

Analisis dan Perancangan Aplikasi Identifikasi Penyakit Daun Jagung Berbasis Mobile

Ahmad Balya Diwanullatif¹, Salamun Rohman Nudin²

Program Studi Manajemen Informatika Universitas Negeri Surabaya
Jl. Ketintang, Ketintang, Kec. Gayungan, Surabaya, Jawa Timur 6031

¹ahmad.22095@mhs.unesa.ac.id

²salamunrohman@unesa.ac.id

Abstrak - Identifikasi penyakit daun jagung yang masih dilakukan secara manual oleh petani rentan menghasilkan diagnosis yang tidak konsisten dan berpotensi menimbulkan keterlambatan penanganan, sehingga penyebaran penyakit sulit dikendalikan. Penelitian ini bertujuan menganalisis kebutuhan pengguna serta merancang aplikasi mobile identifikasi penyakit daun jagung yang dapat digunakan secara offline. Metode yang digunakan meliputi identifikasi masalah, pengumpulan data melalui studi literatur dan observasi, analisis kebutuhan fungsional dan nonfungsional, serta perancangan sistem menggunakan proses bisnis, use case diagram, class diagram, sequence diagram, dan entity relationship diagram (ERD). Hasil penelitian berupa rancangan antarmuka aplikasi dalam bentuk wireframe yang mencakup halaman autentikasi, menu utama, panduan pengambilan citra, hasil deteksi, riwayat, dan profil. Rancangan sistem mengadopsi arsitektur TensorFlow Lite dan basis data lokal SQLite agar aplikasi dapat beroperasi tanpa koneksi internet, sehingga dapat membantu petani mengidentifikasi penyakit daun jagung secara cepat dan akurat langsung dari lapangan.

Kata kunci: Aplikasi Mobile, Identifikasi Penyakit Daun Jagung, Perancangan Sistem, UML, Wireframe

Abstract - The manual identification of corn leaf diseases by farmers is prone to inconsistent outcomes and potential delays in treatment, making disease spread difficult to control. This study aims to analyze user requirements and design a mobile application for identifying corn leaf diseases that can operate offline. The methods used include problem identification, data collection through literature study and observation, functional and non-functional requirement analysis, and system design using business processes, use case diagrams, class diagrams, sequence diagrams, and entity relationship diagrams (ERD). The results are presented as an application interface design in the form of wireframes, covering authentication, main menu, image capture guide, detection result, history, and profile pages. The system design adopts a TensorFlow Lite based architecture and a local SQLite database so the application can operate offline, and is expected to serve as a reference for developing an application that helps farmers identify corn leaf diseases quickly and accurately in the field.

Keywords: Corn Leaf Disease Identification, Mobile Application, System Design, UML, Wireframe

I. PENDAHULUAN

Jagung (*Zea mays* L.) merupakan komoditas pangan strategis yang menjadi sumber karbohidrat utama kedua di Indonesia setelah padi [1]. Produktivitas jagung kerap terganggu oleh serangan penyakit daun seperti Blight, Common Rust, dan Gray Leaf Spot yang apabila tidak ditangani sejak dini dapat menurunkan hasil panen secara signifikan. Identifikasi penyakit yang masih dilakukan secara manual berdasarkan pengamatan visual sangat bergantung pada pemahaman masing-masing petani, sehingga hasil identifikasi sering kali tidak konsisten antara satu petani dengan petani lainnya, bahkan berpotensi menimbulkan keterlambatan penanganan yang memperluas area serangan penyakit [2].

Kemajuan kecerdasan buatan, khususnya Convolutional Neural Network (CNN) berarsitektur AlexNet [3], telah banyak dimanfaatkan untuk mengotomatisasi identifikasi penyakit tanaman melalui citra daun. Penelitian terapan melaporkan bahwa arsitektur CNN mampu mencapai akurasi tinggi dalam mengklasifikasikan penyakit daun jagung [2], dan berbagai studi lain turut membuktikan keberhasilan pipeline serupa mulai dari pelatihan model hingga konversinya ke format TensorFlow Lite (TFLite) agar dapat dijalankan pada perangkat mobile [4].

Di sisi lain, penerapan teknologi digital di sektor pertanian, khususnya bagi petani skala kecil, masih menghadapi berbagai tantangan non-teknis seperti keterbatasan literasi digital dan minimnya aplikasi yang benar-benar dirancang sesuai kebutuhan dan konteks kerja petani di lapangan [5]. Kesenjangan antara keberhasilan model kecerdasan buatan secara algoritmik dengan kesiapannya menjadi aplikasi yang benar-benar dapat digunakan oleh pengguna akhir non-teknis merupakan tantangan tersendiri yang memerlukan disiplin rekayasa perangkat lunak yang memadai, dimulai dari tahap analisis kebutuhan dan perancangan sistem yang matang sebelum masuk ke tahap implementasi [6].

Beberapa penelitian telah mengarah pada perancangan maupun implementasi aplikasi mobile untuk identifikasi penyakit tanaman. AgroAid mengembangkan aplikasi mobile lintas platform untuk klasifikasi spesies dan penyakit tanaman menggunakan TensorFlow Lite dengan antarmuka yang ramah pengguna [7]. Penelitian lain mengimplementasikan MobileNetV3Large untuk deteksi

penyakit daun padi pada aplikasi mobile [8], serta sistem pemantauan real-time berbasis deep learning untuk deteksi penyakit daun tanaman [9]. Pendekatan user-centered design juga telah diterapkan pada perancangan aplikasi manajemen peternakan berbasis mobile untuk memastikan aplikasi sesuai dengan kebutuhan pengguna di lapangan [10].

Berdasarkan uraian tersebut, penelitian ini difokuskan pada tahap analisis kebutuhan dan perancangan sistem aplikasi mobile untuk identifikasi penyakit daun jagung, dengan arsitektur AlexNet berformat TensorFlow Lite [3] dan basis data lokal SQLite [11] ditetapkan sebagai teknologi acuan yang akan diintegrasikan pada tahap pengembangan selanjutnya, menggunakan framework Flutter [12] sebagai kerangka kerja implementasi antarmuka. Fokus utama penelitian ini adalah identifikasi masalah, pengumpulan data, analisis kebutuhan fungsional dan nonfungsional, perancangan proses bisnis, use case diagram, class diagram, sequence diagram, entity relationship diagram (ERD), serta perancangan antarmuka pengguna dalam bentuk wireframe. Rancangan yang dihasilkan pada penelitian ini diharapkan dapat menjadi acuan yang jelas dan terstruktur bagi tahap implementasi aplikasi selanjutnya, sehingga proses pengembangan dapat berjalan lebih terarah dan sesuai dengan kebutuhan nyata petani di lapangan.

II. METODE PENELITIAN

Metode penelitian yang digunakan dalam penelitian ini terdiri atas beberapa tahapan yang dilakukan untuk menghasilkan rancangan aplikasi identifikasi penyakit daun jagung berbasis mobile. Tahapan penelitian dimulai dari identifikasi permasalahan hingga menghasilkan rancangan antarmuka (wireframe) sebagai acuan pengembangan aplikasi.

A. Identifikasi Masalah

Tahap awal penelitian dilakukan dengan mengidentifikasi permasalahan yang terjadi dalam proses identifikasi penyakit daun tanaman jagung. Identifikasi dilakukan untuk mengetahui kendala yang dihadapi petani dalam mengenali jenis penyakit daun jagung, termasuk keterbatasan pengetahuan petani awam mengenai gejala penyakit dan ketergantungan pada koneksi internet yang tidak selalu tersedia di area perkebunan, sehingga dapat disusun kebutuhan aplikasi yang dapat membantu proses identifikasi secara lebih praktis, mandiri, dan efisien.

B. Pengumpulan Data

Tahap selanjutnya merupakan pengumpulan informasi yang menjadi dasar dalam proses analisis dan perancangan sistem. Pengumpulan data dilakukan melalui dua metode, yaitu studi literatur dan observasi. Studi literatur dilakukan dengan mempelajari jurnal ilmiah, artikel, dan penelitian terdahulu yang berkaitan dengan identifikasi penyakit daun jagung serta perancangan aplikasi mobile berbasis kecerdasan buatan. Metode observasi dilakukan untuk memperoleh gambaran

mengenai proses identifikasi penyakit daun jagung yang berlangsung di lapangan, sehingga dapat diketahui kebutuhan pengguna serta alur kerja yang akan diterapkan pada sistem.

C. Analisis Kebutuhan

Data yang telah diperoleh kemudian dianalisis untuk mengidentifikasi kebutuhan sistem, meliputi kebutuhan fungsional, kebutuhan nonfungsional, serta karakteristik pengguna. Hasil analisis menjadi dasar dalam menentukan fitur-fitur yang akan tersedia pada aplikasi identifikasi penyakit daun jagung.

Kebutuhan fungsional aplikasi mencakup: (1) autentikasi pengguna melalui login dan registrasi akun; (2) pengambilan citra daun jagung melalui kamera perangkat atau pemilihan gambar dari galeri; (3) proses identifikasi penyakit secara otomatis menggunakan model klasifikasi; (4) penyajian hasil identifikasi beserta tingkat kepercayaan dan rekomendasi penanganan; (5) penyimpanan riwayat hasil identifikasi secara lokal; serta (6) pengelolaan profil pengguna.

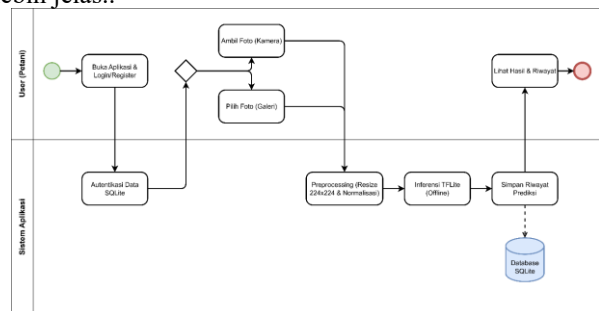
Kebutuhan nonfungsional mencakup: (1) usability, yaitu antarmuka harus mudah dipahami dan dioperasikan oleh petani awam yang belum terbiasa dengan aplikasi teknologi kompleks [10]; (2) reliability, yaitu aplikasi harus tetap dapat beroperasi secara offline mengingat banyak area perkebunan yang rawan blank spot sinyal [13]; (3) performance, yaitu proses identifikasi harus berjalan responsif dengan memanfaatkan model yang relatif ringan [14]; dan (4) security, yaitu data akun dan riwayat pengguna perlu disimpan dengan mekanisme yang memperhatikan aspek keamanan data.

D. Perancangan Sistem

Tahap perancangan sistem dilakukan berdasarkan hasil analisis kebutuhan. Perancangan sistem meliputi proses bisnis, Use Case Diagram, Class Diagram, dan Sequence Diagram menggunakan pendekatan Unified Modeling Language (UML) [15], serta perancangan basis data menggunakan Entity Relationship Diagram (ERD).

1) Proses Bisnis

Proses bisnis menggambarkan alur interaksi pengguna dengan sistem dalam menjalankan fungsi utama aplikasi [16], mulai dari pengguna membuka aplikasi hingga memperoleh hasil identifikasi penyakit daun jagung, sehingga kebutuhan fungsional sistem dapat diidentifikasi lebih jelas..

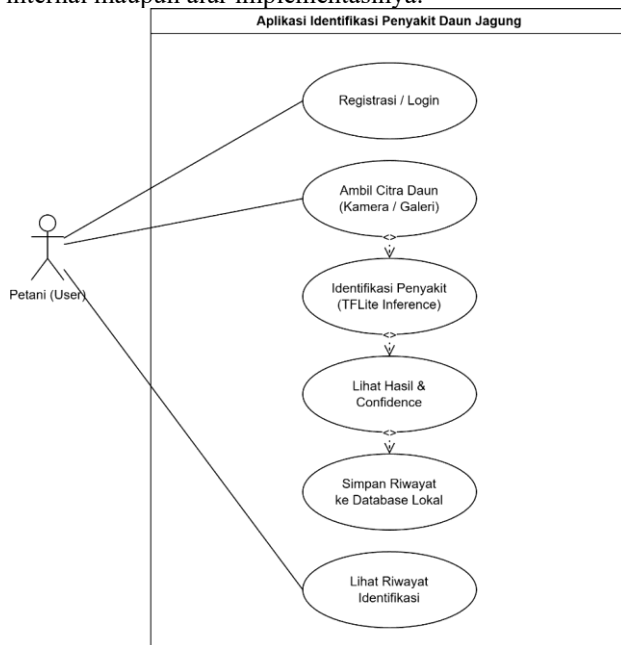


Gambar. 1 Proses Bisnis Aplikasi

Berdasarkan Gambar 1, proses bisnis diawali ketika pengguna membuka aplikasi dan melakukan autentikasi melalui login atau registrasi yang dikelola oleh basis data lokal. Selanjutnya, pengguna memilih fitur identifikasi dengan mengambil gambar daun menggunakan kamera atau mengunggah gambar dari galeri. Sistem kemudian melakukan prapemrosesan terhadap citra sebelum diproses untuk mengidentifikasi penyakit pada daun jagung secara on-device. Hasil identifikasi beserta rekomendasi penanganan ditampilkan kepada pengguna dan secara otomatis disimpan sebagai riwayat, yang dapat diakses maupun dihapus kembali oleh pengguna melalui menu riwayat.

2) Use Case Diagram

Use case diagram merupakan salah satu diagram dalam UML yang digunakan untuk menggambarkan interaksi antara aktor dengan sistem berdasarkan fungsi atau layanan yang tersedia [17]. Diagram ini berfokus pada kebutuhan fungsional sistem dari sudut pandang pengguna sehingga dapat memberikan gambaran mengenai fitur-fitur yang dapat dijalankan tanpa menjelaskan proses internal maupun alur implementasinya.

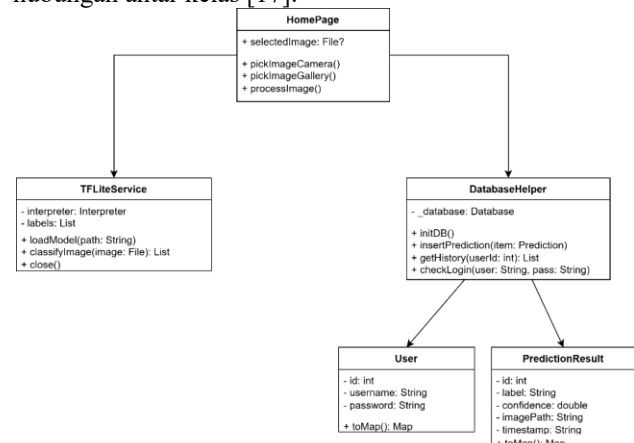


Gambar. 2 Use Case Diagram

Use case diagram pada Gambar 2 menunjukkan interaksi antara aktor berupa petani dengan aplikasi identifikasi penyakit daun jagung. Pengguna dapat melakukan input gambar daun melalui dua cara, yaitu mengambil foto secara langsung menggunakan kamera atau memilih gambar dari galeri perangkat. Setelah gambar diperoleh, sistem menjalankan model identifikasi untuk menganalisis citra daun dan menghasilkan informasi mengenai penyakit yang terdeteksi beserta rekomendasi penanganannya. Pengguna juga dapat melihat dan menghapus riwayat hasil identifikasi yang tersimpan pada basis data lokal.

3) Class Diagram

Class diagram digunakan untuk menggambarkan struktur statis suatu sistem, menunjukkan kelas-kelas yang terdapat pada sistem beserta atribut, operasi (method), dan hubungan antar kelas [17].

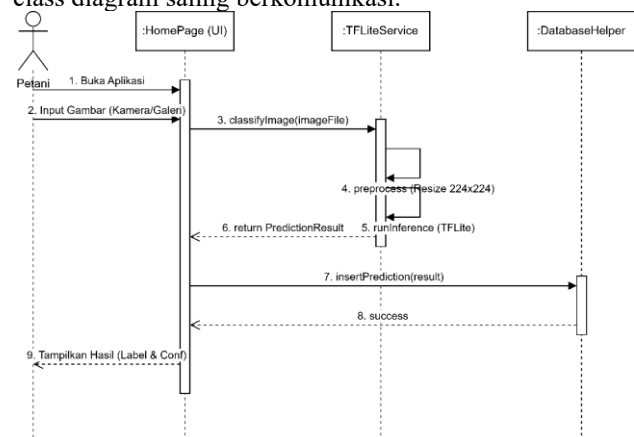


Gambar. 3 Class Diagram

Class diagram pada Gambar 3 memperlihatkan hubungan antar kelas yang direncanakan untuk membangun aplikasi identifikasi penyakit daun jagung. Kelas HomePage berperan sebagai penghubung utama yang meneruskan citra ke kelas TFLiteService untuk diproses menjadi hasil klasifikasi, yang kemudian diteruskan ke kelas DatabaseHelper untuk disimpan sebagai riwayat. Dengan adanya class diagram, struktur aplikasi dapat dipahami secara lebih jelas sehingga memudahkan proses implementasi pada tahap pengembangan selanjutnya.

4) Sequence Diagram

Sequence diagram digunakan untuk menggambarkan urutan interaksi antarobjek dalam menjalankan suatu skenario tertentu berdasarkan waktu [17]. Diagram ini melengkapi use case diagram dengan menunjukkan bagaimana objek-objek yang telah didefinisikan pada class diagram saling berkomunikasi.



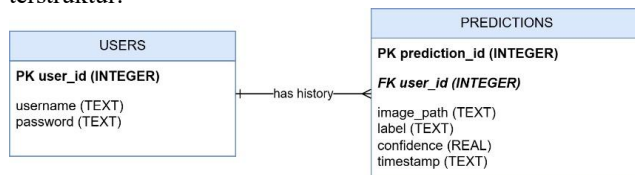
Gambar. 4 Sequence Diagram Proses Identifikasi

Sequence diagram pada Gambar 4 menggambarkan rencana alur interaksi dalam skenario identifikasi penyakit, dimulai ketika pengguna memasukkan gambar

pada objek HomePage, yang kemudian meneruskan berkas citra ke objek TFLiteService melalui fungsi `classifyImage()`. Objek tersebut direncanakan melakukan prapemrosesan citra dan mengembalikan hasil klasifikasi ke HomePage, sebelum akhirnya diteruskan ke objek DatabaseHelper untuk disimpan melalui fungsi `insertPrediction()`.

5) Entity Relationship Diagram (ERD)

ERD merupakan diagram yang digunakan untuk menggambarkan struktur basis data serta hubungan antar entitas dalam sistem [18]. ERD berfungsi sebagai rancangan awal dalam pembuatan basis data agar proses penyimpanan dan pengelolaan data dapat dilakukan secara terstruktur.



Gambar. 5 Entity Relationship Diagram (ERD)

Berdasarkan Gambar 5, basis data yang dirancang terdiri atas dua entitas utama, yaitu Users yang menyimpan data identitas pengguna dengan `user_id` sebagai kunci utama (primary key), dan History yang menyimpan riwayat hasil deteksi mencakup lokasi berkas citra, label penyakit, nilai kepercayaan, dan waktu pemindaian. Relasi antara kedua entitas dirancang bersifat one-to-many melalui kunci tamu (foreign key) `user_id` pada entitas History, sehingga setiap pengguna dapat memiliki banyak riwayat identifikasi yang terisolasi satu sama lain.

E. Perancangan Antarmuka Pengguna

Perancangan antarmuka pengguna (user interface) dilakukan dengan membuat purwarupa (prototype) desain menggunakan aplikasi Figma [19], karena menyediakan fitur untuk mendesain user interface secara mudah dan terstruktur. Hasil dari tahap ini berupa wireframe yang menggambarkan tata letak, navigasi, serta komponen pada setiap halaman aplikasi, tanpa menerapkan detail visual seperti warna maupun aset gambar akhir. Prinsip desain yang diterapkan mengutamakan kesederhanaan tata letak dan ikonografi yang intuitif, mengingat target pengguna utama aplikasi adalah petani yang sebagian besar belum terbiasa dengan aplikasi teknologi kompleks [10]. Rancangan antarmuka mencakup enam halaman utama, yaitu halaman autentikasi, halaman menu utama, halaman panduan pengambilan citra, halaman hasil deteksi, halaman riwayat, dan halaman profil pengguna, yang akan diuraikan lebih lanjut pada bagian Hasil dan Pembahasan.

III. HASIL DAN PEMBAHASAN

Penelitian ini menghasilkan rancangan antarmuka aplikasi identifikasi penyakit daun jagung berbasis mobile yang disusun berdasarkan hasil analisis kebutuhan dan perancangan sistem pada tahap sebelumnya. Rancangan

antarmuka dibuat menggunakan Figma dalam bentuk wireframe untuk menggambarkan struktur tampilan, navigasi, dan interaksi pengguna pada masing-masing halaman aplikasi.

A. Halaman Autentikasi

Halaman autentikasi dirancang sebagai gerbang akses pertama yang ditemui pengguna saat aplikasi dijalankan, sebagaimana ditunjukkan pada rancangan wireframe berikut.



Gambar. 6 Wireframe Halaman Autentikasi

Gambar 6 merupakan rancangan halaman yang muncul saat aplikasi pertama kali dijalankan, berfungsi sebagai gerbang akses awal yang mengelola sesi login maupun registrasi pengguna secara lokal. Wireframe ini menampilkan area logo aplikasi, kolom isian email dan kata sandi, tombol aksi utama untuk masuk, serta tautan menuju halaman registrasi bagi pengguna baru.

B. Halaman Menu Utama

Setelah proses autentikasi berhasil dilakukan, pengguna akan diarahkan menuju halaman menu utama yang berperan sebagai pusat navigasi aplikasi, seperti diperlihatkan pada rancangan wireframe berikut. Gambar 7 menunjukkan rancangan halaman menu utama yang berfungsi sebagai pusat navigasi aplikasi setelah pengguna berhasil masuk. Rancangan ini menyediakan dua tombol aksi utama untuk memulai identifikasi, yaitu mengambil foto langsung dari kamera atau memilih citra dari galeri, disertai kotak informasi penggunaan singkat bagi pengguna baru serta navigasi bawah menuju halaman riwayat dan profil.



Gambar. 7 Wireframe Halaman Menu Utama

C. Halaman Panduan Pengambilan Citra



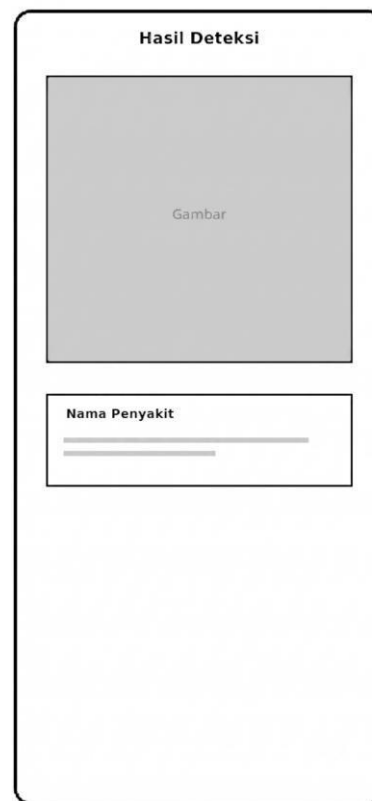
Gambar. 8 Wireframe Halaman Panduan Pengambilan Citra

Sebelum kamera perangkat diaktifkan, pengguna terlebih dahulu disajikan halaman panduan pengambilan

citra yang bertujuan meminimalisasi kesalahan saat pengambilan gambar daun, sebagaimana digambarkan pada rancangan wireframe berikut. Gambar 8 merupakan rancangan dialog panduan yang akan ditampilkan sebelum pengguna diarahkan ke kamera perangkat, sebagai penerapan prinsip Human-Computer Interaction (HCI) yang dirancang untuk meminimalisasi kesalahan pengguna (human error) saat pengambilan citra [20]. Rancangan ini memuat empat instruksi, yaitu menstabilkan kamera, memastikan pencahayaan cukup, menjaga jarak ideal 15–25 cm, dan menunggu autofocus, sebelum pengguna menekan tombol untuk membuka kamera.

D. Halaman Hasil Deteksi

Setelah citra daun selesai diproses oleh model identifikasi, sistem akan menampilkan halaman hasil deteksi kepada pengguna, seperti ditunjukkan pada rancangan wireframe berikut.



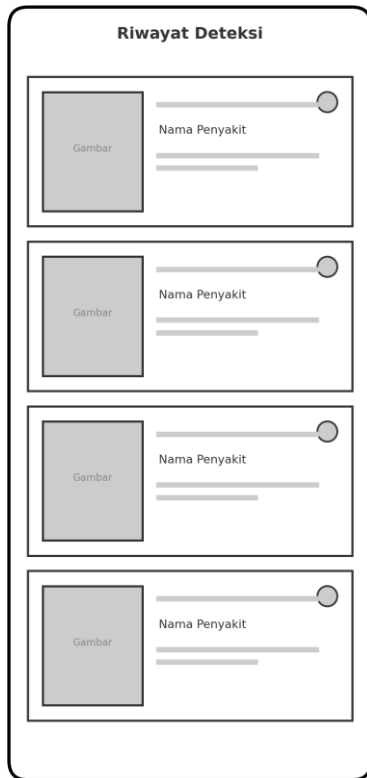
Gambar. 9 Wireframe Halaman Hasil Deteksi

Gambar 9 menampilkan rancangan halaman yang akan menyajikan hasil identifikasi setelah citra diproses, mencakup area pratinjau gambar daun, informasi nama penyakit yang terdeteksi, serta rekomendasi penanganan sesuai penyakit yang teridentifikasi. Selain itu, rancangan ini turut menampilkan tingkat keyakinan (confidence score) dari hasil klasifikasi model, sehingga pengguna dapat mengetahui seberapa akurat prediksi yang diberikan sebelum mengambil keputusan penanganan. Informasi tersebut disusun secara ringkas dan mudah dipahami agar petani dapat langsung memahami kondisi tanaman tanpa

memerlukan pengetahuan teknis mengenai model klasifikasi yang digunakan. Rancangan ini juga menyediakan tombol aksi untuk menyimpan hasil identifikasi ke dalam riwayat, sehingga data tersebut dapat diakses kembali sewaktu-waktu tanpa harus mengulang proses pengambilan maupun identifikasi citra.

E. Halaman Riwayat

Untuk memudahkan pengguna meninjau kembali hasil identifikasi yang pernah dilakukan, aplikasi menyediakan halaman riwayat, sebagaimana digambarkan pada rancangan wireframe berikut.



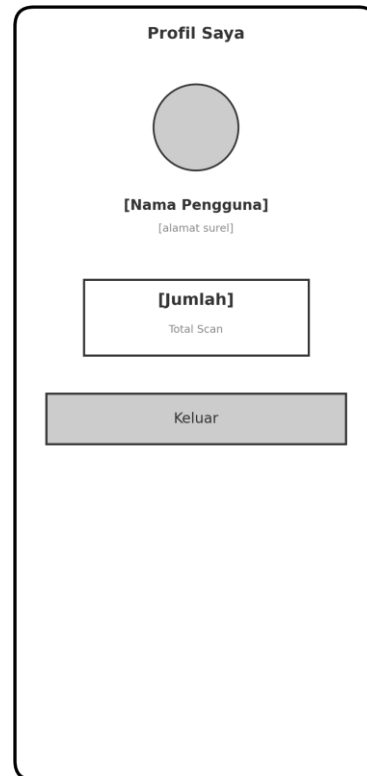
Gambar. 10 Wireframe Halaman Riwayat

Gambar 10 merupakan rancangan halaman yang akan menampilkan daftar riwayat hasil identifikasi yang pernah dilakukan pengguna, berupa pratinjau gambar, label penyakit, dan informasi terkait dalam bentuk daftar. Rancangan ini memungkinkan pengguna untuk meninjau kembali hasil identifikasi sebelumnya tanpa perlu mengulang proses pengambilan gambar.

F. Halaman Profil

Halaman profil dirancang untuk menampilkan informasi akun serta ringkasan aktivitas identifikasi yang telah dilakukan pengguna, seperti diperlihatkan pada rancangan wireframe berikut. Gambar 11 menunjukkan rancangan halaman profil yang akan menampilkan informasi akun pengguna seperti nama dan alamat surel, serta ringkasan jumlah aktivitas identifikasi yang telah dilakukan. Rancangan ini disusun secara sederhana agar pengguna dapat dengan cepat memantau riwayat penggunaan aplikasi tanpa harus berpindah ke halaman

lain. Halaman ini juga menyediakan tombol untuk keluar (logout) dari sesi aktif, sebagai bentuk pengelolaan sesi pengguna yang memastikan keamanan data akun tetap terjaga, khususnya pada perangkat yang digunakan secara bersama oleh beberapa pengguna di lapangan.



Gambar. 11 Wireframe Halaman Profil

G. Pembahasan

Rancangan sistem yang dihasilkan menggambarkan alur kerja aplikasi secara menyeluruh, mulai dari autentikasi, pengambilan citra, proses identifikasi, hingga penyajian hasil dan penyimpanan riwayat. Pemilihan arsitektur berbasis TensorFlow Lite dan basis data lokal SQLite pada tahap perancangan didasari oleh kebutuhan nonfungsional aplikasi yang harus tetap dapat beroperasi secara offline, mengingat banyak area perkebunan yang rawan blank spot sinyal internet.

Rancangan antarmuka panduan pengambilan citra turut mempertimbangkan aspek kualitas citra yang dibutuhkan agar proses ekstraksi fitur oleh model klasifikasi dapat berjalan optimal, sekaligus menjadi bentuk mitigasi kesalahan pengguna yang bersifat preventif dari sisi rancangan antarmuka. Secara keseluruhan, hasil analisis kebutuhan dan perancangan sistem pada penelitian ini diharapkan dapat menjadi acuan yang jelas bagi tahap implementasi aplikasi identifikasi penyakit daun jagung selanjutnya, sehingga proses pengembangan dapat dilakukan secara lebih terarah dan efisien.

IV. KESIMPULAN

Penelitian ini menghasilkan analisis kebutuhan dan rancangan aplikasi identifikasi penyakit daun jagung yang

ditujukan untuk membantu pengembangan lanjutan berbasis mobile agar petani dapat melakukan identifikasi secara lebih cepat, mandiri, dan praktis di lapangan. Analisis kebutuhan yang dilakukan melalui studi literatur dan observasi menghasilkan spesifikasi kebutuhan fungsional dan nonfungsional yang menjadi dasar perancangan sistem. Perancangan sistem direpresentasikan melalui proses bisnis, use case diagram, class diagram, sequence diagram, dan entity relationship diagram (ERD), sedangkan perancangan antarmuka menghasilkan wireframe yang mencakup enam halaman utama, yaitu halaman autentikasi, halaman menu utama, halaman panduan pengambilan citra, halaman hasil deteksi, halaman riwayat, dan halaman profil. Rancangan yang dihasilkan menggambarkan alur kerja sistem secara menyeluruh, mulai dari proses autentikasi dan pengambilan citra, identifikasi penyakit menggunakan model klasifikasi berbasis TensorFlow Lite, hingga penyajian hasil beserta rekomendasi penanganan dan penyimpanan riwayat secara offline melalui basis data lokal SQLite. Dengan adanya rancangan ini, pengembangan aplikasi pada tahap selanjutnya diharapkan dapat dilakukan secara lebih terarah dan terstruktur, sehingga menghasilkan sistem yang mudah digunakan dan mampu mendukung upaya identifikasi dini penyakit daun jagung secara mandiri di lapangan.

REFERENSI

- [1] V. Ardilla, J. Supriyanti, S. Diana P, R. Fevria, and M. Larashinda, "Inventarisasi Penyakit-Penyakit Pada Tanaman Jagung (*Zea Mays* L.) Di Lahan Percobaan Dinas Pertanian Kabupaten Sijunjung," 2024, Accessed: Jul. 06, 2026. [Online]. Available: <https://semnas.agroindustri.fmipa.unp.ac.id/index.php/snagro/article/view/13>
- [2] G. T. Askale, A. B. Yibel, B. M. Taye, and G. D. Wubneh, "Mobile based deep CNN model for maize leaf disease detection and classification," Dec. 01, 2025, *BioMed Central Ltd.* doi: 10.1186/s13007-025-01386-5.
- [3] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," 2012. [Online]. Available: <http://code.google.com/p/cuda-convnet/>
- [4] G. Sakkarvarthi, G. W. Sathianesan, V. S. Murugan, A. J. Reddy, P. Jayagopal, and M. Elsis, "Detection and Classification of Tomato Crop Disease Using Convolutional Neural Network," *Electronics (Switzerland)*, vol. 11, no. 21, Nov. 2022, doi: 10.3390/electronics11213618.
- [5] N. Gumbi, L. Gumbi, and H. Twinomurinzi, "Towards Sustainable Digital Agriculture for Smallholder Farmers: A Systematic Literature Review," *Sustainability*, vol. 15, no. 16, p. 12530, Aug. 2023, doi: 10.3390/su151612530.
- [6] P. Heck and G. Schouten, "Turning Software Engineers into AI Engineers," Jan. 2021, [Online]. Available: <http://arxiv.org/abs/2011.01590>
- [7] M. Reda, R. Suwwan, S. Alkafri, Y. Rashed, and T. Shanableh, "AgroAid: A Mobile App System for Visual Classification of Plant Species and Diseases Using Deep Learning and TensorFlow Lite," *Informatics*, vol. 9, no. 3, Sep. 2022, doi: 10.3390/informatics9030055.
- [8] R. R. Faqih, M. Irsan, and M. F. Fathoni, "Rice Plant Disease Detection System Using Transfer Learning with MobilenetV3Large," *Sinkron*, vol. 8, no. 2, pp. 805–812, Mar. 2024, doi: 10.33395/sinkron.v8i2.13383.
- [9] K. N. Rahman, S. C. Banik, R. Islam, and A. Al Fahim, "A real time monitoring system for accurate plant leaves disease detection using deep learning," *Crop Design*, vol. 4, no. 1, Feb. 2025, doi: 10.1016/j.crope.2024.100092.
- [10] W. Thannithi, P. Intawicha, P. Phuwisaranakom, and S. Saengwong, "Design and evaluation of a mobile application for enhancing farm management and performance assessment in fattening beef cattle," *J. Adv. Vet. Anim. Res.*, no. 0, p. 1, 2024, doi: 10.5455/javar.2024.k766.
- [11] S. T. Bhosale, M. T. Patil, and M. P. Patil, "International Journal of Computer Science and Mobile Computing SQLite: Light Database System," Kolhapur, Apr. 2015. [Online]. Available: www.ijcsmc.com
- [12] J. Christoph, D. Rösch, T. Schuster, and L. Waidelich, "Current Progress in Cross-Platform Application Development-Evaluation of Frameworks for Mobile Application Development 1 PUBLICATION 4 CITATIONS SEE PROFILE 4 PUBLICATIONS 17 CITATIONS SEE PROFILE Current Progress in Cross-Platform Application Development Evaluation of Frameworks for Mobile Application Development," 2019. [Online]. Available: <http://www.iariajournals.org/software/2019>,
- [13] A. N. K. Doodala, "Offline-First Android Architecture for waste management in low connectivity zones," *International Journal of Emerging Trends in Computer Science and Information Technology*, vol. 4, pp. 201–209, 2023, doi: 10.63282/3050-9246.IJETCSIT-V4IIP121.
- [14] J. Pandey and A. R. Asati, "Lightweight convolutional neural network architecture implementation using TensorFlow lite," *International Journal of Information Technology (Singapore)*, vol. 15, no. 5, pp. 2489–2498, Jun. 2023, doi: 10.1007/s41870-023-01320-9.
- [15] S. Baul, J. Ahamed, Md. A. Islam, Md. K. A. Mazumder, T. Sultan, and D. Nandi, "A comprehensive study to assist decision-makers in determining software design between four UML diagrams," in *Proceedings of the 3rd International Conference on Computing Advancements*, New York, NY, USA: ACM, Oct. 2024, pp. 232–238. doi: 10.1145/3723178.3723209.
- [16] S. Farshidi, I. B. Kwantes, and S. Jansen, "Business process modeling language selection for research modelers," *Softw. Syst. Model.*, vol. 23, no. 1, pp. 137–162, Feb. 2024, doi: 10.1007/s10270-023-01110-8.
- [17] M. R. Wayahdi and F. Ruziq, "Pemodelan Sistem Penerimaan Anggota Baru dengan Unified Modeling Language (UML) (Studi Kasus: Programmer Association of Battuta)," *Jurnal Minfo Polgan*, vol. 12, no. 1, pp. 1514–1521, Aug. 2023, doi: 10.33395/jmp.v12i1.12870.
- [18] Suharni, S. Eel, and F. Pakusadewa, "PERANCANGAN WEBSITE RUMAH MAKAN NINIK SEBAGAI MEDIA PROMOSI MENGGUNAKAN UNIFIED MODELLING LANGUAGE," Apr. 2023, Accessed: Jul. 23, 2026. [Online]. Available: <https://journal.istn.ac.id/index.php/rekayasainformasi/article/view/1527/1021>
- [19] N. Kimseng, D. A. Kurnia, I. Vuthy, R. W. Arifin, and D. Setiyadi, "UI/UX Development Using Figma based on Inclusive Design," *JINAV: Journal of Information and Visualization*, vol. 4, no. 2, pp. 227–234, Jan. 2024, doi: 10.35877/454RI.jinav2257.
- [20] A. Wysokińska, K. R. Strömel, and P. W. Woźniak, "Zooming In: A Review of Designing for Photo Taking in Human-Computer Interaction and Future Prospects," *Proc. ACM Hum. Comput. Interact.*, vol. 8, no. ISS, pp. 597–623, Oct. 2024, doi: 10.1145/3698150.