

Perancangan Antarmuka Aplikasi Klasifikasi Suara Kemacetan

Interface Design for a Congestion Sound Classification Application

Frans Dharma Hadi Wijaya¹⁾, Agus Budi Dharmawan^{1)*}, Manatap Dolok Lauro Sitorus¹⁾

¹⁾Program Studi Computer Science, Universitas Tarumanegara

Diajukan 8 November 2025 / Disetujui 30 Maret 2026

Abstrak

Kemacetan lalu lintas merupakan masalah krusial di kota-kota besar Indonesia yang memerlukan sistem pemantauan yang efektif dan efisien. Banyak penelitian deep learning berfokus pada pengembangan model klasifikasi suara yang akurat, namun seringkali berhenti pada tahap validasi tanpa bukti konsep fungsional. Hal ini menciptakan kesenjangan antara potensi teknologi dan manfaatnya bagi masyarakat, di mana sistem yang ada belum dapat diakses dan digunakan secara luas. Penelitian ini bertujuan untuk merancang antarmuka aplikasi klasifikasi suara kemacetan yang intuitif dan mudah digunakan oleh masyarakat umum. Metode perancangan yang digunakan adalah *software development life cycle* (SDLC) dengan model *waterfall*. Proses ini mencakup tahapan analisis kebutuhan, desain, implementasi, dan pengujian. Proses desain awal antarmuka (UI/UX) dilakukan menggunakan Figma untuk membuat blueprint visual. Selanjutnya, implementasi antarmuka dikembangkan menjadi aplikasi web fungsional menggunakan *framework* Streamlit dengan bahasa pemrograman Python, yang dipilih karena kemampuannya dalam membuat prototipe berbasis data secara cepat. Hasil dari perancangan ini adalah sebuah prototipe aplikasi fungsional yang dikembangkan menggunakan Streamlit, yang berhasil merealisasikan semua fitur utama yang telah direncanakan, termasuk halaman home, about, karakteristik kelas, dan halaman klasifikasi interaktif. Fungsionalitas inti sistem, seperti unggah audio dan penampilan hasil prediksi, telah diuji dan berjalan sesuai rencana. Antarmuka yang dihasilkan ini berhasil menyajikan fungsionalitas klasifikasi suara yang kompleks dalam format yang mudah diakses. Penelitian ini sukses menghasilkan bukti konsep (*proof of concept*) yang menunjukkan kelayakan sistem pemantauan lalu lintas berbasis audio sebagai salah satu teknologi pendukung *smart city*.

Kata Kunci: Antarmuka, Klasifikasi Suara, Kemacetan, Streamlit, Bukti Konsep

Abstract

Traffic congestion is a crucial problem in major Indonesian cities, requiring effective and efficient monitoring systems. Much deep learning research focuses on developing accurate sound classification models but often stops at validation without a functional proof of concept. This creates a gap between technological potential and public benefit, as existing systems are not widely accessible. This research aims to design an intuitive and user-friendly interface for a congestion sound classification application. The design method used is the software development life cycle (SDLC) with a waterfall model. This process includes stages of requirements analysis, design, implementation, and testing. The initial interface (UI/UX) design process was conducted using Figma to create a visual blueprint. Subsequently, the interface implementation was developed into a functional web application using the Streamlit framework with the Python programming language, chosen for its ability to rapidly create data-driven prototypes. The result of this design is a functional application prototype developed using Streamlit, which successfully realizes all planned main features, including a home page, about page, class characteristics page, and an interactive classification page. The core functionalities of the system, such as audio uploading and prediction result display, have been tested and function as planned. This resulting interface successfully presents complex sound classification functionality in an accessible format. This research successfully produced a proof of concept demonstrating the feasibility of an audio-based traffic monitoring system as a supporting smart city technology.

Keywords: Interface, Sound Classification, Congestion, Streamlit, Proof of Concept

*Korespondensi Penulis:
E-mail: komebik@gmail.com

Pendahuluan

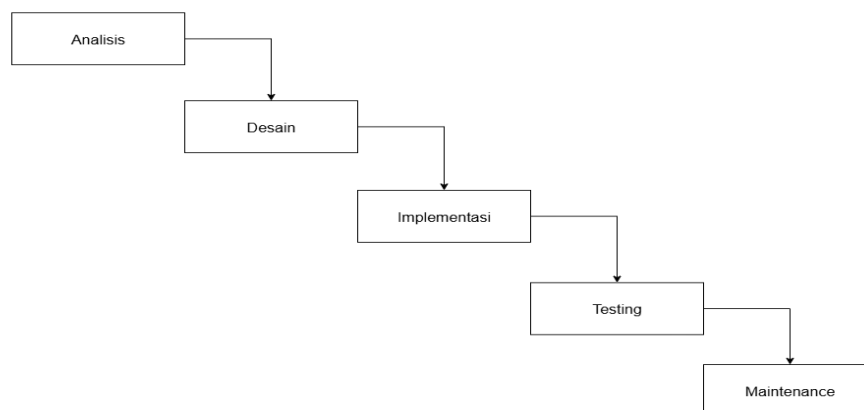
Urbanisasi yang pesat di Indonesia telah menimbulkan berbagai tantangan kompleks di kota-kota besar, dengan kemacetan lalu lintas menjadi salah satu isu utama (Pangestu et al., 2024). Dampak negatif dari kemacetan tidak hanya terasa pada aspek ekonomi, tetapi juga pada kualitas lingkungan melalui polusi suara. Fenomena ini mendorong pengembangan konsep kota pintar (*smart city*), di mana sistem pemantauan yang otomatis dan responsif menjadi komponen vital. Dalam konteks ini, suara lingkungan perkotaan merupakan sumber data akustik yang sangat kaya informasi untuk manajemen lalu lintas (Hisyam et al., 2024).

Banyak penelitian telah membuktikan bahwa arsitektur deep learning seperti *Convolutional Recurrent Neural Network* (CRNN) sangat efektif untuk tugas klasifikasi suara (Amogi et al., 2024). Dengan menggabungkan kemampuan ekstraksi fitur spasial dari CNN dan analisis sekuensial dari LSTM, model ini mampu mengatasi keterbatasan yang ada pada arsitektur CNN murni (Santoso, 2021; Wahyuningtyas, 2021). Namun, keberhasilan sebuah model yang akurat seringkali berhenti pada tahap validasi teknis yang berfokus pada metrik sempit dan tidak diimplementasikan menjadi sebuah aplikasi yang fungsional (Arab et al., 2025; Hutchinson et al., 2022). Kesenjangan antara validasi di lingkungan terkontrol dan implementasi di dunia nyata inilah yang menjadi masalah utama secanggih apapun sebuah model, ia tidak akan memberikan manfaat jika tidak dapat diakses dan digunakan oleh pengguna akhir.

Oleh karena itu, penelitian ini berfokus pada aspek yang menjembatani teknologi kompleks tersebut dengan penggunaannya di dunia nyata. Sebuah model klasifikasi yang akurat saja tidak cukup; agar dapat digunakan dengan baik, sistem tersebut harus dibalut dengan antarmuka yang menarik dan mudah digunakan. Maka dari itu, tujuan dari artikel ini adalah untuk merancang dan mengimplementasikan antarmuka aplikasi klasifikasi suara kemacetan yang dapat diterima oleh masyarakat umum, yang berfungsi sebagai bukti konsep (*proof of concept*) dari kelayakan teknis sistem pemantauan lalu lintas berbasis audio.

Metode Penelitian

Perancangan antarmuka aplikasi ini menggunakan metode *software development life cycle* (SDLC) dengan model *waterfall* (Al Fajar et al., 2022; Anwar et al., 2022). Model *Waterfall* adalah sebuah metode pengembangan perangkat lunak yang prosesnya berjalan satu arah secara berurutan, mirip seperti air terjun. Setiap tahapan harus diselesaikan terlebih dahulu sebelum bisa melanjutkan ke tahapan berikutnya (Saravanos et al., 2023).

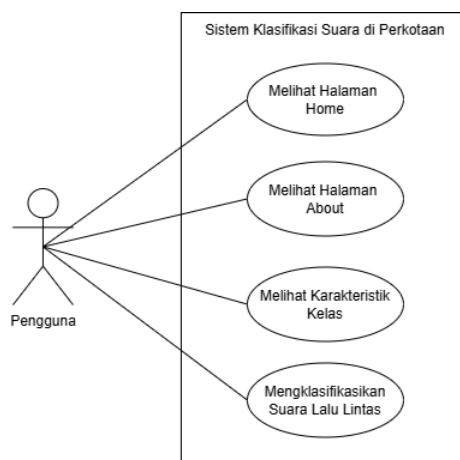


Gambar 1. Alur Metode *Waterfall*

Desain awal dari antarmuka dibuat menggunakan perangkat lunak Figma, sedangkan untuk pengembangan antarmuka dilakukan menggunakan *framework* Streamlit dengan bahasa pemrograman Python. Tahapan model *waterfall* yang diterapkan untuk mengembangkan antarmuka ini dapat dilihat di Gambar 1.

Hasil Dan Pembahasan

Tahap perancangan sistem menghasilkan spesifikasi fungsional dan desain antarmuka yang kemudian direalisasikan menjadi sebuah prototipe aplikasi *web*. Pertama, dirancang terlebih dahulu use case diagram untuk menggambarkan kebutuhan fungsional dari perspektif pengguna. *Use case diagram* adalah diagram yang digunakan untuk memodelkan fungsionalitas sistem dengan mendefinisikan interaksi antara pengguna (*actor*) dengan fitur-fitur utama (*use case*) yang disediakan (Vranić et al., 2024). Gambar 2 memperlihatkan desain *use case diagram* yang merincikan interaksi pengguna dengan antarmuka aplikasi.



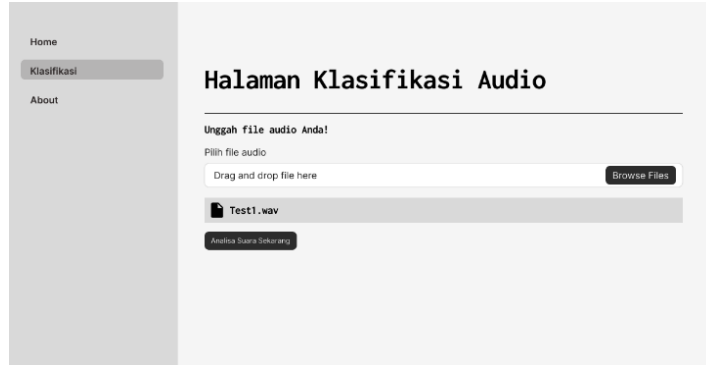
Gambar 2. *Use Case Diagram*

Setelah kebutuhan fungsional didefinisikan, langkah selanjutnya adalah merancang tampilan visual antarmuka (UI) menggunakan perangkat lunak Figma. Rancangan awal ini berfungsi sebagai cetak biru (*blueprint*) dan mencakup empat halaman utama yaitu Halaman Awal (*Home*), Halaman Klasifikasi, Halaman Karakteristik Kelas, dan Halaman *About*. Gambar 3 memperlihatkan rancangan untuk Halaman *Home*, yang berfungsi sebagai halaman penyambut saat pengguna pertama kali mengakses aplikasi. Halaman ini dirancang untuk memberikan deskripsi singkat serta panduan navigasi ke fitur-fitur utama lainnya.



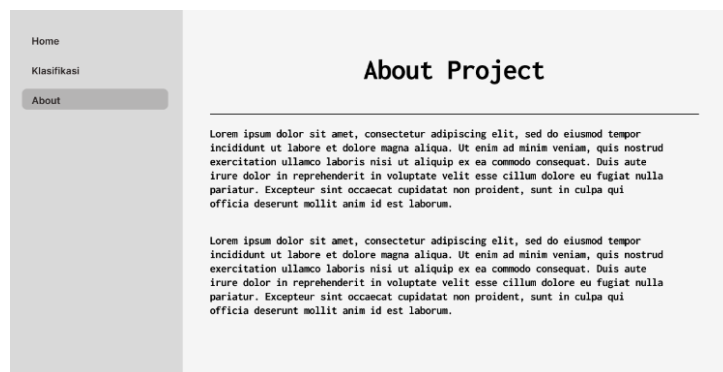
Gambar 3. Desain Halaman *Home*

Selanjutnya, Gambar 4 menunjukkan rancangan untuk Halaman Klasifikasi, yang merupakan fitur inti dari sistem. Halaman ini dirancang dengan antarmuka yang bersih dan minimalis untuk memungkinkan pengguna mengunggah file audio dengan mudah, yang menjadi pusat interaksi dinamis antara pengguna dan model.



Gambar 4. Desain Halaman Klasifikasi

Terakhir, Gambar 5 menampilkan rancangan untuk Halaman *About*. Halaman ini dirancang untuk memberikan konteks dan informasi tambahan kepada pengguna, berisi penjelasan mengenai latar belakang masalah, arsitektur model CRNN yang digunakan, serta detail *dataset* yang dipakai dalam pengembangan proyek ini.



Gambar 5. Desain Halaman About

Terkait desain awal yang telah dirancang, terdapat pengembangan dan penyesuaian pada tahap realisasinya. Hasil implementasi akhir dari antarmuka aplikasi ini terdiri dari empat halaman utama yang fungsional: Halaman Awal (*Home*) yang dapat dilihat pada Gambar 6,

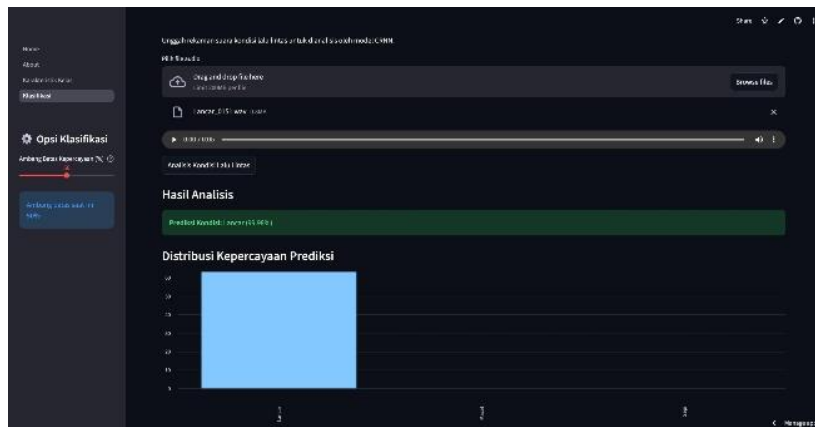


Gambar 6. Realisasi Halaman *Home*

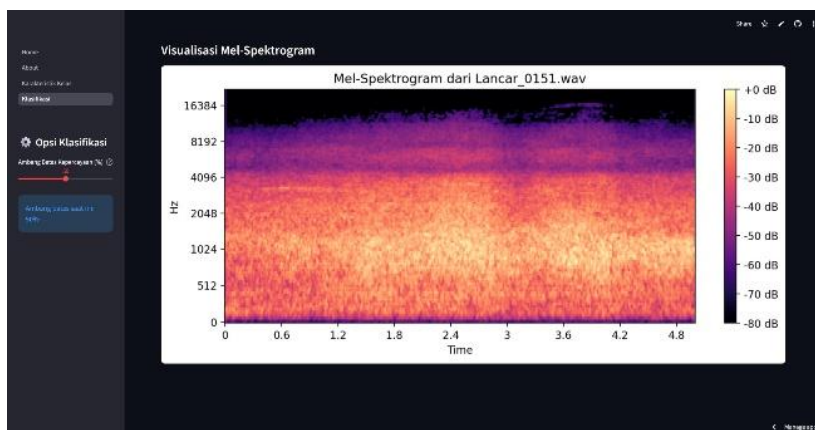
Halaman *About* pada Gambar 7, serta Halaman Klasifikasi yang menjadi fitur inti, yang dapat dilihat pada Gambar 8 dan Gambar 9.



Gambar 7. Realisasi Halaman *About*



Gambar 8. Realisasi Halaman Klasifikasi



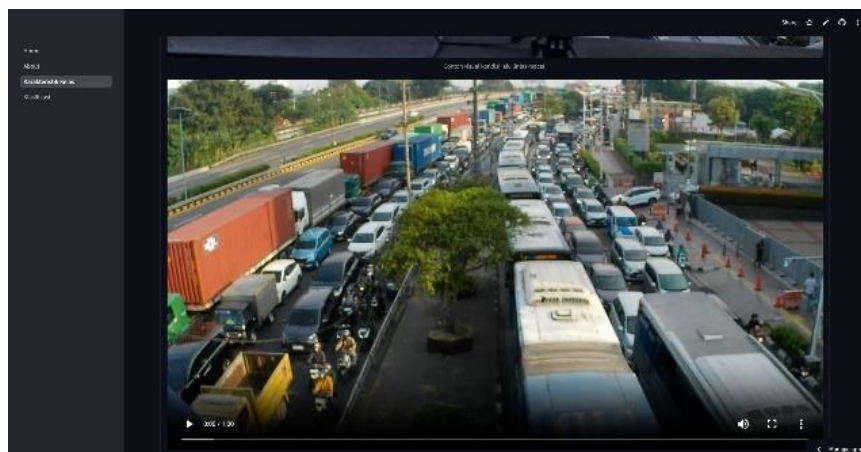
Gambar 9. Realisasi Halaman Klasifikasi

Perbedaan utama dari desain awal adalah adanya penambahan Halaman Karakteristik Kelas. Halaman ini ditambahkan selama proses pengembangan untuk memberikan konteks yang lebih kaya kepada pengguna. Untuk tampilannya dapat dilihat pada Gambar 10. Tujuannya adalah untuk menampilkan dokumentasi video untuk setiap kategori suara (Macet, Lancar, dan Sepi) seperti pada Gambar 11 dan contoh-contoh visual (*Mel-spectrogram*) seperti pada Gambar 12, sehingga pengguna

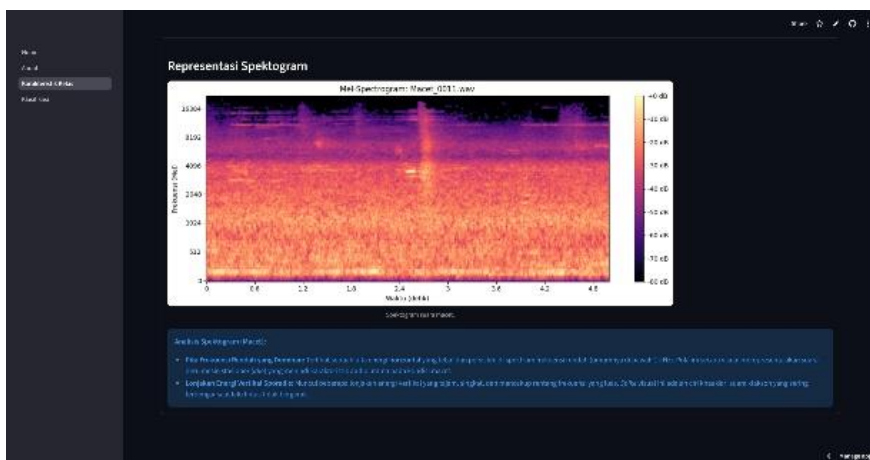
dapat memahami karakteristik data yang digunakan oleh model untuk melakukan klasifikasi. Halaman Klasifikasi tetap menjadi fitur inti pada rancangan, di mana pengguna dapat berinteraksi langsung dengan model untuk melakukan analisis suara, sesuai dengan tujuan utama aplikasi.



Gambar 10. Realisasi Halaman Karakteristik Kelas



Gambar 11. Realisasi Halaman Karakteristik Kelas



Gambar 12. Realisasi Halaman Karakteristik Kelas

Simpulan

Penelitian ini telah berhasil merancang dan merealisasikan antarmuka untuk aplikasi klasifikasi suara kemacetan dengan tujuan utama agar dapat diterima dan digunakan oleh masyarakat umum. Proses perancangan antarmuka ini menerapkan metode *software development life cycle* (SDLC) dengan model *waterfall*, yang memastikan pengembangan sistem berjalan secara terstruktur mulai dari analisis kebutuhan hingga pengujian. Tahapan perancangan desain awal dilakukan menggunakan perangkat lunak Figma, sementara implementasinya diwujudkan menjadi sebuah prototipe aplikasi web fungsional menggunakan *framework* Streamlit dengan bahasa pemrograman Python.

Desain awal antarmuka yang direncanakan, yang terdiri dari Halaman Awal (*Home*), Halaman Klasifikasi, Halaman Karakteristik Kelas, dan Halaman *About*, berhasil direalisasikan menjadi sebuah prototipe yang fungsional dan intuitif. Selama proses pengembangan, dilakukan penambahan halaman Karakteristik Kelas untuk memperkaya pemahaman pengguna terhadap data yang diolah. Fungsionalitas inti dari sistem, yaitu proses unggah file audio oleh pengguna dan penampilan hasil prediksi kondisi lalu lintas, telah diuji secara fungsional dan terbukti berjalan sesuai dengan tujuan perancangan.

Antarmuka yang telah direalisasikan ini berhasil menyajikan fungsionalitas klasifikasi suara yang kompleks dalam format yang mudah diakses dan dipahami. Dengan demikian, penelitian ini berhasil menghasilkan sebuah bukti konsep (*proof of concept*) yang kuat, yang menunjukkan bahwa teknologi klasifikasi suara dapat diimplementasikan menjadi sebuah alat yang praktis dan bermanfaat, sekaligus menjadi fondasi bagi pengembangan sistem pemantauan lalu lintas yang lebih skalabel untuk mendukung inisiatif *smart city*.

Daftar Pustaka

- Al Fajar, M., Dar, M. H., & Rohani, R. (2022). Application of Waterfall Model in Development of Family Planning Participants Information System. *Sinkron*, 7(2), 679–686. doi: 10.33395/sinkron.v7i2.11387
- Amogi, B. R., Ranjan, R., & Khot, L. R. (2024). Mask R-CNN Aided Fruit Surface Temperature Monitoring Algorithm with Edge Compute enabled Internet of Things System for Automated Apple Heat Stress Management. *Information Processing in Agriculture*, 11(4), 603–611. doi: 10.1016/j.inpa.2023.12.001
- Anwar, S., Bagaskara, A., Siahaan, F. B., & Handono, F. W. (2022). Perancangan Sistem Informasi Pengajuan Cuti Karyawan (SIMCAR) Berbasis Web. In *Jurnal INSAN: Journal of Information System Management Innovation* (Vol. 2, Issue 2, pp. 69–76). doi: 10.31294/jinsan.v2i2.1676
- Arab, R. A. El, Abu-Mahfouz, M. S., Abuadas, F. H., Alzghoul, H., Almari, M., Ghannam, A., & Seweid, M. M. (2025). Bridging the Gap: From AI Success in Clinical Trials to Real-World Healthcare Implementation—A Narrative Review. *Healthcare*, 13(7). doi: 10.3390/healthcare13070701
- Hisyam, M. I., & Prasetio, B. H. (2024). Klasifikasi Suara Sirene Kendaraan Berbasis MFCC untuk Meningkatkan Efisiensi Sistem Keamanan Lalu Lintas. *Pengembangan Teknologi Informasi Dan Ilmu Komputer (J-PTIHK)*, 8(7). Retrieved from <https://j-ptiik.ub.ac.id/index.php/j-ptiik/article/view/13947>
- Hutchinson, B., Rostamzadeh, N., Greer, C., Heller, K., & Prabhakaran, V. (2022). Evaluation Gaps in Machine Learning Practice. *ACM International Conference Proceeding Series*, 1859–1876. doi: 10.1145/3531146.3533233

- Pangestu, M. I., Ginting, J. A., Suryantara, I. G. N., & Marvelino, M. (2024). Optimasi Pertanian Di Bekasi Utara: Prediksi Curah Hujan Dan Rekomendasi Tanaman Dengan Menggunakan Model Regresi Linier. *Jurnal Algoritma, Logika Dan Komputasi*, 7(02), 731–742. doi: 10.30813/j-alu.v2i2.8081
- Santoso, J. T. (2021). *Kecerdasan Buatan dan Jaringan Syaraf Buatan*. Yayasan Prima Agus Teknik. Retrieved from *Kecerdasan Buatan dan Jaringan Syaraf Buatan*
- Saravanos, A., & Curinga, M. X. (2023). Simulating the Software Development Lifecycle: The Waterfall Model. *Applied System Innovation*, 6(6). doi: 10.3390/asi6060108
- Vranić, V., Lang, J., Nores, M. L., Arias, J. J. P., Solano, J., & Laseca, G. (2024). Use Case Modeling in a Research Setting of Developing an Innovative Pilgrimage Support System. *Universal Access in the Information Society*, 23(4), 1543–1560. doi: 10.1007/s10209-023-01047-1
- Wahyuningtyas, V. (2021). Implementasi Ekstraksi Fitur untuk Klasifikasi Suara Urban Menggunakan Deep Learning. *Sains, Aplikasi, Komputasi Dan Teknologi Informasi*, 3(1). doi: 10.30872/jsakti.v3i1.4442