *responsive* dengan bahasa pemrograman PHP. Berdasarkan penelitian sebelumnya yang terkait dengan penelitian saat ini, memperlihatkan kekuatan dari penelitian ini adalah membangun aplikasi cerdas dengan algoritma *best first search* untuk menentukan rute terpendek.

2 Metodologi Penelitian

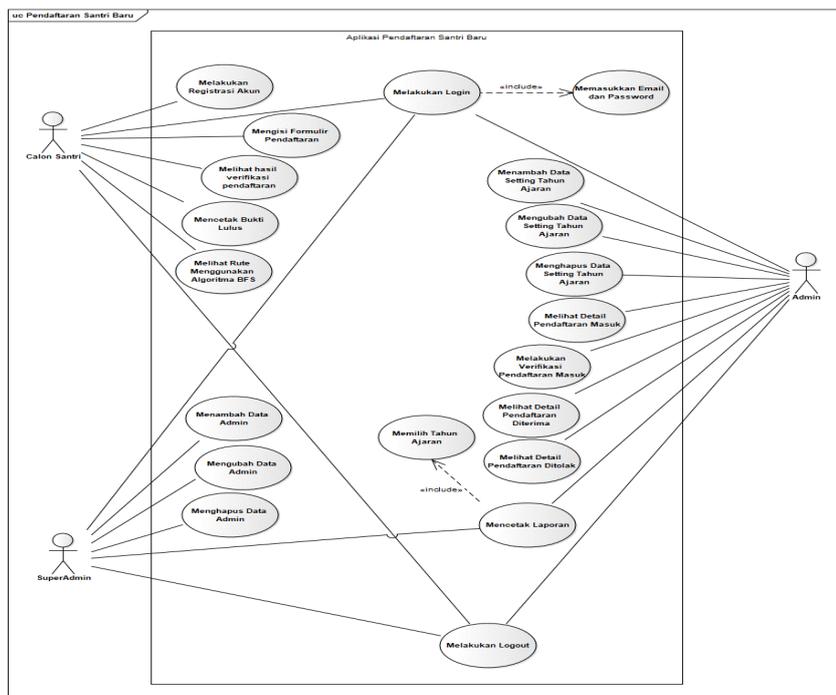
Pembangunan aplikasi cerdas menggunakan metode pengembangan *waterfall*, dikarenakan kebutuhan *user* di awal sudah diketahui sehingga proses pengembangan bisa berurutan [16]. Objek tempat penelitian yang dijadikan studi kasus pada Pondok Pesantren Manbaul Ulum merupakan salah satu yayasan pondok pesantren di daerah Kabupaten Bondowoso tepatnya Jl. KH. Togo Ambarsari, Tangsil Wetan, Rt. 01, Rw.01, Kecamatan Wonosari, Kabupaten Bondowoso yang dipimpin oleh KH. Salwa Arifin. Alamat ini yang menjadi *goal state* pada implementasi algoritma *best first search* pada Program aplikasi cerdas yang di bangun menggunakan *framework bootstrap* dan PHP dengan desain menggunakan pemodelan *Unified Modelling Language* (UML).

2.1 Analisis

Tahap Analisis dilakukan untuk mengetahui kebutuhan dari seluruh sistem yang akan di bangun dengan cara mengumpulkan data-data atau informasi terkait [17]. Metode pengumpulan data yang digunakan adalah kualitatif, dikarenakan menggunakan beberapa data yang diperoleh dan memanfaatkan teori-teori terdahulu. Adapun proses yang dilakukan [18]: (1) Observasi, setelah mengamati langsung proses atau alur kerja pendaftaran santri baru yang sedang berjalan, di temukan suatu kendala pada proses pendaftaran. Sehingga usulan panitia pendaftaran santri baru dengan membangun aplikasi pendaftaran santri baru berbasis web untuk calon santri yang akan mondok. (2) Wawancara secara langsung dengan Ustadz Hairul selaku bagian panitia pendaftaran santri baru, beliau mengungkapkan “Banyak calon santri yang tidak mengetahui lokasi pesantren serta adanya regulasi pendaftaran berdasarkan zona” karenanya algoritma *best first search* akan diterapkan pada aplikasi cerdas pendaftaran santri baru. (3) Studi Pustaka yaitu dengan mengambil data dan informasi dari berbagai sumber, terutama pada <https://scholar.google.com> yang bertujuan sebagai referensi untuk membangun aplikasi.

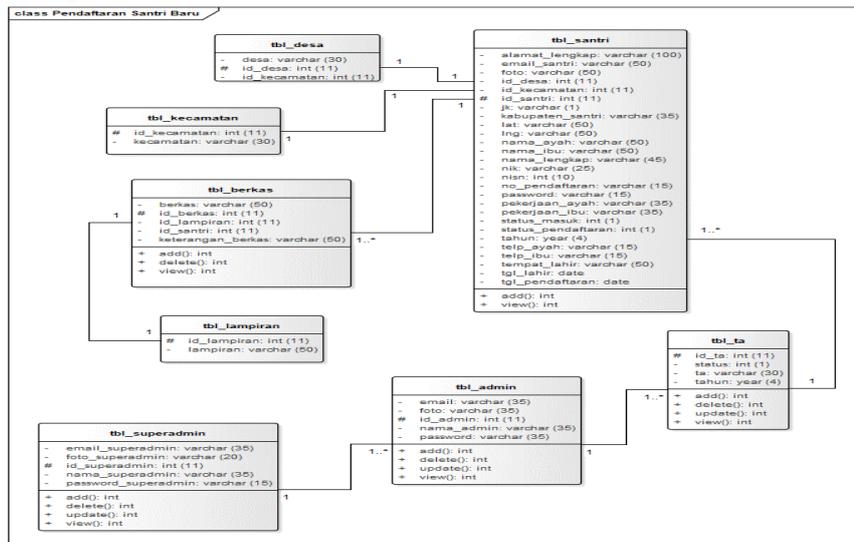
2.2 Desain

Teknik *Unified Modelling Language* (UML) digunakan pada tahap desain di karenakan pengembangan menggunakan paradigma pemrograman berorientasi objek [19]. Dalam membangun aplikasi cerdas pada penelitian ini, *Use Case Diagram* bisa dilihat pada Gambar 1.



Gambar 1. Use Case Diagram Aplikasi Pendaftaran Santri Baru

Pada Gambar 1 diatas merupakan *Use Case Diagram* dalam mengakses aplikasi pendaftaran santri baru pada Pesantren Manbaul Ulum yang menggambarkan tiga aktor yang saling terlibat pada aplikasi pendaftaran santri baru, diantaranya adalah calon santri, *admin*, dan *superadmin*. Peran aktor calon santri hanya dapat melakukan registrasi akun, melakukan *login*, melakukan pengisian formulir pendaftaran, melihat hasil verifikasi, mencetak bukti lulus pendaftaran, melihat rute dari tempat tinggal ke lokasi pesantren menggunakan algoritma *best first search*, serta melakukan *logout*. Sedangkan peran aktor *admin* ialah melakukan *login*, menambah, mengubah, serta menghapus data *setting* tahun ajaran, melihat detail pendaftaran masuk, melakukan verifikasi pendaftaran masuk, melihat detail pendaftaran diterima, melihat detail pendaftaran ditolak dan mencetak laporan pendaftaran, kemudian melakukan *logout*. Aktor yang terakhir adalah *superadmin* yang dapat melakukan *login*, menambah, mengubah, dan menghapus data *admin*, mencetak laporan pendaftaran, serta melakukan *logout*.



Gambar 2. Class Diagram Aplikasi Pendaftaran Santri Baru

Pada Gambar 2 merupakan *Class Diagram* dalam mengakses aplikasi pendaftaran santri baru pada Pesantren Manbaul Ulum yang menggambarkan setiap *class* dapat terlihat dan berhubungan antara *class* satu dengan *class* yang lain dan terdapat beberapa *class* yang memiliki fungsi lihat, tambah, ubah, dan hapus.

2.3 Pengkodean dan Pengujian

Tahap pengkodean menggunakan bahasa pemrograman PHP dan *framework bootstrap* [20]. *Text editor visual studio code* digunakan untuk implementasi aplikasi. Sedangkan pengujian menggunakan metode *black box testing* dengan teknik *equivalence partitioning* dikarenakan berfokus pada persyaratan fungsional perangkat lunak sehingga akan mendapatkan *output* yang sesuai dengan persyaratan fungsional pada aplikasi. Sistem *database* yang digunakan pada aplikasi adalah *MySQL* karena merupakan sistem *database* populer dan gratis [21].

3 Hasil dan Pembahasan

Hasil implementasi pada aplikasi cerdas pendaftaran santri baru meliputi beberapa bagian yaitu sebagai berikut:

3.1 Implementasi Halaman Menu Tampilan Calon Santri

Pada Gambar 3 merupakan implementasi tampilan awal calon santri sebelum masuk pada aplikasi pendaftaran santri baru.



Gambar 3. Implementasi Tampilan Awal Aplikasi

Aplikasi untuk calon santri memiliki fitur: (1) Data Estimasi: digunakan bagi calon santri untuk melihat data jumlah pendaftar laki-laki dan perempuan serta total keseluruhan pendaftar. (2) Home: untuk kembali kehalaman utama. (3) Petunjuk: cara penggunaan aplikasi. (3) Contact Us: untuk menghubungi admin PSB. (4) Mendaftar: digunakan untuk melakukan registrasi akun bagi calon santri yang belum memiliki akun untuk melakukan login. (5) Login : calon santri diharuskan mengisi *email* dan *password* yang sesuai dengan data yang telah di input pada menu Mendaftar untuk dapat mengakses pendaftaran santri baru, yang bisa di lihat pada Gambar 4.



Gambar 4. Implementasi Tampilan Login Calon Santri

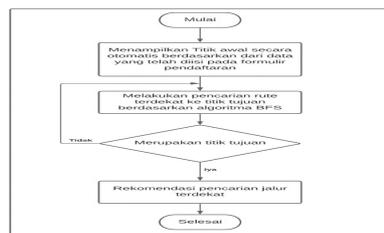
Setelah berhasil *login* maka calon santri akan diarahkan ke (5) Formulir Pendaftaran: dimana calon santri di haruskan mengisi data diri, data alamat tinggal dan berkas pendukung (persyaratan) sebelum mengajukan pendaftaran yang bisa dilihat pada Gambar 5.

Gambar 5. Implementasi Tampilan Formulir Pendaftaran

(6) Verifikasi: setelah melakukan pendaftaran maka calon santi bisa melihat verifikasi diterima atau ditolak dari *admin*. (7) Cetak Bukti Lulus: pendaftaran pada saat hasil verifikasi diterima, maka bukti lulus pendaftaran dapat dicetak. Serta (8) Melihat Rute Terpendek.

3.2 Implementasi Algoritma *Best First Search*

Flowchart dari penentuan jarak terdekat menggunakan algoritma *best first search* bisa di lihat pada Gambar 6.



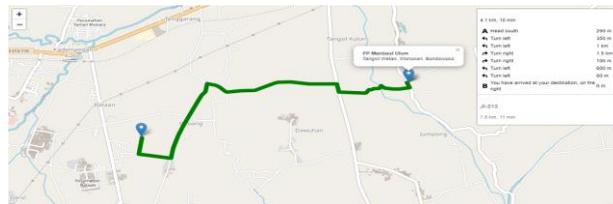
Gambar 6. Flowchart Implementasi Algoritma BFS

Dari gambar 6 dapat dijelaskan bahwa ketika calon santri setelah melakukan mulai kemudian sistem menampilkan titik awal secara otomatis berdasarkan dari titik koordinat dari data yang telah diisi pada formulir pendaftaran. Kemudian melakukan pencarian dari setiap jalur yang berhubungan dengan titik tujuan berdasarkan algoritma *best first search* maka setiap jalur terdapat kemungkinan bukan merupakan titik tujuan maka akan kembali pada tahap titik awal melakukan pencarian. Jika merupakan titik tujuan, maka rekomendasi pencarian jalur terdekat ditemukan. Maka pencarian selesai. Titik tujuan merupakan lokasi Pesantren Manbaul Ulum yang telah otomatis diterapkan pada aplikasi. Jadi ketika hasil verifikasi diterima, maka calon santri dapat melihat rute dari tempat tinggal ke pesantren menggunakan Algoritma *best first search* tanpa mengisi titik awal dan titik tujuan lagi. Berikut merupakan *source code* dari pencarian rute terpendek yang digunakan:

Tabel 2. *Source Code* Algoritma *Best First Search*

<pre> <script> let map = L.map('map').setView([<?=>= Slokasi[lat]?>, <?=>= Slokasi[lng]?>], 13); let latLang1 = L.latLng(<?=>= Slokasi[lat]?>, <?=>= Slokasi[lng]?>); let latLang2 = L.latLng(-7.917773682624171, 113.87320144178913); let wp1 = new L.Routing.Waypoint(latLang1); let wp2 = new L.Routing.Waypoint(latLang2); L.tileLayer('https://s.tile.openstreetmap.org/{z}/{x}/{y}.png', { attribution: '&copy; OpenStreetMap contributors' }).addTo(map); L.Routing.control({ waypoints: [latLang1, latLang2] }).addTo(map); let routeUs = L.Routing.osrmv1(); routeUs.route(wp1, wp2, {err.routes=>{ if(!err) { let best = 1000000000000000; let bestRoute = 0; for(i in routes) </pre>	<pre> if(routes[i].summary.totalDistance < best) { bestRoute = i; best = routes[i].summary.totalDistance; } } console.log('best route', routes[bestRoute]); L.Routing.line(routes[bestRoute], { styles: [{ color: 'green', weight: 10 }] }).addTo(map); L.marker([-7.917773682624171, 113.87320144178913]).addTo(map) .bindPopup('PP Manbaul Ulum
Tangsil Wetan, Wonosari, Bondowoso').openPopup(); </script> </pre>
--	---

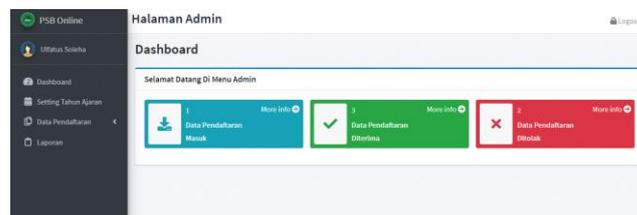
Dari Tabel 1 dapat dijelaskan bahwa let latLng1 merupakan titik awal atau tempat tinggal calon santri yang telah tersimpan pada data formulir pendaftaran, sedangkan let latLng2 merupakan titik tujuan yaitu Pesantren Manbaul Ulum. Pada pengambilan titik awal dan tujuan ini bersifat otomatis, karena telah ditentukan pada masing-masing titik. ketika routes[i] lebih kecil dari best, maka jarak terdekat (best) telah ditemukan dan jika sebaliknya, maka akan mencari jarak lain yang lebih dekat. Gambar 7 merupakan implementasi dari tampilan rute dari tempat tinggal santri ke pesantren menggunakan algoritma *best first search* yang dapat dilihat setelah hasil verifikasi diterima.



Gambar 7. Implementasi Tampilan Rute

3.3 Implementasi Halaman Menu Tampilan Admin

Pada Gambar 8 merupakan implementasi tampilan *Dashboard Admin* sesudah melakukan *login*.

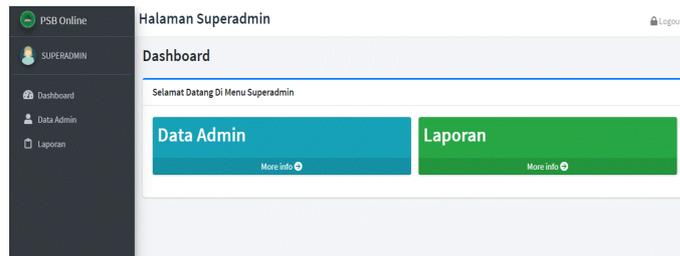


Gambar 8. Implementasi Tampilan *Dashboard Admin*

Adapun fitur yang dimiliki adalah: (1) *Setting Tahun Ajaran*: digunakan *admin* untuk menampilkan, tambah, ubah, hapus data *setting* tahun ajaran. (2) *Data Pendaftaran*: untuk menampilkan data pendaftaran masuk calon santri yang telah melakukan pengajuan pendaftaran, *admin* bisa melakukan aksi terima dan tolak serta melihat detail. (3) *Pendaftaran Diterima dan Ditolak*: *admin* bisa melihat data pendaftaran yang diterima dan ditolak. (4) *Laporan*: untuk mencetak laporan pendaftaran yang diterima sesuai dengan tahun yang ingin dicetak.

3.4 Implementasi Halaman Menu Tampilan Superadmin

Pada Gambar 9 merupakan implementasi tampilan *Dashboard Superadmin* sesudah melakukan *login*.



Gambar 9. Implementasi Tampilan *Dashboard Superadmin*

Adapun fitur yang dimiliki adalah: (1) *Data Admin*: digunakan untuk menampilkan, tambah, ubah, hapus data *admin*. (2) *Laporan*: untuk mencetak laporan pendaftaran yang diterima sesuai dengan tahun yang ingin dicetak.

4 Pengujian Aplikasi

Pada pengujian aplikasi ini menggunakan metode *black box testing* dengan teknik *test case equivalence partitioning* [22]. Berikut merupakan hasil tahapan yang dilakukan dalam pengujian *black box* menggunakan teknik *test case equivalence partitioning*:

Tabel 2. Hasil Pengujian *Test Case Equivalence Partitioning- Black Box Testing*

Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengujian	Keterangan
ID A0-9: Hasil pengujian registrasi akun			
Mengisi semua data registrasi akun dengan benar, lalu klik tombol "Mendaftar"	Berhasil mendaftar akun	Berhasil mendaftar akun	Berhasil
ID B0-5: Hasil pengujian <i>login</i> calon santri			
Memasukkan <i>email</i> dan <i>password</i> dengan benar sesuai yang terdaftar, lalu klik tombol "Login"	Berhasil masuk ke halaman formulir pendaftaran santri baru	Berhasil masuk ke halaman formulir pendaftaran santri baru	Berhasil
ID C0-7: Hasil pengujian formulir pendaftaran			
Mengisi semua data alamat lengkap	Berhasil menyimpan data alamat lengkap	Berhasil menyimpan data alamat lengkap	Berhasil
ID D0-5: Hasil pengujian <i>login admin</i>			
Memasukkan <i>email</i> dan <i>password</i> dengan benar sesuai yang terdaftar, lalu klik tombol "Login"	Berhasil masuk ke dalam halaman <i>admin</i>	Berhasil masuk ke dalam halaman <i>admin</i>	Berhasil
ID E0-5: Hasil pengujian <i>login superadmin</i>			
Memasukkan <i>email</i> dan <i>password</i> dengan benar sesuai yang terdaftar, lalu klik tombol "Login"	Berhasil masuk ke dalam halaman <i>admin</i>	Berhasil masuk ke dalam halaman <i>admin</i>	Berhasil

5 Kesimpulan

Berdasarkan penelitian yang telah dilakukan maka ditarik kesimpulan: (1) Metode pengembangan menggunakan *waterfall*. (2) Aplikasi cerdas pendaftaran santri baru berhasil dibangun dengan menerapkan algoritma *best first search* untuk pencarian rute terpendek. (3) Pengujian perangkat lunak dengan metode metode *black box testing* dengan teknik *test case equivalence partitioning* mendapatkan kesimpulan berhasil dari berbagai jenis pengujian. (4) Aplikasi yang dibangun memiliki fitur untuk santri yaitu Data Estimasi, Home, Petunjuk, Contact Us, Mendaftar, Login, mengisi Formulir Pendaftaran, lihat Verifikasi, Cetak Bukti Lulus, dan Melihat Rute Terpendek, untuk *admin Setting* Tahun Ajaran, lihat Pendaftaran, lihat Pendaftaran Diterima dan Ditolak serta melihat Laporan. Sedangkan untuk *superadmin* yaitu CRUD Data Admin dan lihat Laporan. Penelitian ini diperlukan pengembangan lebih lanjut. Karenanya diharapkan kedepannya bisa menambahkan fitur chat dan konfirmasi pembayaran.

6 Referensi

- [1] A. Rindiani, A. Nurwadjah, A. Suhartini, U. Sunan, and G. Djati Bandung, "Eksistensi dan Revitalisasi Pesantren di Era 4.0," *Jurnal Dirosah Islamiyah*, vol. 4, p. 78, 2022, doi: 10.17467/jdi.v4i1.679.
- [2] J. Pendidikan dan Kewirausahaan, R. Nurfarida, M. Tisna Nugraha, A. Suhartini, N. Ahmad, and U. Islam Negeri Sunan Gunung Djati Bandung, "Karakteristik Pondok Pondok Pesantren Al-Ihsan Dalam Tinjauan Analisis SWOT ... Karakteristik Pondok Pondok Pesantren Al-Ihsan Dalam Tinjauan Analisis Swot," vol. 10, pp. 2022–162, 2022, doi: 10.47668/pkwu.v9i1.334.
- [3] M. Falikul Isbah, "Pesantren in the changing indonesian context: History and current developments," *Qudus International Journal of Islamic Studies*, vol. 8, no. 1, pp. 65–106, 2020, doi: 10.21043/QIJS.V8I1.5629.
- [4] D. Lase, "Pendidikan di Era Revolusi Industri 4.0," *SUNDERMANN: Jurnal Ilmiah Teologi, Pendidikan, Sains, Humaniora dan Kebudayaan*, vol. 12, no. 2, pp. 28–43, Nov. 2019, doi: 10.36588/sundermann.v1i1.18.
- [5] A. Susilo Yuda Irawan, N. Heryana, H. Siti Hopipah, D. Rahma Putri, and J. Hs Ronggo Waluyo Puseurjaya Telukjambe Timur Karawang Jawa Barat, "Identifikasi Website Phishing dengan Perbandingan Algoritma Klasifikasi," 2021. [Online]. Available: www.phishtank.com
- [6] P. Wijaya, R. Widiya Sembiring, S. Tunas Bangsa, S. Utara, A. Tunas Bangsa, and I. A. Jln Sudirman Blok No, "Analisis Metode Backpropagation Memprediksi Penerimaan Santri/Wati di Pondok Pesantren Modern Al-Kautsar," vol. 6, pp. 140–149, [Online]. Available: <https://tunasbangsa.ac.id/ejurnal/index.php/jurasik>
- [7] I. Herliana, H. Qosim Mubah, and A. Ahmadi, "MANAJEMEN SISTEM INFORMASI DALAM KEGIATAN PENERIMAAN SANTRI BARU DI PONDOK PESANTREN PUTERI KHADIJAH PAMEKASAN," *re-JIEM*

- (*Research Journal of Islamic Education Management*), vol. 4, no. 1, pp. 48–59, Jun. 2021, doi: 10.19105/re-jiem.v4i1.4843.
- [8] C. Satria and A. Anggrawan, “Aplikasi K-Means berbasis Web untuk Klasifikasi Kelas Unggulan,” *MATRIK: Jurnal Manajemen, Teknik Informatika dan Rekayasa Komputer*, vol. 21, no. 1, pp. 111–124, Nov. 2021, doi: 10.30812/matrik.v21i1.1473.
- [9] S. M. Lee, D. Lee, and Y. S. Kim, “The quality management ecosystem for predictive maintenance in the Industry 4.0 era,” *International Journal of Quality Innovation*, vol. 5, no. 1, Dec. 2019, doi: 10.1186/s40887-019-0029-5.
- [10] Suyanto, *ARTIFICIAL INTELLIGENCE*, Edisi 3. Bandung, Indonesia : Informatika Bandung, 2021.
- [11] Suyanto, *Artificial Intelligence*, Edisi ke II. Bandung, Indonesia: Informatika Bandung, 2014.
- [12] E. Nofarita, “The Utilization of The Best First Search Algorithm in The Solution of The Traveling Salesman Problem Case In City X,” *International Journal of Information System & Technology Akreditasi*, vol. 5, no. 4, pp. 360–365, 2021.
- [13] N. Wibowo, C. E. Widodo, and K. Adi, “Implementing the Shortest Time Route Search Algorithm in Semarang Using the Best First Search Method,” in *IOP Conference Series: Materials Science and Engineering*, Aug. 2020, vol. 879, no. 1. doi: 10.1088/1757-899X/879/1/012165.
- [14] Y. Afero *et al.*, “ALGORITMA BEST FIRST SEARCH MENENTUKAN LINTASAN JALUR TERPENDEK PADA KOTA WISATA BUKITTINGGI,” *JOISIE Journal Of Information System And Informatics Engineering*, vol. 5, no. Desember, pp. 138–145, 2021.
- [15] R. Apriandi *et al.*, “PENERAPAN METODE BEST FIRST SEARCH (BFS) UNTUK PENCARIAN LOKASI SPBU TERDEKAT MENGGUNAKAN ARDUINO BERBASIS ANDROID,” 2018.
- [16] M. A. Ajam, *Project Management beyond Waterfall and Agile*. Auerbach Publications, 2018. doi: 10.1201/9781315202075.
- [17] R. Habibi Sukarna, Y. Ansori, H. Herfandi, and Y. Yuliadi, “SISTEM PENDUKUNG KEPUTUSAN PEMILIHAN MITRA FREIGHT FORWARDER UNTUK KEGIATAN EKSPOR MENGGUNAKAN METODE ANALYTICAL HIERARCHY PROCESS (AHP) DAN SIMPLE ADDITIVE WEIGHTING (SAW),” 2021.
- [18] L. L. Ellingson and P. Sotirin, *Making Data in Qualitative Research*. Routledge, 2020. doi: 10.4324/9780429058240.
- [19] S. Sundaramoorthy, *UML Diagramming*. Boca Raton: Auerbach Publications, 2022. doi: 10.1201/9781003287124.
- [20] D. Paper, *Web Programming for Business*. Routledge, 2015. doi: 10.4324/9780203582084.
- [21] S. bin Uzayr, *Mastering MySQL for the Web*. Boca Raton: CRC Press, 2022. doi: 10.1201/9781003229629.
- [22] P. C. Jorgensen and B. DeVries, *Software Testing*. Boca Raton: Auerbach Publications, 2021. doi: 10.1201/9781003168447.