

## PENGEMBANGAN *PROTOTYPE* ALAT KLASIFIKASI KUALITAS LAMPU DENGAN *LONG SHORT-TERM MEMORY* BERBASIS IoT PADA PT. XYZ

Rizki Ramadhani<sup>1</sup>, Susanto Hariyanto<sup>2\*</sup>

<sup>1,2</sup> Program Studi Teknik Informatika, Fakultas Sains & Teknologi, Universitas Buddhi Dharma

\*Corresponding Author, email: [susanto.hariyanto@ubd.ac.id](mailto:susanto.hariyanto@ubd.ac.id)

### ABSTRAK

Proses pengujian kualitas lampu LED di PT. XYZ yang selama ini dilakukan secara manual cenderung menghasilkan penilaian yang subjektif, tidak konsisten, dan sulit dianalisis secara kuantitatif. Untuk mengatasi permasalahan tersebut, dikembangkan sebuah prototipe sistem klasifikasi kualitas lampu berbasis *Internet of Things* (IoT) dengan algoritma *Long Short-Term Memory* (LSTM). Sistem ini menggunakan sensor BH1750FVI untuk mengukur intensitas cahaya, sensor PZEM-004T untuk memantau daya dan tegangan, serta mikrokontroler ESP32 untuk pengumpulan dan pengiriman data selama sesi pengujian berdurasi 15 menit. Data *time-series* dari 100 sesi (50 layak, 50 *defect*) digunakan sebagai dataset untuk pelatihan model LSTM. Evaluasi model dilakukan melalui *validation testing* menggunakan berbagai metrik yang dihitung dari *confusion matrix*, menghasilkan akurasi sebesar 0,95, presisi 0,90, *recall* 1,00, dan *F1-score* 0,947. Sistem ini juga secara otomatis mengidentifikasi lampu yang mati berdasarkan pola pembacaan lux yang bernilai nol secara berurutan. Prototipe juga dilengkapi dengan *dashboard* berbasis web untuk memantau, memulai, membatalkan, dan melihat hasil klasifikasi sesi pengujian. Pengujian sistem dilakukan melalui *functional testing* dengan pendekatan *blackbox testing*, untuk memastikan seluruh fungsi berjalan sesuai kebutuhan, termasuk validasi input, pengelolaan sesi, pengiriman data, dan tampilan data sensor melalui *dashboard*. Selain itu, *performance testing* menunjukkan bahwa sistem mampu mengirimkan 450 hingga 750 baris data sensor per sesi ke database. Meskipun terjadi kehilangan sebagian data akibat *noise* dari ESP32 dan gangguan jaringan, proses klasifikasi tetap berjalan dengan akurat dan andal. Dengan demikian, proses pengujian kualitas lampu dapat diotomatisasi melalui sistem yang dikembangkan, dan memiliki potensi untuk menggantikan metode pengujian manual yang telah diterapkan sebelumnya.

**Kata kunci:** *Deep Learning*, ESP32, IoT, LSTM.

### I. PENDAHULUAN

Dalam dunia bisnis, perusahaan tidak hanya berorientasi pada keuntungan, tetapi juga berupaya memenuhi kebutuhan dan kepuasan pelanggan. Kepuasan tersebut berdampak pada peningkatan profit dan loyalitas konsumen (Supriyadi et al., 2021). Oleh karena itu, menjaga kualitas produk menjadi hal yang sangat penting dalam mempertahankan kepercayaan dan daya saing, serta mencegah distribusi produk cacat yang dapat menimbulkan keluhan, biaya tambahan, dan penurunan reputasi.

Kontrol kualitas menjadi kunci dalam menjamin standar produk. Namun, banyak industri masih mengandalkan pemeriksaan manual yang cenderung tidak akurat dan inkonsisten. Pemanfaatan teknologi seperti sistem informasi dan otomasi menjadi solusi efektif karena memungkinkan pengujian dilakukan secara objektif, menghasilkan data kuantitatif, dan mendukung pemantauan kualitas secara *real-time* (Supriyadi et al., 2021). Hal ini juga mempermudah deteksi dini terhadap masalah serta menjaga efisiensi produksi (Kumar et al., 2025). Dukungan teknologi *Internet of Things* (IoT) memungkinkan perangkat saling terhubung dan bertukar data otomatis, sehingga meningkatkan integrasi sistem (Wenando, 2023). Seiring berkembangnya industri, *machine learning* banyak dimanfaatkan untuk otomatisasi dan pengambilan keputusan berbasis data (Purmala, 2021). Didukung oleh sensor dan IoT, data *time-series* kini dapat dikumpulkan secara otomatis dari proses produksi. Data ini kemudian dianalisis menggunakan *deep learning* yang unggul dalam mengekstraksi fitur dan mengklasifikasikan data *time-series* kompleks (Farahani et al., 2024). Berbagai arsitektur seperti LSTM, GRU, CNN, dan *attention mechanism* telah diterapkan luas pada pemrosesan data multivariat berbasis IoT (Papastefanopoulos et al., 2023). *Deep learning* dinilai efektif dalam mengenali pola kompleks dan mengekstraksi informasi penting dari deret waktu secara akurat (Qiao & Zhou, 2023).

Penelitian ini bertujuan untuk merancang sistem pengujian kualitas lampu berbasis mikrokontroler yang dapat mengukur parameter kelayakan dan mengklasifikasikan kualitas berdasarkan data sensor menggunakan metode *deep learning*. Sistem ini dirancang untuk menghasilkan data kuantitatif yang mendukung analisis dan pengambilan keputusan, serta membantu mengurangi risiko distribusi produk cacat. Manfaat sistem mencakup mengurangi ketergantungan pada evaluasi manual, mempermudah proses dokumentasi dan analisis kualitas lampu melalui penyimpanan data hasil pengujian, dan mendukung upaya perusahaan untuk meningkatkan daya saing dengan menyediakan produk berkualitas tinggi yang telah melalui proses pengujian yang andal dan konsisten. Ruang lingkup penelitian ini difokuskan pada pengembangan prototipe pengujian otomatis untuk lampu LED X 18W. Penelitian tidak mencakup aspek teknis seperti daya tahan komponen, suhu operasional, atau efisiensi komunikasi data. Fokus

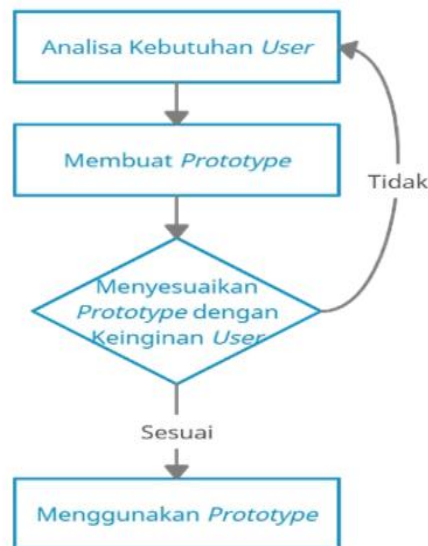
diarahkan pada perancangan alur pengujian dan evaluasi performa model klasifikasi berdasarkan data sensor, yang ditujukan untuk penggunaan di lingkungan pengujian PT. XYZ.

## II. METODOLOGI

### 2.1 Metode Pengumpulan Data

Data dalam penelitian ini diperoleh melalui pengujian langsung terhadap sepuluh unit lampu LED dari PT. XYZ. Lampu telah diklasifikasikan oleh pihak perusahaan ke dalam dua kategori, yaitu lima layak dan lima *defect*. Untuk mendapatkan dataset yang cukup dan bervariasi, setiap lampu diuji secara berulang dalam beberapa sesi menggunakan alat prototipe IoT yang telah dirancang khusus. Hasilnya, diperoleh 100 dataset, terdiri dari 50 data lampu layak dan 50 *defect*. Setiap dataset mewakili satu sesi pengujian yang dilakukan selama 15 menit dan memuat lima fitur utama: *lamp\_type*, *lamp\_wattage*, *voltage*, *power*, dan *lux*. Data diekspor dalam format CSV untuk pelatihan model.

### 2.2 Metode Pengembangan Sistem



Gambar 1. Tahapan *Evolutionary Prototype*

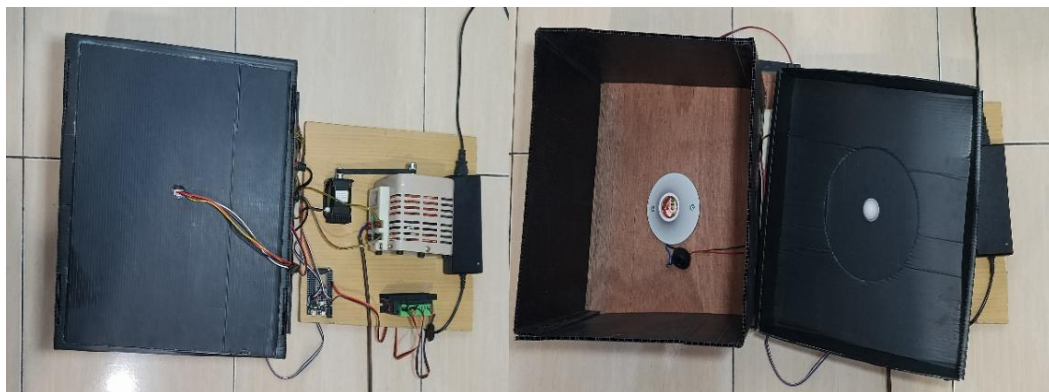
Pengembangan sistem dilakukan dengan pendekatan *evolutionary prototyping*, yaitu metode pengembangan di mana prototipe awal dibangun terlebih dahulu, kemudian disempurnakan secara bertahap berdasarkan evaluasi pengguna

atau hasil uji fungsional (Firmansyah, 2021). Tahapan dimulai dari analisis kebutuhan, pembangunan prototipe, evaluasi, hingga penyempurnaan berkelanjutan sampai diperoleh versi akhir yang siap digunakan. Dalam penelitian ini, pendekatan tersebut digunakan untuk mengembangkan sistem secara bertahap, dimulai dari analisis kebutuhan, pembangunan prototipe awal, pengujian fungsional, hingga penyempurnaan berdasarkan hasil evaluasi. Sistem yang dikembangkan terdiri dari dua bagian utama, yaitu perangkat keras dan perangkat lunak.

### 2.3 Metode Pengujian

Pengujian sistem dilakukan melalui *functional testing*, *performance testing*, dan *validation testing*. *Functional testing* untuk memastikan setiap komponen berjalan sesuai fungsinya, seperti pembacaan sensor, pengiriman data, klasifikasi, dan tampilan dashboard, dengan pendekatan *blackbox*. *Performance testing* untuk menguji kestabilan sistem dan mengamati kemungkinan kehilangan data. *Validation testing* untuk mengevaluasi akurasi model LSTM dengan membandingkan hasil klasifikasi terhadap kondisi aktual menggunakan *confusion matrix* dan metrik evaluasi seperti akurasi, presisi, *recall*, dan *F1-score*.

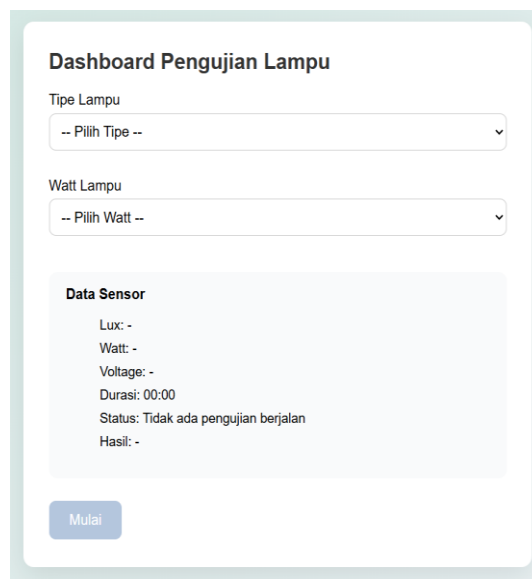
## III. HASIL DAN PEMBAHASAN



Gambar 2. Alat IoT

Perangkat keras terdiri dari NodeMCU ESP32 Doit V1 yang kuat dan hemat biaya untuk pengembangan IoT (Hercog et al., 2023). Mikrokontroler ini dirancang dengan efisiensi daya tinggi serta mendukung komunikasi nirkabel dan pengolahan data, sehingga banyak digunakan sebagai pusat kendali dalam perangkat IoT

(Syamsuri et al., 2022). Komponen lain meliputi *expansion board* dan kabel *jumper* sebagai konektor, sensor PZEM-004T untuk membaca tegangan dengan *current transformer* (CT) untuk membaca daya, serta sensor BH1750FVI untuk mengukur intensitas cahaya. *Slide regulator* AC 500W digunakan untuk mengatur tegangan input lampu, yang dikendalikan oleh *stepper motor* Nema 17 melalui *driver* TB6600 dan dihubungkan menggunakan *timing belt*. Tegangan diatur secara bertahap selama pengujian ke 110V, 220V, dan 250V sesuai prosedur perusahaan. Boks uji dengan fitting E27 dirancang untuk mencegah cahaya luar masuk, sehingga hanya cahaya dari lampu yang terbaca oleh sensor.



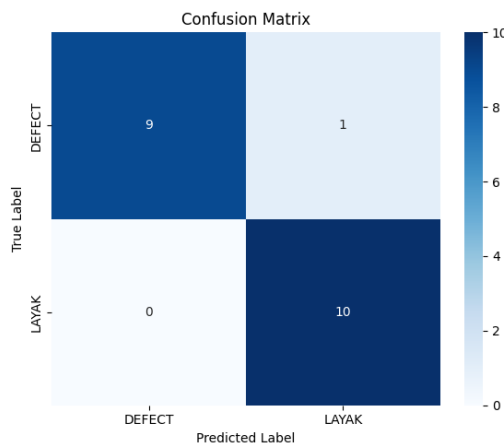
Gambar 3. Dashboard Pengujian

Perangkat lunak terdiri dari *dashboard* web berbasis PHP-MySQL dan model klasifikasi menggunakan algoritma *Long Short-Term Memory* (LSTM). Komunikasi antara ESP32 dan server dilakukan melalui *backend* API berbasis REST menggunakan protokol HTTP dan format JSON, memungkinkan pertukaran data secara terstruktur dan *real-time* (Akmal & Dasaprawira, 2022). Sistem juga mampu mendeteksi lampu mati secara otomatis melalui pembacaan lux nol secara berurutan. Jika kondisi tersebut terpenuhi, pengujian dihentikan lebih awal dan sistem langsung menyimpan hasil klasifikasi sebagai “Mati”.

LSTM merupakan arsitektur *Recurrent Neural Network* (RNN) yang dirancang untuk mengingat informasi dalam jangka panjang (Muhammad & Irawan, 2023), memproses data *time-series* secara efektif, serta mengatasi masalah

*vanishing gradient* (Alshingiti et al., 2023). Model ini memiliki tiga gerbang utama: *input gate*, *forget gate*, dan *output gate* yang masing-masing berfungsi untuk menyimpan, menghapus, dan meneruskan informasi (Susetyo et al., 2025). *Input gate* menentukan informasi yang relevan untuk disimpan, *forget gate* membuang data yang tidak diperlukan, dan *output gate* meneruskan informasi ke tahap berikutnya (Hanifa et al., 2021).

Data sensor dinormalisasi menggunakan *MinMaxScaler* dan diproses dengan *padding* agar panjang input seragam. Data kemudian dibagi menjadi 80% untuk pelatihan dan 20% untuk pengujian. Pelatihan dilakukan menggunakan *Keras Tuner* dengan metode *Random Search* untuk menemukan kombinasi *hyperparameter* terbaik, seperti jumlah unit LSTM, *dropout*, dan *learning rate*. Proses tuning berlangsung selama 20 epoch dengan *batch size* 4.



**Gambar 4. Confusion Matrix**

Model LSTM terdiri dari lapisan *masking*, LSTM, *dropout*, dan *dense* dengan aktivasi sigmoid. Hasil tuning menghasilkan konfigurasi terbaik dengan 64 unit LSTM, *dropout* 0.2, dan *learning rate* 0.001. Evaluasi dilakukan melalui *validation testing*, dengan *confusion matrix* menunjukkan 10 *True Positive*, 9 *True Negative*, 1 *False Positive*, dan 0 *False Negative*, menghasilkan akurasi 0.95, presisi 0.90, *recall* 1.00, dan *F1-score* 0.947. Hasil ini menunjukkan bahwa model mampu mengklasifikasikan lampu layak dan *defect* dengan sangat baik. *Functional testing* dengan pendekatan *blackbox* menunjukkan seluruh fungsi utama sistem, mulai dari pembacaan sensor hingga tampilan hasil klasifikasi, berjalan sesuai rancangan. Pada *performance testing*, sistem seharusnya dapat mengirimkan hingga 900 baris

data sensor dalam satu sesi berdurasi 15 menit, namun jumlah yang berhasil dikirimkan hanya berkisar antara 450–750 baris karena terpengaruh oleh kualitas jaringan dan noise. Meskipun demikian, proses klasifikasi tetap berjalan secara akurat dan stabil.

#### IV. SIMPULAN

Penelitian ini menghasilkan prototipe sistem klasifikasi kualitas lampu berbasis IoT dan *Long Short-Term Memory* (LSTM) yang mampu mengotomatisasi proses pengujian lampu LED di PT. XYZ. Model menunjukkan performa tinggi, dengan akurasi 95%, presisi 90%, *recall* 100%, dan *F1-score* 94,7% serta mampu mendeteksi lampu mati secara otomatis tanpa menggunakan model. *Functional* dan *performance testing* menunjukkan bahwa sistem berfungsi sesuai rancangan dan tetap dapat memproses data, meskipun hanya 450–750 data yang berhasil disimpan dalam satu sesi. Dengan demikian, sistem ini berpotensi menggantikan metode manual dan meningkatkan konsistensi proses *quality control*.

#### DAFTAR PUSTAKA

- Akmal, N. K., & Dasaprawira, M. N. (2022). Rancang Bangun Application Programming Interface (Api) Menggunakan Gaya Arsitektur GraphQL Untuk Pembuatan Sistem Informasi Pendataan Anggota Unit Kegiatan Mahasiswa (Ukm) Studi Kasus Ukm Starlabs. *Jurnal SITECH*, 5(1), 37–40.
- Alshingiti, Z., Alaqel, R., Al-Muhtadi, J., Haq, Q. E. U., Saleem, K., & Faheem, M. H. (2023). A Deep Learning-Based Phishing Detection System Using CNN, LSTM, and LSTM-CNN. *Electronics*, 12(1), 1–18.
- Farahani, M. A., Kalach, F. El, Harik, R., & Wuest, T. (2024). High-resolution time-series classification in smart manufacturing systems. *Manufacturing Letters*, 41, 1170–1181.
- Firmansyah, B. (2021). Sistem Informasi Manajemen Dan Layanan Aset Ti Menggunakan Framework Codeigniter. *Teknimedia*, 2(1), 8–16.
- Hanifa, A., Fauzan, S. A., Hikal, M., & Ashfiya, M. B. (2021). Perbandingan Metode Lstm Dan Gru (Rnn) Untuk Klasifikasi Berita Palsu Berbahasa Indonesia. *Dinamika Rekayasa*, 17(1), 33–39.

- Hercog, D., Lerher, T., Truntič, M., & Težak, O. (2023). Design and Implementation of ESP32-Based IoT Devices. *Sensors*, 23(15), 1–20.
- Kumar, H. N. J., Thamizharasan, N., & Krishna, A. S. (2025). Manufacturing Quality Control Analysis System. *International Research Journal of Modernization in Engineering Technology and Science*, 07(4), 4794–4803.
- Muhammad, T. A. F., & Irawan, M. I. (2023). Implementasi Long Short-Term Memory (LSTM) untuk Prediksi Intensitas Curah Hujan (Studi Kasus: Kabupaten Malang). *JURNAL SAINS DAN SENI ITS*, 12(1), A34–A39.
- Papastefanopoulos, V., Linardatos, P., Panagiotakopoulos, T., & Kotsiantis, S. (2023). Multivariate Time-Series Forecasting: A Review of Deep Learning Methods in Internet of Things Applications to Smart Cities. *Smart Cities*, 6(5), 2519–2552.
- Purmala, Y. A. (2021). Penerapan machine learning dalam meningkatkan produktivitas di industri manufaktur: Tinjauan literatur. *Operations Excellence: Journal of Applied Industrial Engineering*, 13(2), 267–275.
- Qiao, J., & Zhou, E. (2023). LSTM-Based Anomaly Detection in Manufacturing Environmental Monitoring Data. *Automation and Machine Learning*, 4(3), 55–59. <https://doi.org/10.23977/autml.2023.040307>
- Supriyadi, E., Sofiana, M., & Dwipangga, S. (2021). Sistem Kendali Lampu Defect Dan Reject Berbasis Web Server Menggunakan Raspberry Pi 3 Model B. *JURNAL TEKNIK INFORMATIKA STMIK ANTAR BANGSA*, VII(1), 9–15.
- Susetyo, Y. A., Parhusip, H. A., Trihandaru, S., & Susanto, B. (2025). LSTM-IOT (LSTM-based IoT) untuk Mengatasi Kehilangan Data Akibat Kegagalan Koneksi. *Jurnal Teknologi Informasi Dan Ilmu Komputer*, 12(1), 175–186.
- Syamsuri, T. U., Amalia, R. N., Mudjiono, & Imron, A. (2022). Rancang Bangun Alat Monitoring Daya Listrik di Asrama Berbasis Web Menggunakan ESP32. *ELPOSYS: Jurnal Sistem Kelistrikan*, 9(3), 139–145.
- Wenando, F. A. (2023). Peran Penggunaan IoT dengan Machine Learning dalam Penanganan Pandemi COVID-19: Systematic Literatur Review. *JURNAL FASILKOM*, 13(2), 318–325.