Journal Management, Informatics, Software Engineering and Communication Technology
Vol. 5 No. 1, June (2026) pp. 67-73
Published by: Marq & Cha Institute
Journal Management, Informatics, Software Engineering and Communication Technology
Journal homepage: www.ejournal.marqchainstitute.or.id/index.php/jmirte

CNN-Based Identification of Waterlogging-Tolerant Plants for Consumption Safety

Anggi Sri Rezeki Nasution[1], Mulkan Azhari[2]
[1,2] Information Systems, Faculty of Computer Science and Information Technology, Universitas Muhammadiyah Sumatera Utara, Medan, Indonesia

Article Info

Article history

Received : Jun 15, 2026
Revised : Jun 27, 2026
Accepted : Jun 30, 2026

Keywords:

Convolutional Neural Network;
Consumption;
Feasibility.MobileNetV2;
Plant Identification;
Waterlogging-Tolerant Plants.

Abstract

Indonesia's tropical climate creates a high risk of flooding that produces prolonged waterlogging in many regions. Several plant species survived under such conditions and were potentially used as alternative food sources, yet not all were safe for direct consumption, while public knowledge to distinguish safe from toxic species remained limited. This study aimed to design an image-based identification system for waterlogging-tolerant plants using a Convolutional Neural Network (CNN) with the MobileNetV2 architecture to determine consumption feasibility. The model was trained through transfer learning on 1,500 images covering 15 species, split 70:15:15 for training, validation, and testing. The system integrated a Laravel web application with a FastAPI-based Machine Learning Service via REST API. Testing on 225 images yielded an accuracy of 84%, precision of 85%, recall of 84%, and an F1-score of 84%, with each result accompanied by the species' consumption status, characteristics, edible parts, and processing method.

Corresponding Author:

Batista Leonardo P Dasuha,
Information Systems,
Universitas Muhammadiyah Sumatera Utara,
Jl. Kapten Muchtar Basri No.3, Glugur Darat II, Kec. Medan Tim., Kota Medan, Sumatera Utara 20238, Indonesia.
Email : anggisrinst23@gmail.com

This is an open access article under the CC BY-NC license.

1. Introduction

Indonesia is located in a tropical region with high rainfall and dynamic seasonal wind patterns, giving it a high potential for hydrometeorological disasters, particularly floods (National Disaster Management Authority [BNPB], 2024). Recurring floods produce prolonged waterlogging across many regions, which affects plant growth and survival. Waterlogged conditions reduce oxygen availability in the root zone, causing hypoxia that disrupts plant metabolism and reduces productivity (Fahmi & Wakhid, 2018; Pan, Sharif, Xu, & Chen, 2021). Nevertheless, several plant species can survive under such conditions through physiological adaptations, including aerenchyma formation for oxygen distribution, increased glycolytic activity, and genetic regulation of hypoxic stress response (Loreti & Perata, 2020). This adaptive capacity allows waterlogging-tolerant plants to keep growing in flood-affected areas and to be used as alternative food sources (Prabaningrum, Nugroho, & Kaswinarni, 2018).

However, not every plant that grows in waterlogged areas is safe for consumption. Edible plants generally contain beneficial nutrients and are free of toxic compounds that endanger health (Saidi, Azara, Ramadhani, & Yanti, 2022). Limited public knowledge in identifying plant species and determining their consumption feasibility can pose serious health risks. Conventional plant identification typically requires specialized botanical expertise and cannot always be performed quickly, especially during emergencies or when expert access is limited, creating an urgent need for an automatic, fast, and informative identification system, particularly in post-flood emergency conditions where access to trained botanists is limited.

Advances in artificial intelligence, particularly in digital image processing, offer a way to address this problem. Convolutional Neural Network (CNN) is a deep learning model widely used for image classification because it can automatically extract visual features such as shape, texture, color, and pattern through a layered structure consisting of convolution, pooling, and fully connected layers (Alzubaidi et al., 2021). CNN's ability to learn hierarchical feature representations makes it highly effective at recognizing objects with high accuracy (Alzubaidi et al., 2021).

CNN-based plant identification has been widely studied. Abdiel and Yuhri (2026) built a website-based classification system for 10 Indonesian herbal leaf species using MobileNetV2, achieving 92% validation accuracy. Adelia, Fitri, and Agusniar (2025) built a website-based herbal-and-toxic leaf detection system using custom CNN, EfficientNetB0, MobileNetV2, and ResNet50V2, reaching 98.96% validation accuracy. Setiyono et al. (2023) identified 86 Indonesian medicinal plant species from leaf images using EfficientNetV2-S at 98% average accuracy. Butar-Butar and Marpaung (2023) applied MobileNetV2 transfer learning to identify four medicinal leaf species at 94.16% accuracy, while Nasution, Niska, Al Idrus, Indra, and Taufik (2025) achieved 92.73% accuracy in classifying rhizome-type herbal plants using the Xception architecture. CNN-based classification has also been applied to other plant-image tasks, such as TOGA (Indonesian medicinal home garden) plant classification at 80% accuracy (Alamsyah, 2019), leaf-based plant-type identification at 76% accuracy (Felix, Wijaya, Sutra, Kosasih, & Sirait, 2020), early-blight and late-blight tomato-disease classification at 80% accuracy (Ningsih, Suryadi, Bakti, & Imran, 2022), and web-based plant-disease identification (Mulyana et al., 2025). These studies show that CNN methods generally perform well on plant-image classification, while MobileNetV2 specifically offers high computational efficiency suited to web-based systems. However, these studies mostly focused on herbal plants, leaf-disease classification, or general plant-type recognition, and none specifically addressed waterlogging-tolerant plants or presented consumption-safety information to the public.

Based on this gap, the present study designs and builds an image-based identification system for waterlogging-tolerant plants using CNN with the MobileNetV2 architecture to determine plant consumption feasibility, integrating the model's classification results with a consumption-feasibility reference database through a Laravel web application and a FastAPI-based Machine Learning Service. The novelty of this study lies in its research scope: 15 waterlogging-tolerant species grouped by consumption-feasibility status (directly edible, edible after processing, or inedible/toxic), so that the system not only recognizes plant species but also helps the public understand their consumption safety.

2. Research Methodolgy

This study is applied research using a quantitative image-classification experiment approach. The research was conducted through training and testing of a CNN model on a labeled dataset of waterlogging-tolerant plant images, after which the predicted species was linked to a database containing consumption-feasibility status, edible parts, and processing methods.

The dataset consisted of 1,500 images covering 15 waterlogging-tolerant species, each represented by 100 images to maintain class balance. The fifteen species were grouped into three consumption-feasibility categories: (a) inedible/toxic — Jatropha curcas, Cerbera manghas, Eichhornia crassipes, Datura metel, and Derris elliptica; (b) directly edible — Musa paradisiaca, Syzygium aqueum, Pilea trinevia, Lactuca sativa, and Morus alba; and (c) edible after processing — Ipomoea aquatica, Nelumbo nucifera, Artocarpus altilis, Bambusa vulgaris, and Colocasia esculenta. Images were collected from public platforms (Kaggle, iNaturalist, and Pinterest) to obtain diverse visual variation in shape, color, angle, and lighting. The dataset was then proportionally split at a 70:15:15 ratio into training (1,050 images), validation (225 images), and testing (225 images) sets.

Before training, all images were preprocessed through resizing to 224×224 pixels to match the MobileNetV2 input specification, followed by pixel-value normalization using MobileNetV2's built-in preprocess_input function, defined as $x' = (x / 127.5) - 1$, so that pixel values fall within the [-1, 1] range. This preprocessing pipeline is part of the broader digital image-processing workflow commonly applied in image-based plant phenotyping technology (Putra, Cahyani, Marpaung, & Fil'aini, 2020). No data augmentation (e.g., rotation, flipping, zoom, or shifting) was applied, so the amount of training data the model learned from depended entirely on the original collected images.

The classification model was built on the MobileNetV2 architecture using a transfer-learning approach based on feature extraction. MobileNetV2 was selected for its lightweight and computationally efficient architecture, which still delivers strong classification performance (Hariman, Mulyana, & Yel, 2023). The MobileNetV2 weights pretrained on ImageNet were frozen as the feature extractor, while the classification head was redesigned with Global Average Pooling, a Dropout layer at a rate of 0.3, and a Dense layer with a Softmax activation to output probabilities across the 15 plant classes. The transfer-learning approach was chosen because it speeds up training and reduces the risk of overfitting on a limited-size dataset (Sitorus et al., 2025).

Training used an input image size of 224×224 pixels, a batch size of 16, and 30 epochs. Model weights were optimized using the Adam optimizer with a default learning rate of 0.001, while Sparse Categorical Crossentropy was used as the loss function. Validation data was used throughout training to monitor model performance per epoch and detect overfitting, while final performance was measured on the held-out test set.

The system was designed with a four-layer architecture. The Presentation Layer provides the web interface for users and administrators. The Application Layer was implemented in Laravel 12 (Sinlae, Irwanda, Maulana, & Syahputra, 2024), handling input validation, administrator authentication, plant-data management, identification history, and communication with the Processing Layer via REST API. The Processing Layer is a FastAPI-based Machine Learning Service that performs image preprocessing, MobileNetV2 inference, and model training/management. The Data Layer uses a relational MySQL database storing four main entities: Admin, Plant, Identification, and Trained Model. Communication between Laravel and the Machine Learning Service uses REST API endpoints such as /predict for identification, /training/start and /training/status/{task_id} for training and monitoring, and /models and /models/activate for model management, allowing identification and training to run independently while remaining integrated with the web application.

Model performance was evaluated using accuracy, precision, recall, and F1-score, along with a confusion matrix to illustrate the relationship between actual and predicted classes (Ichwan & Sumantri, 2024). F1-score, the harmonic mean of precision and recall, was used to balance model-performance assessment in this multiclass classification task (Bismi, Novianti, & Qomaruddin, 2024). In addition to model evaluation, system testing was performed using Black Box Testing to confirm that all functional features worked as designed, along with API integration testing to confirm that data exchange between the Laravel application and the Machine Learning Service worked according to the intended mechanism.

3. Results And Discussions

Training of the CNN MobileNetV2 model showed a significant increase in training accuracy from the first epoch, reaching approximately 99% by the end of training, while validation accuracy increased over the early epochs and stabilized around 84%–86% (Figure 1). Training loss decreased steadily from around 2.2 toward 0, while validation loss decreased sharply in the early epochs before stabilizing around 0.5 (Figure 2). The gap between training accuracy approaching 99% and validation accuracy stable at 84%–86% indicates mild overfitting, which is reasonable given that the feature-extraction approach only retrains the classification head on a relatively limited dataset without data augmentation.

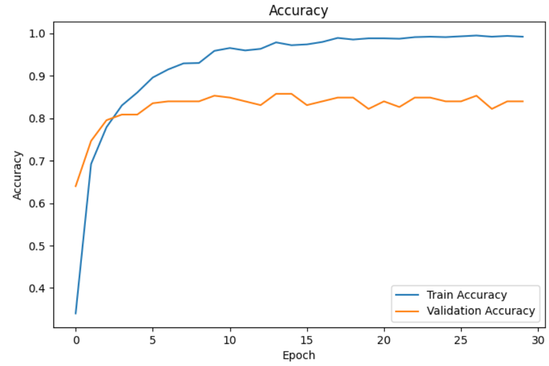

Figure 1. Training Accuracy Graph
Source: Author's analysis.

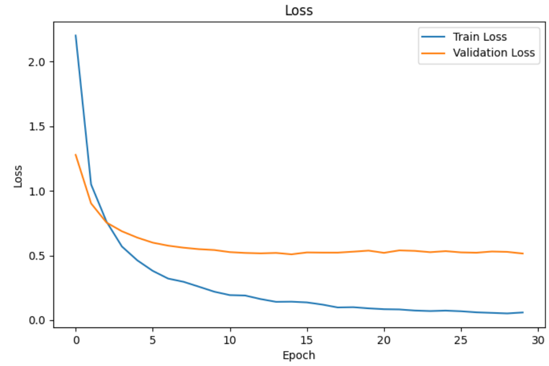

Figure 2. Training Loss Graph
Source: Author's analysis.

Evaluation on the 225-image test set showed the model achieved 84% accuracy, with a macro-average precision of 85%, recall of 84%, and F1-score of 84% (Table 1). This indicates that model performance was relatively consistent across all 15 plant classes, with no single class dominating the performance imbalance, given the balanced test distribution of 15 images per species.

Table 1. Classification Report of the CNN MobileNetV2 Model

Species	Precision	Recall	F1-Score	Support
Cerbera manghas	0.80	0.53	0.64	15
Eichhornia crassipes	0.87	0.87	0.87	15
Syzygium aqueum	0.91	0.67	0.77	15
Jatropha curcas	0.79	1.00	0.88	15
Ipomoea aquatica	0.91	0.67	0.77	15
Datura metel	0.74	0.93	0.82	15
Morus alba	1.00	1.00	1.00	15
Musa paradisiaca	1.00	1.00	1.00	15
Pilea trinevia	0.76	0.87	0.81	15
Bambusa vulgaris	0.88	0.93	0.90	15
Lactuca sativa	1.00	1.00	1.00	15
Artocarpus altilis	0.83	0.67	0.74	15
Colocasia esculenta	0.83	0.67	0.74	15
Nelumbo nucifera	0.82	0.93	0.88	15
Derris elliptica	0.62	0.87	0.72	15

Accuracy			0.84	225
Macro average	0.85	0.84	0.84	225
Weighted average	0.85	0.84	0.84	225

Source: Author's analysis.

Confusion-matrix analysis (Figure 3) shows the model achieved perfect classification (15 of 15 correct) for *Jatropha curcas*, *Morus alba*, *Musa paradisiaca*, and *Lactuca sativa*, indicating the architecture extracted strong distinguishing features for these four species. High performance was also shown for *Datura metel*, *Bambusa vulgaris*, and *Nelumbo nucifera* (14 of 15 correct), and for *Eichhornia crassipes*, *Pilea trinevia*, and *Derris elliptica* (13 of 15 correct). The highest misclassification rate occurred for *Cerbera manghas*, with only 8 of 15 images correctly classified; the remainder were misclassified as *Artocarpus altilis* (2 images), *Derris elliptica* (2 images), *Eichhornia crassipes* (1 image), *Syzygium aqueum* (1 image), and *Datura metel* (1 image). Similar but milder misclassification occurred for *Ipomoea aquatica*, *Syzygium aqueum*, *Artocarpus altilis*, and *Colocasia esculenta* (10 of 15 correct each). These misclassifications are likely due to visual similarity between species in leaf shape, color, and texture, particularly since the images captured whole plants rather than single leaves, so variation in background and camera angle also affected the features the model learned. Compared with prior CNN-based herbal-plant studies using single-leaf images, which typically report 92%–99% accuracy (Abdiel & Yuhri, 2026; Adelia, Fitri, & Agusniar, 2025; Setiyono et al., 2023), the 84% accuracy achieved here is reasonable given that the research object was whole waterlogging-tolerant plant images from more diverse data sources and without data augmentation.

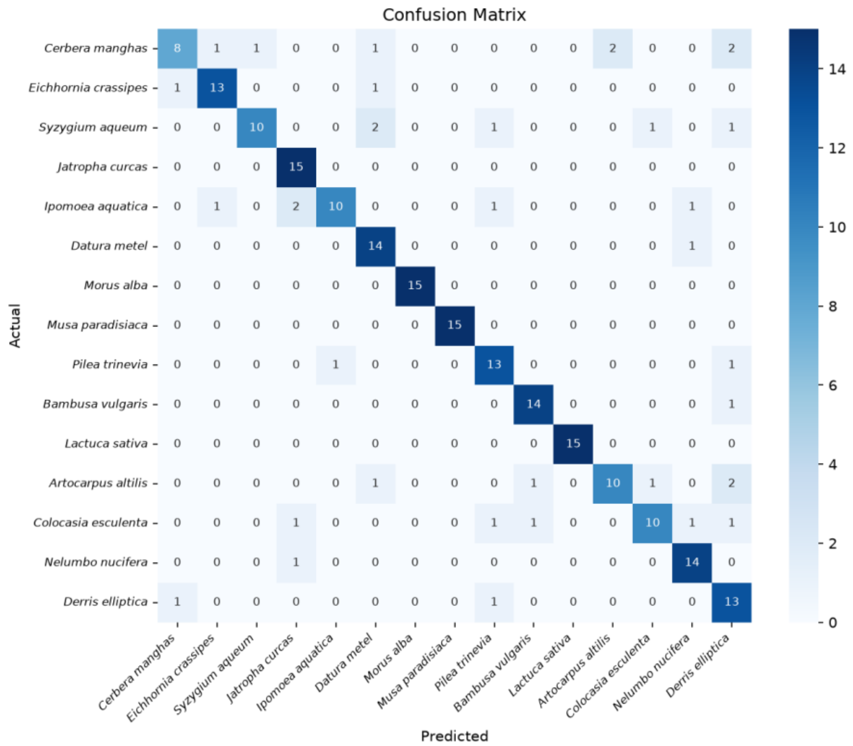

Figure 3. Confusion Matrix of the Test Results
Source: Author's analysis.

Black Box Testing of 19 core feature scenarios — including administrator authentication, plant identification, plant-data management, identification history, model training, and model management — showed that all scenarios produced output consistent with expected results (valid).

API integration testing of 11 communication processes between the Laravel application and the Machine Learning Service, including sending images to the /predict endpoint, asynchronous training via /training/start and /training/status/{task_id}, and model activation via /models/activate, also confirmed that all data-exchange processes worked according to the system design (Table 2).

Table 2. Summary of Black Box and API Integration Testing

Test Type	Count	Coverage	Result
Black Box Testing	19 scenarios	Administrator authentication, plant identification, plant-data management, identification history, model training, model management	All scenarios consistent with design (valid)
API Integration Testing	11 processes	Endpoints /predict, /training/start, /training/status/{task_id}, /models, /models/activate, /models/report/{model_name}	All processes consistent with design (valid)

Source: Author's analysis.

Overall, the developed system can automatically identify waterlogging-tolerant plants and display consumption-feasibility status, characteristics, edible parts, and processing methods based on the reference database. The training and model-management features through the web interface also allow administrators to update the classification model without manually running scripts or stopping the running Machine Learning service. Nevertheless, the system still has limitations, including its restricted coverage of only 15 plant species, imperfect model accuracy, and dependence of identification results on the quality of images uploaded by users.

4. Conclusion

Based on the research, implementation, and testing conducted, an image-based identification system for waterlogging-tolerant plants was successfully designed and built using CNN with the MobileNetV2 architecture, integrating a Laravel web application with a FastAPI-based Machine Learning Service through REST API. The CNN MobileNetV2 model, trained via transfer learning (feature extraction), successfully classified 15 waterlogging-tolerant plant species with 84% accuracy, 85% average precision, 84% recall, and 84% F1-score on the test data. The system can determine and display plant consumption-feasibility status along with characteristics, edible parts, and processing methods based on identification results integrated with the plant database. Black Box Testing of 19 core scenarios and API integration testing of 11 communication processes confirmed that all system features worked according to the established design. Nevertheless, misclassification still occurred for species with high visual similarity, such as Cerbera manghas, Ipomoea aquatica, Syzygium aqueum, Artocarpus altilis, and Colocasia esculenta, indicating that the model still has limitations in distinguishing similar visual characteristics between classes. Future development should apply data augmentation, perform further fine-tuning by unfreezing part of the final MobileNetV2 layers, add more images and variation for lower-performing classes, expand the coverage of waterlogging-tolerant species, and add an authentication mechanism (token/API key) to inter-service communication endpoints to improve system security.

Acknowledgements

The authors would like to thank the Information Systems Study Program, Faculty of Computer Science and Information Technology, Universitas Muhammadiyah Sumatera Utara, for its support throughout this research.

References

Abdiel, F. K., & Yuhri, M. S. (2026). Klasifikasi 10 jenis daun tanaman herbal Indonesia menggunakan CNN dengan arsitektur MobileNetV2 berbasis WEB. Seminar Nasional Teknologi & Sains (STAINS), 5(1).

Adelia, D., Fitri, Z., & Agusniar, C. (2025). Deteksi daun herbal dan beracun menggunakan convolutional neural network untuk klasifikasi tanaman herbal dan beracun. Rabit: Jurnal Teknologi dan Sistem Informasi Univrab, 10(2), 204–216.

Alamsyah, S. F. (2019). Implementasi deep learning untuk klasifikasi tanaman TOGA berdasarkan ciri daun berbasis android. Ubiquitous: Computers and its Applications Journal, 2(2), 113–122.

Alzubaidi, L., Zhang, J., Humaidi, A. J., Al-Dujaili, A., Duan, Y., Al-Shamma, O., Santamaría, J., Fadhel, M. A., Al-Amidie, M., & Farhan, L. (2021). Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions. Journal of Big Data, 8, Article 53.

Badan Nasional Penanggulangan Bencana. (2024). Buku data bencana Indonesia tahun 2023. Pusdatinkom BNPB. https://bnpb.go.id/buku/buku-data-bencana-indonesia-tahun-2023

Bismi, W., Novianti, D., & Qomaruddin, M. (2024). Analisis perbandingan klasifikasi citra genus Panthera dengan pendekatan deep learning model MobileNet. Jurnal Informatika dan Rekayasa Perangkat Lunak, 6(1), 1–9.

Butar-Butar, R. J. H., & Marpaung, N. L. (2023). Deep learning untuk identifikasi daun tanaman obat menggunakan transfer learning MobileNetV2. Jurnal Informatika: Jurnal Pengembangan IT, 8(2).

Fahmi, A., & Wakhid, N. (2018). Karakteristik lahan rawa. Dalam Agroekologi Rawa (pp. 91–118). PT Rajagrafindo Persada.

Felix, F., Wijaya, J., Sutra, S. P., Kosasih, P. W., & Sirait, P. (2020). Implementasi convolutional neural network untuk identifikasi jenis tanaman melalui daun. Jurnal Sifo Mikroskil, 21(1), 1–10.

Hariman, A. A., Mulyana, D. I., & Yel, M. B. (2023). Klasifikasi jajanan tradisional Jawa Tengah dengan metode transfer learning dan MobileNetV2. Informasi Interaktif: Jurnal Informatika dan Teknologi Informasi, 8(1).

Ichwan, M., & Sumantri, H. (2024). Evaluasi kinerja model Inception ResNet-V2 dan Inception-V4 dalam mengklasifikasi kualitas biji kakao. MIND (Multimedia Artificial Intelligent Networking Database) Journal, 9(1), 25–41.

Loreti, E., & Perata, P. (2020). The many facets of hypoxia in plants. Plants, 9(6), 745.

Mulyana, S., Mansur AS, Warjaya, A., Muthmainnah, I., Al Idrus, S. I., & Indra, Z. (2025). Identifikasi penyakit tanaman berdasarkan citra daun berbasis web dengan pendekatan algoritma convolutional neural network. SKANIKA: Sistem Komputer dan Teknik Informatika, 8(2), 305–317.

Nasution, W. A., Niska, D. Y., Al Idrus, S. I., Indra, Z., & Taufik, I. (2025). Implementasi metode convolutional neural network (CNN) dalam klasifikasi tanaman herbal berbentuk rimpang pada sistem informasi berbasis website. JATI: Jurnal Mahasiswa Teknik Informatika, 9(5).

Ningsih, N. P., Suryadi, E., Bakti, L. D., & Imran, B. (2022). Klasifikasi penyakit early blight dan late blight pada tanaman tomat berdasarkan citra daun menggunakan metode convolutional neural network berbasis website. Jurnal Kecerdasan Buatan dan Teknologi Informasi, 1(3), 27–35.

Pan, J., Sharif, R., Xu, X., & Chen, X. (2021). Mechanisms of waterlogging tolerance in plants: Research progress and prospects. Frontiers in Plant Science, 11, Article 627331.

Prabaningrum, H., Nugroho, A. S., & Kaswinarni, F. (2018). Keanekaragaman tumbuhan yang berpotensi sebagai bahan pangan di Cagar Alam Gebugan Semarang. Jurnal Biologi dan Pembelajarannya (JB&P), 5(2).

Putra, G. A., Cahyani, D., Marpaung, D. S. S., & Fil'aini, R. (2020). Fenotiping digital tanaman: Tinjauan terhadap teknologi dan algoritma pengolahan citra. Jurnal Industri Pertanian, 2(1), 130–141.

Saidi, I. A., Azara, R., Ramadhani N., S., & Yanti, E. (2022). Nutrisi dan komponen bioaktif pada sayuran daun. UMSIDA Press.

Setiyono, B., dkk. (2023). Identifikasi tanaman obat Indonesia melalui citra daun menggunakan metode convolutional neural network (CNN). Jurnal Teknologi Informasi dan Ilmu Komputer, 10(2).

Sinlae, F., Irwanda, E., Maulana, Z., & Syahputra, V. E. (2024). Penggunaan framework Laravel dalam membangun aplikasi website berbasis PHP. Jurnal Siber Multi Disiplin, 2(2), 119–132.

Sitorus, R. E. W. B., Batee, D., Hutagaol, G. H., Rachmika, M., Christian, D., & Sinaga, F. M. (2025). Klasifikasi jenis tumbuhan batang daun dan buah. Jurnal Sifo Mikroskil (JSM), 26(1), 55–70.