

**IDENTIFIKASI DAUN TANAMAN HERBAL BERKHASIAMAT BAGI
KESEHATAN MENGGUNAKAN ARSITEKTUR *RESNET50*
BERBASIS *WEBSITE***

SKRIPSI

**BELINTAWATI ZELDA BR GINTING
201401003**



**PROGRAM STUDI S-1 ILMU KOMPUTER
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
UNIVERSITAS SUMATERA UTARA
2024**

**IDENTIFIKASI DAUN TANAMAN HERBAL BERKHASIAH BAGI
KESEHATAN MENGGUNAKAN ARSITEKTUR *RESNET50*
BERBASIS *WEBSITE***

SKRIPSI

Diajukan untuk melengkapi tugas dan memenuhi syarat memperoleh ijazah
Sarjana Ilmu Komputer

**BELINTAWATI ZELDA BR GINTING
201401003**



**PROGRAM STUDI S-1 ILMU KOMPUTER
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
UNIVERSITAS SUMATERA UTARA
2024**

PERSETUJUAN

Judul	: IDENTIFIKASI DAUN TANAMAN HERBALBERKHASIAT BAGI KESEHATAN MENGUNAKAN ARSITEKTUR <i>RESNET50</i> BERBASIS <i>WEBSITE</i>	
Kategori	: SKRIPSI	
Nama	: BELINTAWATI ZELDA BR GINTING	
Nomor Induk Mahasiswa	: 201401003	
Program Studi	: SARJANA (S-1) ILMU KOMPUTER	
Fakultas	: ILMU KOMPUTER DAN TEKNOLOGI INFORMASI UNIVERSITAS SUMATERA UTARA	
Komisi Pembimbing	:	
Dosen Pembimbing I		Dosen Pembimbing II

Sri Melvani Hardi S.Kom., M.Kom.
NIP. 198805012015042006

Fanindia Purnamasari S.TI., M.IT.
NIP. 198908172019032023

Diketahui/disetujui oleh
Program Studi S1 Ilmu
Komputer Ketua,

Dr. Amalia ST., M.T.
NIP. 197812212014042001

PERNYATAAN

IDENTIFIKASI DAUN TANAMAN HERBAL BERKHASIAH BAGI KESEHATAN MENGGUNAKAN ARSITEKTUR *RESNET50* BERBASIS *WEBSITE*

SKRIPSI

Saya mengakui bahwa skripsi ini adalah hasil penelitian saya sendiri, kecuali beberapa kutipan dan ringkasan yang masing-masing telah dicantumkan sumbernya.

Medan, 16 Juni 2024

Belintawati Zelda Br Ginting
201401003

PENGHARGAAN

Syukur kepada Tuhan Yang Maha Esa atas rahmat-Nya yang memungkinkan penulis menyelesaikan skripsi ini sebagai salah satu syarat untuk meraih kelulusan di Program Studi S-1 Ilmu Komputer, Fakultas Ilmu Komputer dan Teknologi Informasi, Universitas Sumatera Utara.

Penulis ingin menyampaikan rasa hormat dan terima kasih yang sebesar-besarnya kepada pihak yang telah membantu penulis dalam bentuk doa, bimbingan, semangat, dan ilmu selama pembuatan dan penyelesaian skripsi ini.

1. Bapak Dr. Muryanto Amin, S.Sos., M.Si. selaku Rektor Universitas Sumatera Utara.
2. Ibu Dr. Maya Silvi Lydia, B.Sc., M.Sc. selaku Dekan Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Sumatera Utara.
3. Bapak Dr. Mohammad Andri Budiman, S.T., M.Comp.Sc., M.E.M., selaku Wakil Dekan 1 Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Sumatera Utara.
4. Ibu Dr. Amalia, S.T., M.T. selaku Ketua Program Studi S-1 Ilmu Komputer Universitas Sumatera Utara.
5. Ibu Sri Melvani Hardi, S.Kom., M.Kom., sebagai Sekretaris Program Studi S-1 Ilmu Komputer Universitas Sumatera Utara sekaligus Dosen Pembimbing 1, yang telah memberikan banyak motivasi, saran, dan bimbingan kepada penulis hingga skripsi ini dapat diselesaikan.
6. Ibu Fanindia Purnamasari S.TI., M.IT. selaku Dosen Pembimbing 2 yang telah memberikan banyak motivasi, saran, dan bimbingan kepada penulis hingga skripsi ini dapat terselesaikan.
7. Seluruh dosen, staf pengajar, dan pegawai di Program Studi S-1 Ilmu Komputer, Fakultas Ilmu Komputer dan Teknologi Informasi, Universitas Sumatera Utara.
8. Terkhusus kepada ayah terkasih, Sabta Ginting, yang biasa penulis panggil Bapak, dan ibu tercinta, Denorita Br Sebayang, yang biasa penulis panggil Mamak, yang telah mendukung dan memotivasi penulis baik secara moral maupun materi. Terima kasih atas semangat dan kepercayaan diri yang telah Bapak dan Mamak berikan

kepada penulis, serta selalu siap mendengarkan cerita-cerita penulis. Terima kasih, Mamak dan Bapak, untuk doa dan dukungan yang tiada henti selama ini.

9. Teruntuk adik-adik penulis Juan Barnaby Ginting dan Elite Wilantara Ginting, terimakasih atas dukungan dan semangat juga doa kepada penulis selama pengerjaan skripsi ini.
10. Sahabat penulis sejak SMA, Cheryl Angeline Kaban, Agita Angela Br Tarigan, Erlycasna Br Pelawi, Nitha Esifayona Br Pelawi, Gloury Arizona Sitepu yang selalu mendukung dan mendengarkan cerita penulis selama perjalanan pengerjaan skripsi ini.
11. Marchella Stephanie Putri Agaska Parapat dan Dian Putri Angelia Sitorus yang selalu mendukung, menasehati, dan menemani penulis.
12. Sahabat penulis di Perkuliahan, Sawaliyah Nur Siregar, Syaripa Anum Nasution, Sitti Safiatun Naja Koto, M. Rizky Prayoga yang selalu mendukung, membantu, dan menemani penulis.
13. Teman-Teman di perkuliahan penulis, Chealiza, Andrew, Puan, Rezha, Ayu Wulandari, dan semua teman-teman Ilkomp stambuk 2020.

Semoga Tuhan Yang Maha Esa memberikan berkah kepada semua pihak yang telah mendukung dalam pengerjaan skripsi ini. Semoga skripsi ini dapat memberikan manfaat bagi individu, keluarga, masyarakat dan negara , serta kemajuan dunia pendidikan.

ABSTRAK

Indonesia dikenal sebagai negara yang memiliki hutan tropis dengan kekayaan hayati memiliki lebih dari 30.000 spesies tumbuhan yang tumbuh, di antaranya sebanyak 9.600 spesies telah teridentifikasi memiliki khasiat yang bermanfaat bagi kesehatan manusia. Meskipun demikian, masih terdapat kesalahan dalam mengidentifikasi daun herbal tersebut khususnya daun-daun tanaman herbal karena memiliki ciri yang hampir sama, khususnya bagi masyarakat yang belum memiliki pengetahuan tentang tanaman tersebut. Penelitian dilakukan dengan memanfaatkan metode deep learning dengan model *ResNet-50* untuk mengidentifikasikan daun-daun tanaman herbal. Penelitian ini dilakukan dengan melakukan beberapa tahapan antara lain pengumpulan dataset, augmentasi data, pembuatan model, pelatihan data, dan pengujian. Hasil penelitian menunjukkan bahwa model yang digunakan mencapai tingkat akurasi pelatihan sebesar 99.74% dan tingkat akurasi validasi sebesar 98.75% pada epoch ke-30. Dengan demikian, sistem identifikasi daun herbal berbasis *website* yang dikembangkan dalam penelitian ini diharapkan dapat menjadi alat yang bermanfaat dalam mendukung pemahaman terhadap pemanfaatan daun herbal berkhasiat bagi kesehatan masyarakat.

Kata Kunci : Identifikasi Daun Herbal, *ResNet-50*, Aplikasi Berbasis Web, *Deep Learning*, Kesehatan, Tanaman Herbal.

ABSTRACT

IDENTIFICATION OF HERBAL PLANT LEAVES WITH HEALTH BENEFITS USING WEB-BASED RESNET50 ARCHITECTURE

Indonesia is known as a country that has tropical forests with biological riches that have more than 30,000 species of plants growing, of which 9,600 species have been identified as having properties that are beneficial to human health. However, there are still errors in identifying these herbal plants, especially the leaves of herbal plants because they have almost the same characteristics, especially for people who do not have any knowledge about these plants. The research was carried out using deep learning methods with the ResNet-50 model to classify herbal plant leaves. This research was carried out by carrying out several stages, including dataset collection, data augmentation, model building, data training, and testing. The research results show that the model used achieved a training accuracy level of 99.74% and validation accuracy of 98.75% at the 30th epoch. Thus, the website-based herbal leaf classification system developed in this research is expected to be a useful tool in supporting understanding of the use of herbal plants for their health benefits.

Keywords : Herbal Leaf Classification, ResNet-50, Web Based Application, Deep Learning, Health, Herbal Plants.

DAFTAR ISI

PERSETUJUAN	iii
PERNYATAAN	iv
PENGHARGAAN	v
ABSTRAK.....	vii
ABSTRACT.....	viii
DAFTAR ISI.....	ix
DAFTAR TABEL.....	xi
DAFTAR GAMBAR	xii
BAB 1	1
1.1. Latar Belakang.....	1
1.2. Rumusan Masalah.....	3
1.3. Batasan Masalah	3
1.4. Tujuan Penelitian.....	3
1.5. Manfaat Penelitian	3
1.6. Penelitian Relevan.....	4
1.7. Sistematika Penulisan	5
BAB 2	6
2.1. Daun Herbal.....	6
2.2. <i>ResNet 50</i>	10
2.3. <i>Website</i>	11
2.4. <i>Application Programming Interface (API)</i>	11
BAB 3	13
3.1. Analisis Sistem	13
3.1.1 Analisis Masalah	13
3.1.2 Analisis Kebutuhan	15
3.1.3 Arsitektur Umum Sistem	16
3.2. Pengambilan Dataset	16
3.3. Preprocessing Data	16
3.3.1. Augmentasi Data.....	17

3.3.2	Pemisahan Data Training, Validasi, dan Testing.....	17
3.4.	Pemodelan Sistem.....	17
3.4.1	<i>Use Case Diagram</i>	18
3.4.2	<i>Activity Diagram</i>	18
3.5.	Pelatihan Model	19
3.6.	Pengujian dan Evaluasi.....	20
3.7.	<i>Flowchart</i>	20
3.7.1	<i>Flowchart</i> Sistem	21
3.7.2	<i>Flowchart</i> Algoritma <i>ResNet50</i>	22
3.8.	Perancangan Antarmuka.....	23
3.8.1	Halaman Utama Beranda	23
3.8.2	Halaman Identifikasi	24
3.8.3	Halaman Hasil Identifikasi	24
3.8.4	Halaman Info Data Herbal	25
BAB 4	26
4.1.	Implementasi Sistem.....	26
4.1.1	Perangkat Keras	26
4.1.2	Perangkat Lunak	26
4.2.	Implementasi Dataset.....	27
4.3.	Implementasi Model	28
4.4.	Implementasi Tahap Evaluasi dan Pengujian Model.....	30
4.5.	Hasil Pengujian Sistem	32
4.6.	Implementasi Pada <i>Website</i>	36
4.7.	Faktor Pengaruh Algoritma <i>ResNet50</i>	38
BAB 5	40
5.1.	Kesimpulan	40
5.2.	Saran	40
DAFTAR PUSTAKA	41
LAMPIRAN	A-1

DAFTAR TABEL

Tabel 3. 1. Tabel Pelatihan Model	20
Tabel 4. 1. Daftar Daun Herbal	30
Tabel 4. 2. Daftar Hasil Percobaan.....	32

DAFTAR GAMBAR

Gambar 2. 1. Daun Katuk.....	6
Gambar 2. 2. Daun Kelor	7
Gambar 2. 3. Daun Kemangi.....	7
Gambar 2. 4. Daun Basil	8
Gambar 2. 5. Daun Kari	8
Gambar 2. 6. Daun Salam	9
Gambar 2. 7. Daun Temulawak.....	9
Gambar 2. 8. Daun Kunyit	10
Gambar 2. 9. Ilustrasi Pengolahan Gambar dengan <i>ResNet50</i>	11
Gambar 2. 10. Ilustrasi API.....	12
Gambar 3. 1. Analisis Sistem	13
Gambar 3. 2. Arsitektur Umum Sistem.....	16
Gambar 3. 3. Use Case Diagram	18
Gambar 3. 4. Activity Diagram	19
Gambar 3. 5. Flowchart Sistem	21
Gambar 3. 6. Flowchart Algoritma <i>ResNet50</i>	22
Gambar 3. 7. Gambar Rangka Halaman Utama Beranda.....	24
Gambar 3. 8. Gambar Rangka Halaman Identifikasi	24
Gambar 3. 9. Gambar Rangka Halaman Hasil Identifikasi	25
Gambar 3. 10. Gambar Rangka Halaman Info Daun Herba.....	25
Gambar 4. 1. Target Dataset.....	27
Gambar 4. 2. Jumlah Dataset Tiap Folder.....	28
Gambar 4. 3. Implementasi Model	29
Gambar 4. 4. Grafik Training dan Validation Accuracy	31
Gambar 4. 5. Grafik Training dan Validation Loss.....	31
Gambar 4. 6. Tampilan keluaran model	35
Gambar 4. 7. Tampilan keluaran model	35
Gambar 4. 8. Beranda.....	36
Gambar 4. 9. Halaman Identifikasi dengan Gambar	36

Gambar 4. 10. Halaman Hasil Identifikasi dengan Gambar.....	37
Gambar 4. 11. Halaman Info Daun Herbal.....	37

BAB 1

PENDAHULUAN

1.1. Latar Belakang

Salah satu negara dengan keanekaragaman jenis tanaman herbal terbesar di dunia adalah Indonesia. Hutan tropis di Indonesia adalah rumah bagi sekitar 30.000 spesies tanaman yang berbeda, dan 9.600 di antaranya diketahui mempunyai efek medis yang positif. (Riska et al., 2023). Tanaman herbal memiliki berbagai manfaat untuk kehidupan manusia, mulai dari menjadi bahan makanan, penyedia oksigen, obat-obat tradisional dan terapi medis di bidang kesehatan. (Bella Dwi Mardiana et al., 2023)

Sejak zaman nenek moyang kita, tanaman obat telah digunakan sebagai obat. Sejak saat itu, informasi mengenai pemanfaatan tanaman herbal sebagai obat diturunkan secara turun temurun hingga berdirinya kerajaan pertama Indonesia sekitar tahun 1300 Masehi. Banyak yang beranggapan bahwa bentuk pengobatan alternatif mempunyai dampak yang lebih negatif dibandingkan dengan pengobatan yang berasal dari tanaman herbal, atau bahkan ada. (Purwanto & Sumardi, 2022)

Ada banyak jenis obat herbal dan berbeda-beda tergantung lokasinya, menghasilkan pola “ukuran” dan “bentuk” yang serupa. Tanaman ini mempunyai khasiat obat yang sangat baik mulai dari akar hingga daunnya. (Naeem et al., 2021). Daun herbal merupakan jenis yang sering digunakan masyarakat di bidang kesehatan. Permasalahan yang dihadapi adalah kurangnya pengetahuan tentang jenis-jenis daun herbal dan sulitnya membedakan jenis-jenis daun herbal bagi masyarakat awam yang belum paham tanaman. (Bella Dwi Mardiana et al., 2023)

Sangatlah penting untuk mengetahui secara otomatis berbagai jenis tumbuhan untuk identifikasi tumbuhan mengacu pada ciri-ciri khusus mereka karena terbatasnya jumlah sumber daya dan orang yang berpengetahuan. Satu cara mengidentifikasi tumbuhan adalah melalui identifikasi tumbuhan. (Dali et al., 2018).

Pada penelitian, penulis membangun system Pengembangan Model Identifikasi dengan *ResNet* untuk Mengidentifikasi Daun Tanaman Herbal.

Karena pelatihan pembelajaran mendalam biasanya membutuhkan waktu lama untuk dihitung dan hanya mampu dilakukan pada sejumlah lapisan tertentu, arsitektur *ResNet* diciptakan untuk mengatasi tantangan ini. Model *ResNet* memiliki keunggulan dibandingkan model arsitektur lainnya karena model ini dapat mempertahankan performanya seiring dengan semakin kompleksnya desain. Selain itu, kekuatan pemrosesan menjadi lebih ringan dan kapasitas pelatihan jaringan meningkat. Untuk menunjukkan bahwa model *ResNet* mengungguli model lain dalam klasifikasi gambar dan bahwa fitur gambar diekstraksi secara efektif oleh *ResNet*, koneksi di antara dua hingga tiga lapisan yang mengandung ReLU dan normalisasi batch diteruskan antar arsitektur. (Dewantoro, 2022)

Banyak penelitian yang menggunakan *ResNet*. Berdasarkan penelitian (Awaludin et al., 2022), penyakit pada daun kopi Robusta diklasifikasikan menggunakan Analisis Kinerja *ResNet-50*. Temuan penelitian menunjukkan bahwa meskipun akurasi dalam situasi multikelas hanya 88,98%, akurasi mencapai 92,68% dalam contoh kelas biner.

(Ridhovan et al., 2022) telah melakukan penelitian Penggunaan Metode Jaringan Residual (*ResNet*) untuk Klasifikasi Penyakit pada Daun Gandum. Pengujian menghasilkan tingkat akurasi 98%, rasio data pelatihan-ke-pengujian 90:10, dan nilai Confusion Matrix 0,35, yang menunjukkan bahwa pendekatan *ResNet* mampu mendiagnosis penyakit pada tanaman gandum.

(Miranda et al., 2020) melakukan studi Convolutional Neural Network pada kategorisasi sidik jari menggunakan *Resnet-50*. Dengan tingkat akurasi pelatihan mencapai 99,52%, akurasi validasi sebesar 95,05%, *loss* sebesar 0,016, dan *val_loss* sebesar 0,229, sistem mampu mengenali lima pola sidik jari.

Berdasarkan penelitian terdahulu, pada penelitian ini peneliti melakukan identifikasi daun herbal yang beraneka ragam mulai dari daun katuk, daun kelor, daun saga, daun kari, daun salam, daun kemangi, daun basil, daun jahe, daun kunyit, daun lengkuas dengan pengimplementasian metode *ResNet50*.

1.2. Rumusan Masalah

Banyaknya jenis daun herbal di Indonesia menyebabkan masih banyak masyarakat Indonesia saat ini masih belum dapat membedakan berbagai jenis daun herbal tersebut. Hal ini terkadang menjadi permasalahan tersendiri bagi masyarakat karena kurangnya pengetahuan mengenai daun herbal juga kegunaannya bagi kesehatan. Dengan demikian harus ada sistem yang bisa membantu user dalam mengidentifikasi beraneka ragam daun herbal dan untuk mengetahui manfaat dari daun herbal tersebut bagi kesehatan.

1.3. Batasan Masalah

Berikut Batasan Masalah dalam penelitian ini :

- a. Daun-daun herbal yang akan diidentifikasi seperti daun herbal yang beraneka ragam mulai dari daun katuk, daun kelor, daun kari, daun salam, daun kemangi, daun basil, daun kunyit, dan daun temulawak.
- b. Bahasa yang digunakan untuk membangun program dan model adalah bahasa pemrograman Python.
- c. Arsitektur khusus yang digunakan adalah *ResNet50*.
- d. Sistem diimplementasikan dalam pada aplikasi berbasis *website*.

1.4. Tujuan Penelitian

Tujuan dari penelitian ini adalah membangun sebuah aplikasi berbasis *website* yang dapat melakukan pengidentifikasian daun-daun tanaman herbal dengan menggunakan algoritma *ResNet50*.

1.5. Manfaat Penelitian

Pada penelitian ini diharapkan dapat berguna bagi pengguna/masyarakat agar dapat lebih mengenal jenis daun tanaman herbal dengan lebih baik, dimana daun herbal tersebut dapat digunakan dalam bidang kesehatan.

1.6. Penelitian Relevan

Berikut adalah Penelitian-penelitian terdahulu yang relevan dengan yang akan dilakukan antara lain:

1. Penelitian (Awaludin et al., 2022) “Analisis Kinerja *ResNet-50* dalam Klasifikasi Penyakit pada Daun Kopi Robusta.” Temuan penelitian menunjukkan bahwa meskipun akurasi dalam situasi multikelas hanya 88,98%, akurasi mencapai 92,68% dalam contoh kelas biner.
2. Penelitian oleh (Ridhovan et al., 2022) yang berjudul “Penerapan Metode Residual Network (*ResNet*) Dalam Klasifikasi Penyakit Pada Daun Gandum.” Pengujian menghasilkan tingkat akurasi 98%, rasio data pelatihan-ke-pengujian 90:10, dan nilai Confusion Matrix 0,35, yang menunjukkan bahwa pendekatan *ResNet* mampu mendiagnosis penyakit pada tanaman gandum.
3. "Jaringan Neural Konvolusional dalam Klasifikasi Sidik Jari Menggunakan *Resnet-50*," sebuah studi oleh Miranda et al. (2020). Dengan akurasi pelatihan sebesar 99,52%, akurasi validasi sebesar 95,05%, loss 0,016, dan val_loss 0,229, sistem mampu mengenali lima pola sidik jari.
4. Penelitian (Muneer & Fati, 2020) “*Efficient and Automated Herbs Classification Approach Based on Shape and Texture Features using Deep Learning*” menghasilkan akurasi pelatihan sebesar 99% dan akurasi validasi sebesar 97%.
5. “*Classification of Alzheimer’s Disease with and without Imagery Using Gradient Boosted Machines and ResNet-50*” penelitian oleh (Fulton et al., 2019) menghasilkan akurasi 98,99% set validasi dan 99,34% dari pelatihan dengan 152 epoch.
6. Penelitian berjudul “Implementasi Algoritma *Convolutional Neural Network (Resnet-50)* untuk Klasifikasi Kanker Kulit Benign dan Malignant” oleh (Homepage et al., 2024) mencapai akurasi sebesar 93,88% dan loss sebesar 13,24%, menggunakan dengan epoch 50 menggunakan *optimizer* Adam dan *batch size* sebesar 54.
7. “*Classification of Pneumonia Cell Images Using Improved ResNet50 Model*” penelitian oleh (Çınar et al., 2021) menghasilkan akurasi 97,22% dengan menggunakan 3.730 gambar pneumonia dan 1.341 gambar biasa.

1.7. Sistematika Penulisan

Berikut merupakan sistematika penulisan skripsi ini, yang terbagi ke dalam beberapa bagian utama, sebagaimana dijelaskan berikut ini.

BAB 1 PENDAHULUAN

Bab ini memuat uraian tentang segala latar belakang informasi yang berkenaan dengan pokok bahasan yang akan diteliti, meliputi rumusan masalah, tujuan penelitian, batasan masalah, manfaat penelitian, metodologi penelitian, dan sistematika penulisan.

BAB 2 LANDASAN TEORI

Tinjauan teoritis yang berkaitan dengan *ResNet50* dan Daun Herbal dan juga yang berhubungan dengan penelitian akan dibahas pada bab ini.

BAB 3 ANALISIS PERANCANGAN

Melakukan analisis terhadap masalah dan sistem yang dibangun akan dibahas pada bab ini. Setelah itu, kita akan melanjutkan dengan desain sistem, dengan menggunakan Algoritma *ResNet 50*.

BAB 4 IMPLEMENTASI DAN PENGUJIAN SISTEM

Bab ini akan membahas implementasi dan pengujian sistem menggunakan tahapan analisis dan desain yang telah dibahas sebelumnya.

BAB 5 PENUTUP

Kesimpulan dan Saran dari penelitian yang telah selesai akan dimasukkan dalam bab ini untuk kemajuan penelitian di masa depan.

BAB 2

LANDASAN TEORI

2.1. Daun Herbal

Tanaman herbal adalah tanaman yang daun, batang, dan akarnya dapat dimanfaatkan untuk tujuan pengobatan. Masyarakat memanfaatkan beberapa tanaman herbal tersebut sebagai terapi komplementer atau alternatif. Selain itu, penggunaan pengobatan herbal juga mempunyai manfaat, seperti tidak adanya efek negatif dan kemampuan anggota keluarga untuk mengelola sendiri pengobatannya. (Yulianto, 2017).

Daun herbal merupakan salah satu jenis yang sering digunakan masyarakat di bidang kesehatan. Permasalahan yang dihadapi adalah kurangnya pengetahuan tentang jenis-jenis daun herbal dan sulitnya membedakan jenis-jenis daun herbal bagi masyarakat awam yang belum paham tanaman. (Bella Dwi Mardiana et al., 2023)

Terdapat delapan Jenis daun herbal yang dimanfaatkan pada penelitian kali ini adalah :



Gambar 2. 1. Daun Katuk

Selain segudang manfaat kesehatan lainnya, daun katuk memiliki sifat antibakteri, antijamur, antikanker, dan antilipidemik. Masyarakat Indonesia dan sejumlah negara lain sudah mengenal dan sering mengonsumsi katuk. Selain dimakan sebagai sayur, katuk juga digunakan sebagai obat herbal untuk melancarkan produksi ASI (ASI) dan sebagai obat untuk membantu menurunkan berat badan. (Prof. Dr.Ir. Urip Santoso, 2016)



Gambar 2. 2. Daun Kelor

Daun kelor adalah obat cepat untuk kondisi peradangan seperti peradangan kelenjar, sakit kepala, dan bronkitis. Daunnya bermanfaat bagi pasien yang menderita insomnia dan mengobati luka. Kelor digunakan secara luas dalam industri kosmetik saat ini, dan dalam sejarah Mesir kuno, kelor juga digunakan untuk menyiapkan salep kulit. Manfaat lain dari daun kelor, memiliki kualitas anti-inflamasi yang dapat membantu menurunkan stres oksidatif, yang dapat membantu penyembuhan luka dan mengurangi peradangan kulit. Daun kelor kaya akan polifenol tanaman, yang menjadikannya tinggi antioksidan dan memberikan manfaat anti-inflamasi. Kandungan vitamin A dan C, yang keduanya bermanfaat bagi kulit, dan penelitian juga menunjukkan mereka dapat membantu memperlambat proses penuaan.



Gambar 2. 3. Daun Kemangi

Daun kemangi memiliki daun kecil berwarna hijau cerah dengan aroma yang khas dan rasa yang segar. Kemangi adalah sumber nutrisi penting bagi tubuh kita. Daun ini mengandung vitamin A, vitamin C, vitamin K, kalsium, zat besi, magnesium, kalium, dan serat. Masing-masing nutrisi ini diperlukan agar tubuh tetap sehat secara keseluruhan. Ciri-ciri kemangi yang paling menonjol adalah manfaat kesehatannya. Pengobatan tradisional telah menggunakan daun ini untuk menyembuhkan berbagai penyakit. Beberapa manfaat kesehatan kemangi termasuk meningkatkan sistem

kekebalan tubuh, melawan peradangan, menjaga kesehatan jantung, mengontrol gula darah, meningkatkan kepadatan tulang.



Gambar 2. 4. Daun Basil

Daun basil berwarna hijau dan bentuknya bulat telur, namun bentuknya juga tergantung dari varietasnya. Manfaat tanaman basil dari sisi kesehatan dapat menjadi anticancer, antioxidant, antimicrobial, antibacterial, antiinflammatory dan antyhipertensive. (Thalib et al., 2023)



Gambar 2. 5. Daun Kari

Daun kari mengandung senyawa antioksidan yang dapat membantu dalam mempertahankan tubuh dari radikal bebas berbahaya yang merusak sel. Daun kari juga mengandung alkaloid karbazol yang memiliki sifat anti-inflamasi dan antikanker. Daun kari juga mengandung kalsium, vitamin A dan C, serta nutrisi lainnya. Daun kari memiliki banyak manfaat bagi Kesehatan diantaranya sebagai obat antiinflamasi, antidiabetes, antimikroba, antioksidan, dan hipertensi. (Eva Sulistiany et al., n.d.)



Gambar 2. 6. Daun Salam

Di antara banyak manfaat daun salam bagi kesehatan adalah kemampuannya menurunkan dislipidemia, khususnya hipertrigliseridemia. (Setya Hardhani, n.d.), Menurunkan kadar LDL. Berdasarkan peningkatan dosis, Karena daun salam memiliki komponen aktif seperti quercetin, daun salam berpotensi menurunkan kadar kolesterol LDL serum secara drastis, suatu flavonoid yang juga merupakan antioksidan yang dapat menghambat sekresi Apo-B100 ke dalam usus sehingga menurunkan jumlah Apo. B . (Warta et al., n.d.), Selain digunakan sebagai bumbu kuliner oleh masyarakat Indonesia, daun salam memiliki kegunaan sebagai terapi, seperti mengobati asam urat.(Warta et al., n.d.)



Gambar 2. 7. Daun Temulawak

Kualitas anti inflamasi pada daun temulawak dapat membantu mengurangi peradangan dan nyeri. Sifat antiseptiknya juga membantu mencegah infeksi pada luka, Ekstrak daun temulawak diketahui dapat membantu merangsang produksi empedu dan enzim pencernaan, yang mendukung proses pencernaan dan penyerapan nutrisi, Daun temulawak dapat membantu membersihkan racun dari hati dan meningkatkan fungsi hati secara keseluruhan, dan mengurangi kolestrol.



Gambar 2. 8. Daun Kunyit

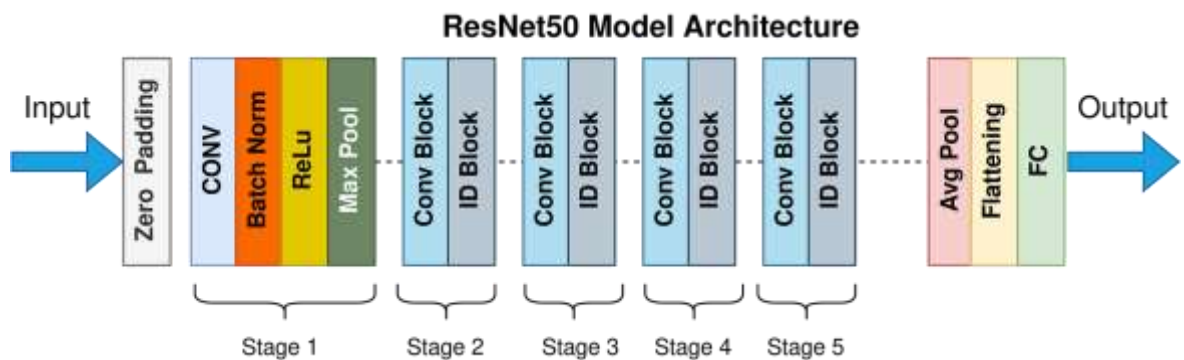
Manfaat daun kunyit bagi Kesehatan diantaranya adalah untuk mencegah terbentuknya karies gigi, mengatasi penyakit asam lambung, menurunkan tekanan darah, meredakan radang sendi, memelihara kesehatan jantung.

2.2. *ResNet50*

Menggunakan model yang telah dilatih sebelumnya, *ResNet* adalah sejenis arsitektur *Convolution Neural Network* (CNN). *ResNet* dapat menghemat waktu karena menghilangkan persyaratan untuk membuat data pelatihan dari awal. (Ridhovan et al., 2022). Dengan 50 lapisan, sesuai dengan namanya, model *ResNet50* relatif lebih dalam dibandingkan desain CNN sebelumnya seperti VGG atau AlexNet. Ini terdiri dari banyak blok sisa, atau blok penyusun, yang masing-masing memiliki beberapa lapisan konvolusional. Ada tiga bagian utama arsitektur :

1. **Lapisan Konvolusional:** Untuk mengekstrak karakteristik tingkat rendah dari gambar masukan, lapisan pertama *ResNet50* menjalankan operasi konvolusi konvensional. Lapisan ini membantu pembelajaran jaringan tentang pola dasar seperti tekstur dan tepian.
2. **Blok Sisa:** Sejumlah blok sisa membentuk inti *ResNet50*. Setiap blok memiliki fungsi aktivasi ReLU, normalisasi batch, dan dua atau lebih lapisan konvolusional. Jaringan dapat mempelajari sisa data di berbagai tingkat abstraksi berkat blok-blok ini.
3. **Lapisan Terhubung Sepenuhnya:** Probabilitas kelas akhir dihasilkan oleh lapisan terhubung penuh dengan aktivasi softmax

yang berada setelah lapisan pengumpulan rata-rata global di dekat akhir jaringan. (LP2M, 2023).



Gambar 2. 9. Ilustrasi Pengolahan Gambar dengan *ResNet50*

2.3. Website

Sebuah *website* terdiri dari beberapa halaman web yang terhubung ke file lain yang relevan. Halaman beranda merupakan salah satu halaman pada sebuah *website*. Ketika seseorang mengunjungi sebuah situs web, mereka awalnya melihat halaman beranda. Mengklik hyperlink di halaman beranda memungkinkan pengguna untuk menavigasi ke halaman situs web lain. (Hendrianto, 2013)

2.4. Application Programming Interface (API)

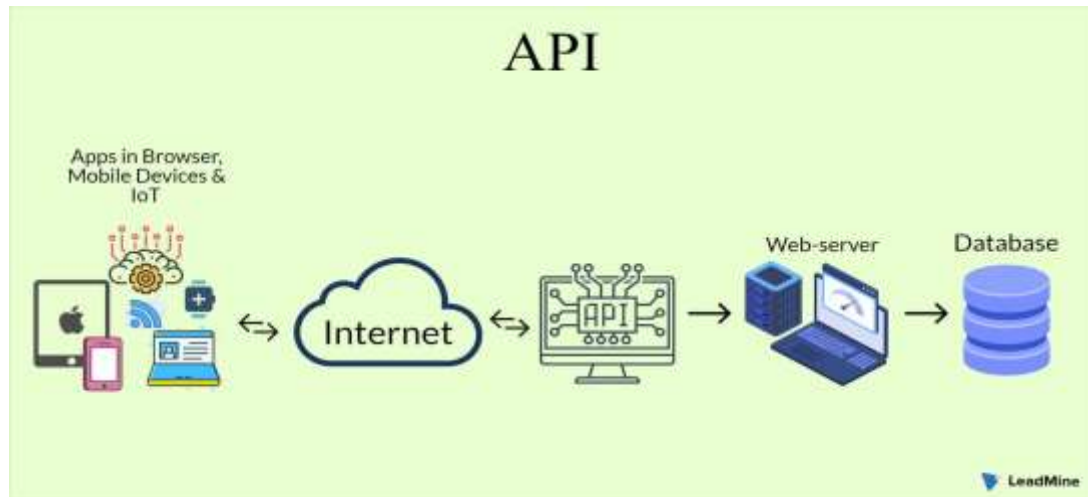
Application Programming Interface (API) adalah sekelompok pedoman dan perintah yang memungkinkan *software* untuk saling berinteraksi satu sama lain. API bertindak sebagai penghubung untuk mentransfer informasi antara dua atau lebih program.

Dengan adanya API, para pengembang tidak perlu repot untuk mempelajari aplikasi lain secara terperinci. Pengembang dapat langsung melakukan integrasi fungsionalitas dari aplikasi lain ke dalam aplikasi yang hendak dibuat.

API dapat terdiri atas beberapa instruksi, protokol komunikasi, atau juga beberapa fungsi dari sebuah *library*. API dapat digunakan untuk berbagai tujuan seperti mengakses halaman *website*, dan integrasi dengan *platform* lainnya.

API dapat digunakan sebagai sarana untuk mendistribusikan *Open Data*. Seperti contoh, apabila seseorang ingin memanipulasi data tanpa harus mengubah halaman *website* yang dimiliki maka dapat digunakan API. Hal ini seakan-akan pengembang

menitipkan sebagian potongan situs *website* pada API. Untuk menampilkan data seperti itu tidak diperlukan manipulasi halaman *website* pengembang. Hanya perlu menyisipkan potongan kecil dari *website* lain untuk meminta data seperti itu. (Sugimoto, 2019)



Gambar 2. 10. Ilustrasi API

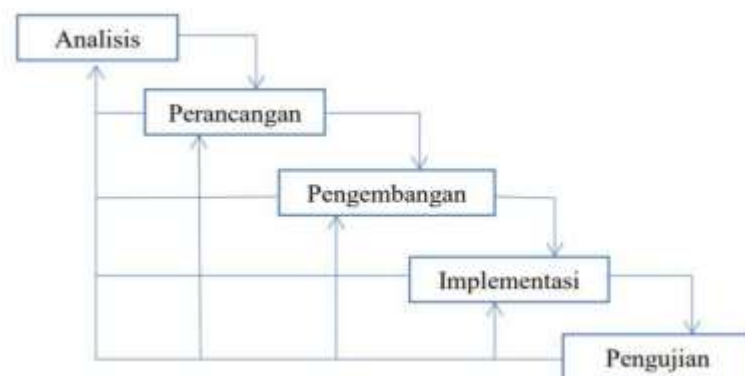
BAB 3

ANALISIS DAN PERANCANGAN SISTEM

3.1. Analisis Sistem

Analisis sistem juga dianggap sebagai pendekatan pemecahan masalah yang melibatkan pembongkaran sistem menjadi bagian-bagian penyusunnya. Hal ini dilakukan untuk memantau seberapa baik kinerja masing-masing komponen serta bagaimana mereka berinteraksi untuk membantu sistem secara keseluruhan mencapai tujuannya.

Biasanya, analisis sistem dilakukan pada tingkat perencanaan sistem (system design). Salah satu dari sekian banyak proses dalam proses rekayasa pemecahan masalah adalah perancangan sistem, yang menggabungkan berbagai komponen pembentuk untuk menciptakan suatu sistem secara keseluruhan.



Gambar 3. 1. Analisis Sistem

3.1.1 Analisis Masalah

Dalam penelitian, analisis masalah bertujuan untuk mengetahui penyebab dari permasalahan kemudian dilakukan pengkajian lebih dalam pada masalah yang akan diselesaikan sehingga tercipta sistem yang efektif. Dalam penelitian ini, permasalahan yang dianalisis adalah bagaimana algoritma *ResNet50* dapat membantu pengguna dalam mengidentifikasi jenis-jenis daun herbal sehingga dapat membantu Masyarakat dalam mengetahui fungsi dan kegunaan dari daun herbal tersebut.

"Dalam analisis masalah pada skripsi identifikasi daun herbal menggunakan *ResNet50*, fokus utama adalah pada evaluasi kinerja model dalam membedakan antara berbagai jenis daun herbal dengan akurasi yang tinggi. Penelitian ini juga mencakup identifikasi potensi masalah seperti overfitting atau underfitting yang dapat mempengaruhi kinerja model, serta evaluasi terhadap faktor-faktor lain seperti kualitas dataset, keberagaman kelas, dan teknik pemrosesan gambar yang digunakan."

1. Mengapa identifikasi daun herbal penting?
 - Daun herbal memiliki berbagai manfaat kesehatan, seperti efek antiinflamasi, antimikroba, antikanker, dan lain-lain. Identifikasi yang tepat memastikan bahwa Anda menggunakan tanaman dengan sifat yang diinginkan untuk kondisi kesehatan tertentu. Tanaman yang berbeda dapat memiliki profesi kimia yang berbeda dan menghasilkan efek yang berbeda pada tubuh manusia.
2. Mengapa *ResNet50* dipilih sebagai metode untuk identifikasi daun herbal?
 - *ResNet50* adalah salah satu arsitektur jaringan saraf konvolusional (CNN) yang telah terbukti memberikan kinerja yang sangat baik dalam tugas klasifikasi gambar. Dengan menggunakan *ResNet50*, kita dapat memanfaatkan keunggulan kinerja model ini dalam mengenali pola yang kompleks dan mempelajari fitur-fitur penting dari gambar-gambar daun tanaman herbal. *ResNet50* memiliki kedalaman yang cukup untuk menangkap fitur-fitur yang penting dalam gambar-gambar yang kompleks tanpa terlalu banyak parameter, sehingga memungkinkan pelatihan yang efisien bahkan dengan keterbatasan sumber daya komputasi.
3. Mengapa diperlukan studi kasus untuk menguji pendekatan ini dalam situasi nyata?
 - Dengan adanya studi kasus, akan memberikan pemahaman lebih mendalam tentang *ResNet50*. Dengan situasi nyata, pengguna diharapkan mampu mengidentifikasi dan mengetahui manfaat daun herbal bagi keberadaan sehari-hari, terutama dalam bidang medis.

3.1.2 Analisis Kebutuhan

Analisis kebutuhan adalah langkah penting untuk mengenali data dan prosedur yang diperlukan dalam merancang sistem. Dalam proses perancangan, analisis kebutuhan dapat mencakup persyaratan fungsional dan non-fungsional yang perlu dibuat agar sistem yang dikembangkan dapat mencapai tujuannya.

3.1.2.1 Kebutuhan Fungsional

Tindakan yang harus dilakukan sistem untuk mencapai tujuannya semuanya dianggap sebagai persyaratan fungsional. Kebutuhan Fungsional yang diterapkan dalam penelitian dan sistem ini adalah :

1. Sistem dapat membaca *input* gambar dari pengguna.
2. Sistem dapat mengenali jenis daun herbal ketika dilakukan proses identifikasi oleh model.
3. Sistem dapat memberikan hasil seperti manfaat daun herbal tersebut dalam kehidupan pengguna untuk pemahaman pengguna lebih lanjut.

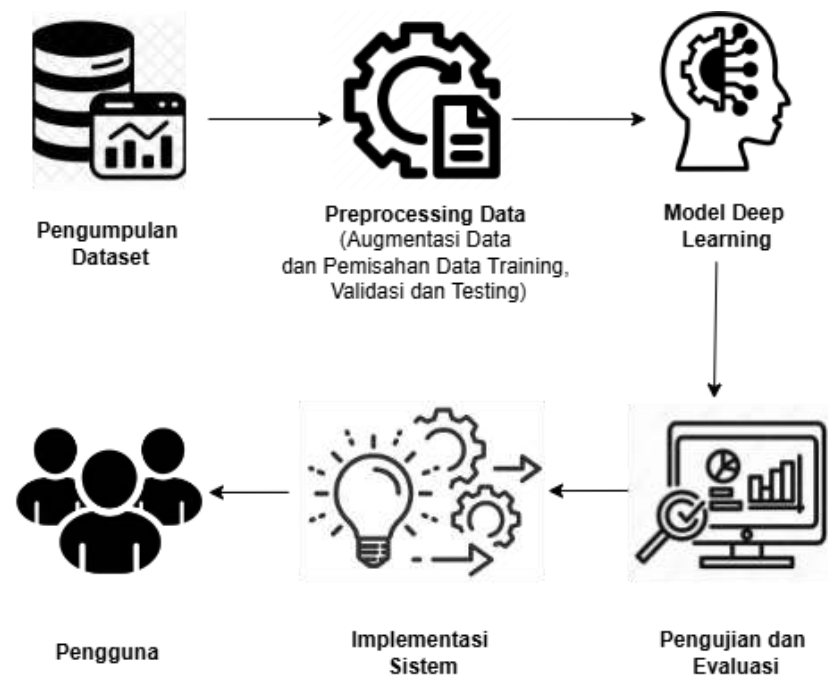
3.1.2.1 Kebutuhan Non-Fungsional

Kebutuhan non-fungsional adalah persyaratan yang mencakup fitur, atribut, pembatasan layanan, atau aspek-aspek lain dari sistem seperti batasan waktu, kendala proses pengembangan, dan standar sistem yang berfungsi sebagai pelengkap. Di bawah ini terdapat daftar kebutuhan fungsional yang diperlukan dalam sistem ini:

1. Batasan ekstensi gambar yang dapat digunakan sebagai *input* adalah .JPG, .PNG, dan .JPEG.
2. Batasan daun herbal yang digunakan adalah, daun katuk, daun kelor, daun kemangi, daun basil, daun kari, daun salam, daun temulawak, dan daun kunyit.
3. *User Interface* pada sistem akan mudah dipahami oleh pengguna.
4. Tidak membutuhkan biaya yang besar dan tidak memerlukan perangkat tambahan.
5. Internet diperlukan untuk terhubung dengan sistem.

3.1.3 Arsitektur Umum Sistem

Arsitektur sistem umum merupakan gambaran visual tentang bagaimana sistem akan dibangun.



Gambar 3. 2. Arsitektur Umum Sistem

3.2. Pengumpulan Dataset

Tahap awal pada penelitian ini dengan melakukan pengambilan dataset dari Kaggle <https://www.kaggle.com/datasets/anefiamutiaraatha/dataset-tanaman-herbal> dan juga google yang berupa 8 jenis daun herbal, yaitu Daun Kari, Daun Katuk, Daun Kelor, Daun Basil, Daun Kemangi, Daun Kunyit, Daun Salam, dan Daun Temulawak dengan jumlah masing-masing daun sebanyak 100 daun.

3.3. Preprocessing Data

Tahap selanjutnya dengan melakukan pra-pemrosesan data yang bertujuan agar model yang dibangun dapat mempelajari data dengan baik. Beberapa sub-bagian dalam tahap pra-pemrosesan data meliputi, Augmentasi data dan Pembagian Data: Training, Validation, dan Testing

3.3.1. Augmentasi Data

Augmentasi data adalah teknik untuk memperbesar jumlah dataset dengan membuat modifikasi dari dataset yang sudah ada. Hal ini berguna untuk mengatasi masalah overfitting, terutama ketika dataset asli berukuran kecil. Pada program kali ini augmentasi dilakukan dengan memutar gambar sebesar 45 derajat dan menyimpan hasilnya dimana dalam setiap folder, kode menghitung jumlah gambar yang ada dan menambahkan gambar baru hingga mencapai target jumlah gambar yang diinginkan.

3.3.2 Pemisahan Data Training, Validasi, dan Testing

Setelah augmentasi, langkah selanjutnya adalah memisahkan data menjadi set pelatihan sebesar 75%, validasi sebesar 20%, dan pengujian sebesar 5%. Ini dilakukan menggunakan *Image Data Generator* dari Keras.

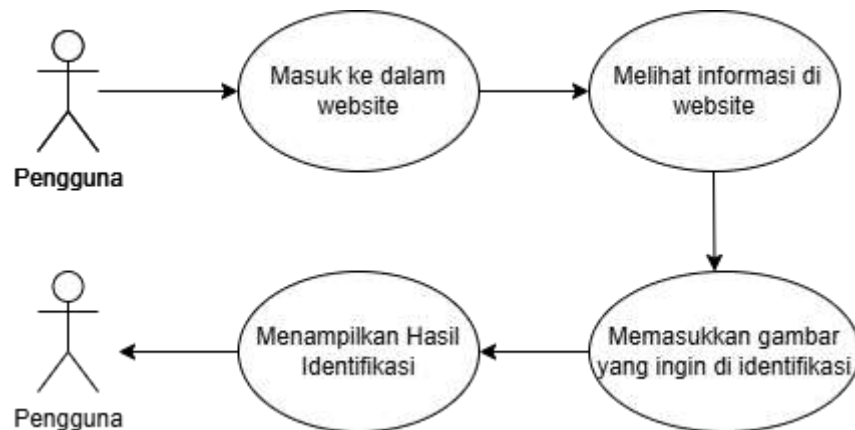
Di sini, penulis menggunakan beberapa parameter untuk melakukan augmentasi seperti rescaling, shear range, zoom range, rotation range, width shift range, height shift range, horizontal flip, vertical flip, dan validation split. Generator ini digunakan untuk memuat gambar dari direktori, melakukan augmentasi, dan membagi data menjadi batch untuk pelatihan, validasi, dan pengujian.

3.4. Pemodelan Sistem

Pemodelan sistem adalah proses menguraikan langkah-langkah interaksi antara pengguna dengan aplikasi yang dibangun, sehingga sistem dapat beroperasi secara efisien. Biasanya, pemodelan sistem diwujudkan dalam format *Unified Modelling Language* (UML). UML adalah bahasa pemodelan yang umum digunakan untuk menggambarkan hubungan antara komponen-komponen dalam sistem, sehingga memungkinkan interaksi melalui pengguna. Dalam penelitian ini, *Diagram use case*, *diagram aktivitas*, dan *diagram urutan* adalah tiga jenis model UML yang digunakan.

3.4.1 Use Case Diagram

Menurut *Unified Modeling Language* (UML), diagram *use case* adalah jenis diagram yang digunakan untuk menggambarkan bagaimana aktor berinteraksi dengan suatu sistem atau aplikasi. Aktor adalah entitas luar yang berperan ketika berinteraksi dengan perangkat keras, sistem lain, atau pengguna manusia. Diagram use case yang digunakan adalah sebagai berikut :

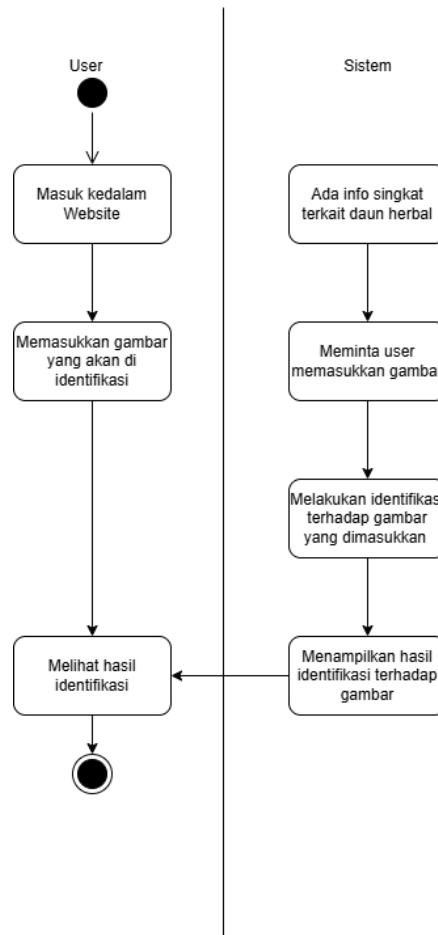


Gambar 3. 3. Use Case Diagram

Gambar 3.3. menunjukkan interaksi pengguna dengan sistem. Pengguna akan memulai interaksi dengan masuk aplikasi dan dapat langsung masuk ke inti program. Pengguna dapat memilih Input Gambar dimana dengan memasukkan sebuah gambar kedalam aplikasi maka akan dilakukan identifikasi terhadap gambar dan dalam beberapa saat akan ditampilkan hasilnya. *User* dapat melihat identifikasi gambar daun berdasarkan gambar yang diberikan.

3.4.2 Activity Diagram

Activity Diagram adalah representasi visual dari urutan alur kerja atau proses kegiatan dalam sistem, dimulai dari langkah awal hingga mencapai langkah akhir. *Activity Diagram* juga berfungsi untuk menggambarkan bagian-bagian komponen yang ada dalam *Use Case Diagram*. Berikut *Activity Diagram* yang dipakai dalam sistem:



Gambar 3. 4. Activity Diagram

Gambar 3.4. menunjukkan *activity diagram* sistem. Pertama *user* membuka halaman aplikasi dan sistem, dan User akan diminta memasukkan gambar. *User* kemudian melakukan *input* gambar dan sistem akan menampilkan hasil identifikasi tersebut kepada *User*.

3. 5. Pelatihan Model

Setelah melakukan augmentasi data dan memisahkan data menjadi data training, validasi, dan testing, langkah berikutnya adalah melatih model deep learning menggunakan data yang telah diproses.

Model ini menggunakan *ResNet50* yang telah dilatih sebelumnya di dataset ImageNet. *ResNet50* merupakan arsitektur deep learning yang sangat kuat dan efisien untuk klasifikasi gambar. Model dilatih selama 30 epoch. Satu epoch adalah satu kali iterasi penuh melalui dataset training.

Tabel 3. 1. Tabel Pelatihan Model

Epoch	Accuracy	Loss	Val_Acc	Val_Loss
8	0.9760	0.0716	0.9729	0.0628
9	0.9786	0.0593	0.9917	0.0491
10	0.9839	0.0450	0.9688	0.1094
11	0.9885	0.0356	0.9792	0.0581
12	0.9797	0.0535	0.9854	0.0307
13	0.9891	0.0347	0.9854	0.0360
...
29	0.9958	0.0146	0.9917	0.0330
30	0.9974	0.0116	0.9875	0.0252

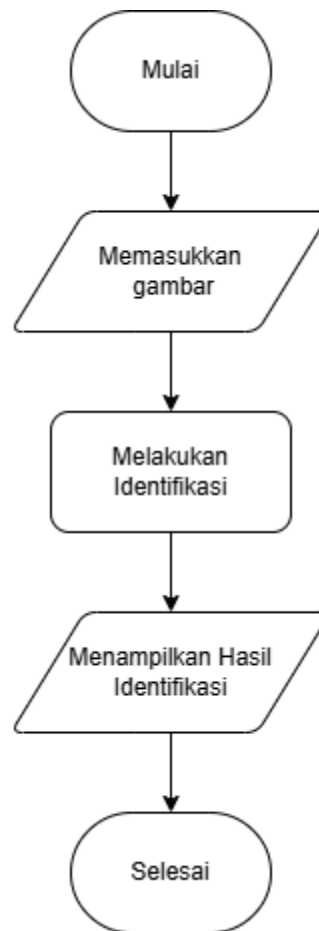
3.6. Pengujian dan Evaluasi

Setelah proses pelatihan model selesai dilakukan, kemudian dilakukan tahap evaluasi dan pengujian model. Pada penelitian ini, evaluasi dilakukan pada saat proses pelatihan dengan data latih (train set) dan data validasi (validation set). Sedangkan tahap pengujian model dilakukan menggunakan data uji (test set) dengan cara memprediksi data uji.

3.7. Flowchart

Flowchart (Diagram Alir) adalah representasi visual dari serangkaian langkah-langkah yang dijalankan oleh sebuah algoritma secara teratur. Setiap tahapan prosesnya direpresentasikan oleh simbol-simbol yang berbeda dan dilengkapi dengan penjelasan untuk setiap tahapnya.

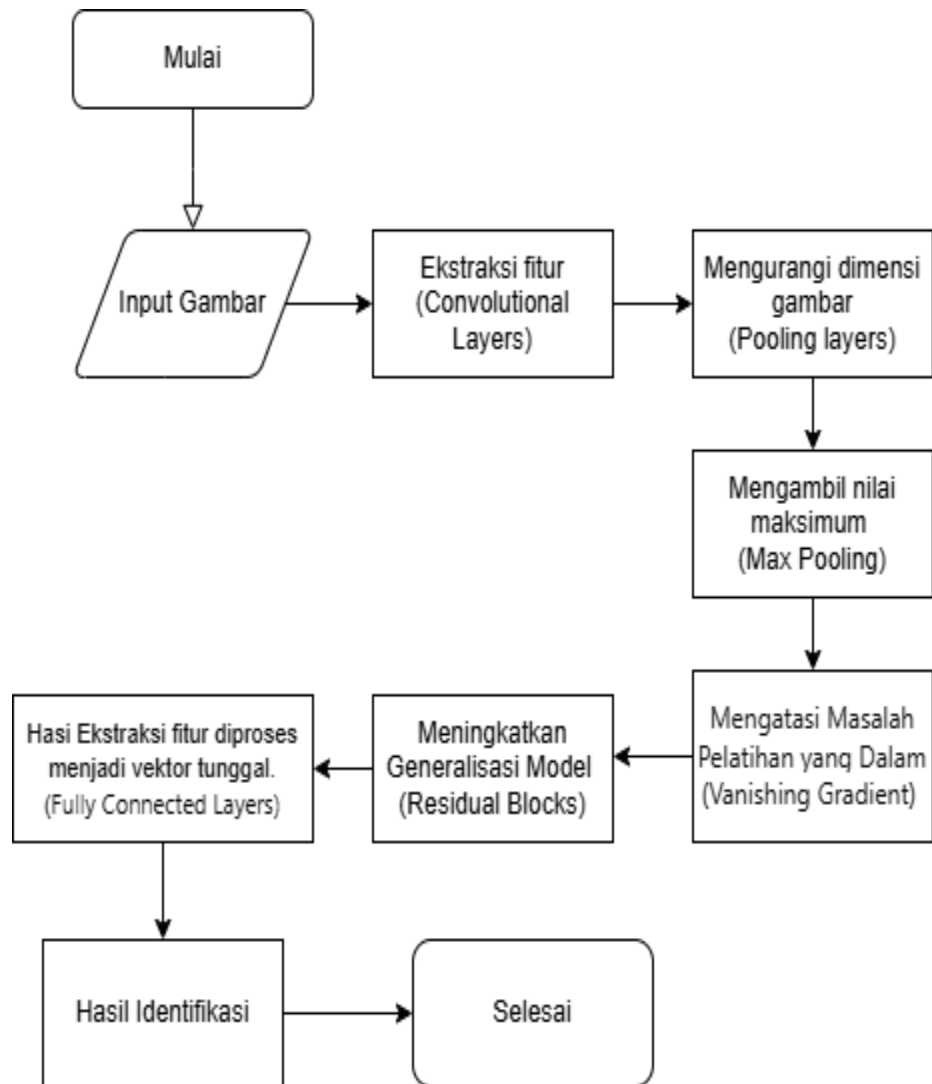
3.7.1 Flowchart Sistem



Gambar 3. 5. Flowchart Sistem

Pada gambar 3.5. dapat dilihat alur kerja dari sistem yang pertama yakni *user* memasukkan gambar yang ingin diidentifikasi. Kemudian Model akan melakukan identifikasi dan akan mengembalikan *response* berdasarkan hasil identifikasi kepada *user*, kemudian *website* menampilkan hasil dan informasi relevan terhadap identifikasi yang dilakukan oleh model.

3.7.2 Flowchart Algoritma ResNet50



Gambar 3. 6. Flowchart Algoritma *ResNet50*

Pada gambar 3.6. dapat dilihat bagaimana cara kerja algoritma *ResNet50*.

1. **Mulai:** Proses dimulai
2. **Input Gambar:** Gambar dimasukkan oleh *user* sebagai input ke dalam model.
3. **Ekstraksi fitur (*Convolutional Layers*):** Ekstraksi fitur dilakukan oleh model *ResNet50* yang sudah dilatih sebelumnya (*pre-trained model*).
4. **Mengurangi dimensi gambar (*Pooling layers*):** Lapisan pooling digunakan untuk mengurangi dimensi gambar sambil mempertahankan

fitur-fitur penting dan pada program yang digunakan adalah *Global Average Pooling*.

5. **Mengambil nilai maksimum (*Max Pooling*):** Proses *max pooling* diterapkan untuk memilih nilai maksimum dari setiap wilayah fitur yang dipool.
6. **Mengatasi Masalah Pelatihan yang Dalam (*Vanishing Gradient*):** Menggunakan model pre-trained *ResNet50*, menggunakan ReLU sebagai fungsi aktivasi dalam lapisan dense, dan memanfaatkan arsitektur residual ini untuk menghindari vanishing gradient.
7. **Meningkatkan Generalisasi Model (*Residual Blocks*):** Blok residual digunakan untuk meningkatkan kemampuan generalisasi model dan memperbaiki aliran gradien selama pelatihan.
8. **Hasil Ekstraksi fitur diproses menjadi vektor tunggal (*Fully Connected Layers*):** Lapisan fully connected (juga dikenal sebagai dense layers) digunakan untuk klasifikasi akhir setelah fitur diekstraksi oleh bagian konvolusi dari model. Dalam kode ini, lapisan fully connected ditambahkan setelah Global Average Pooling dan Flatten layer.
9. **Hasil Identifikasi:** Model menghasilkan identifikasi akhir berdasarkan fitur-fitur yang telah diproses dan dapat mengidentifikasi 8 jenis daun herbal.
10. **Selesai:** Proses berakhir.

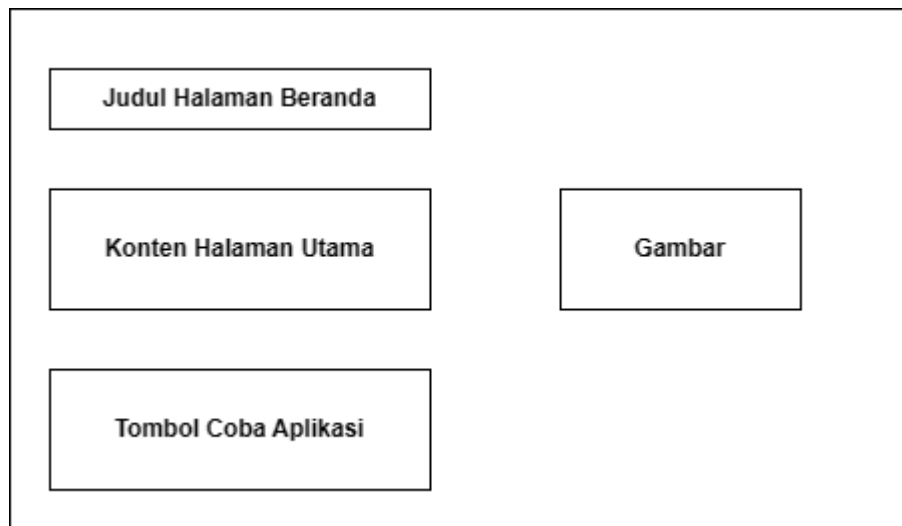
3.8. Perancangan Antarmuka

Perancangan *Interface* adalah proses menciptakan kerangka desain sistem yang dikonstruksi. Perancangan *Interface* adalah langkah yang diperlukan untuk memastikan bahwa proses pembuatan sistem dapat berfungsi lebih efektif berdasarkan panduan yang telah didefinisikan dalam desain antarmuka tersebut.

3.8.1 Halaman Utama Beranda

Halaman utama beranda atau homepage *website* berisi tampilan informasi *website* secara umum yang muncul ketika pengguna mengakses website ini. Struktur halaman utama beranda atau homepage *website* terdiri dari judul halaman utama beranda, konten

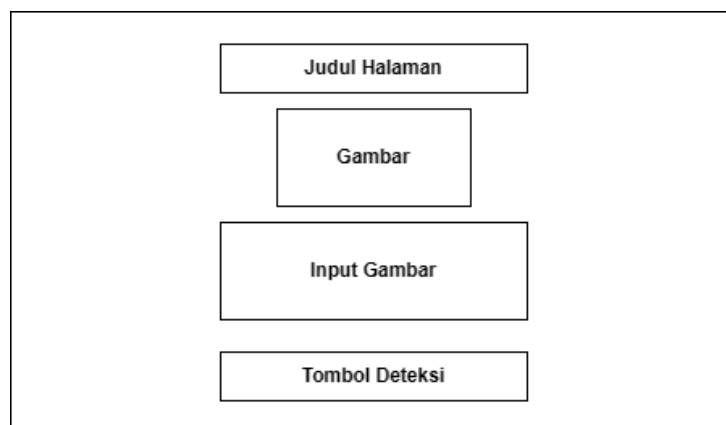
halaman utama beranda, dan footer. Tata letak (layout) dan tampilan gambar rangka (wireframe) halaman utama beranda *website* dapat dilihat pada Gambar 3.7.



Gambar 3. 7. Gambar Rangka Halaman Utama Beranda

3.8.2 Halaman Identifikasi

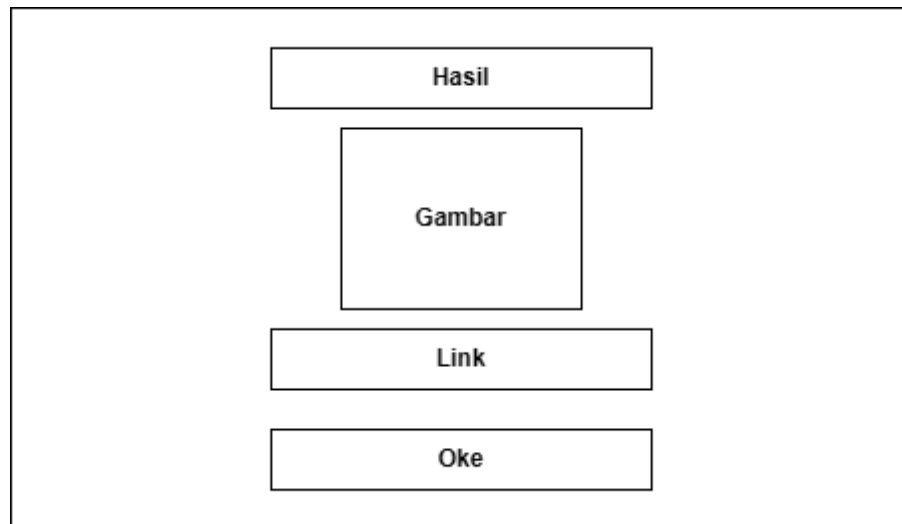
Halaman Identifikasi *website* berisi tampilan input gambar pengguna terkait daun tanaman herbal.



Gambar 3. 8. Gambar Rangka Halaman Identifikasi

3.8.3 Halaman Hasil Identifikasi

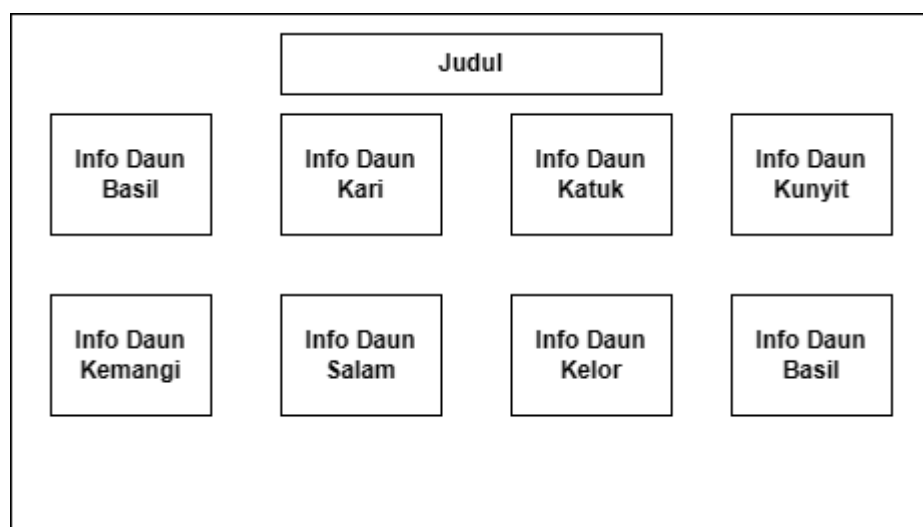
Halaman Hasil Identifikasi *website* berisi tampilan pop up yang memunculkan hasil identifikasi dan link menuju *website* penjelasan manfaat daun herbal.



Gambar 3. 9. Gambar Rangka Halaman Hasil Identifikasi

3.8.4 Halaman Info Data Herbal

Halaman Info Data Herbal *website* berisi tampilan dan informasi terkait daun herbal dan juga manfaat nya bagi kesehatan.



Gambar 3. 10. Gambar Rangka Halaman Info Daun Herba

BAB 4

IMPLEMENTASI DAN PENGUJIAN SISTEM

4.1. Implementasi Sistem

Pada penelitian ini, sistem dibangun dengan memanfaatkan algoritma *ResNet50*. Sistem dibangun dengan dua komponen yaitu *front-end* dan *back-end*. Bagian *front-end* dibangun dengan bantuan *bootstrap* dan *back-end* dibangun menggunakan bantuan bahasa pemrograman *python*. Sistem juga dibantu menggunakan aplikasi seperti *visual studio code*.

4.1.1 Perangkat Keras

1. Intel(R) Core(TM) i7-1165G7 @ 2.80GHz
2. NVIDIA GeForce MX450
3. 16 GB RAM

4.1.2 Perangkat Lunak

1. Implementasi Pengambilan Dataset
 - Kaggle
2. Implementasi pra-pemrosesan dataset
 - Google Colab
 - Library
3. Implementasi membangun, melatih, dan mengevaluasi model
 - Google Colab
 - Library
4. Implementasi model ke aplikasi berbasis *website*
 - Visual Studio Code
 - Bahasa : HTML, CSS, JavaScript, Python
 - Framework : Bootstrap, Flask

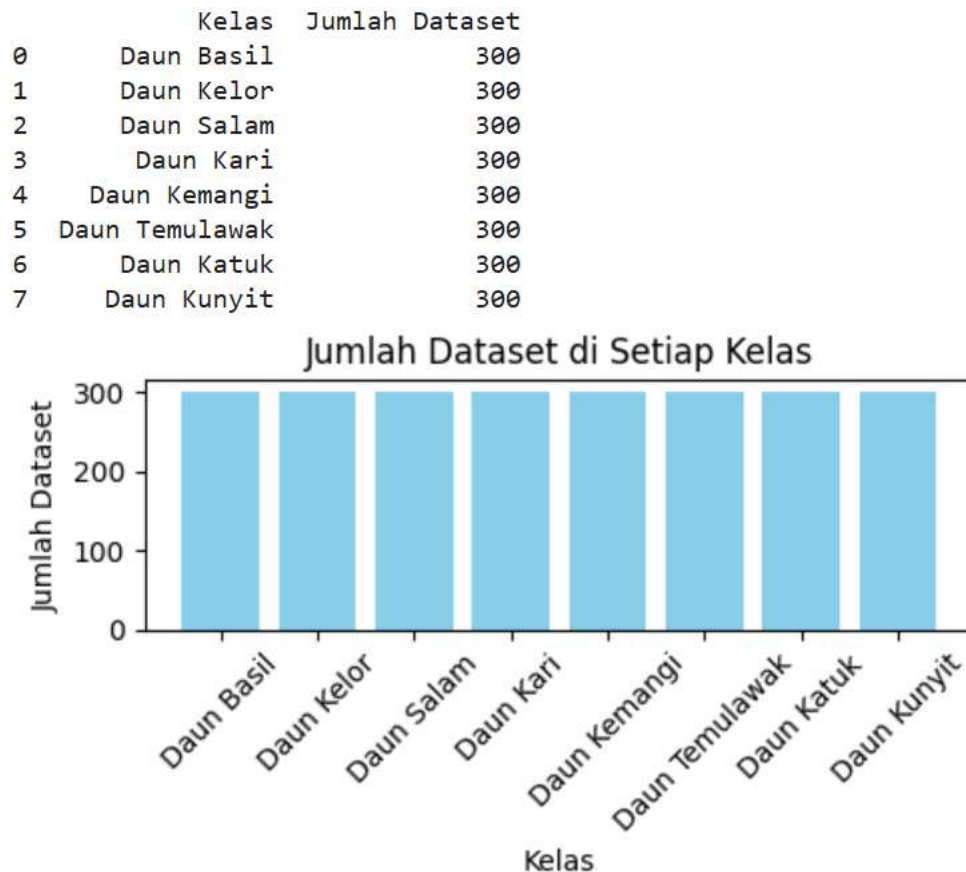
4.2. Implementasi Dataset

Dataset yang digunakan diambil dari Kaggle Dimana terdapat 8 jenis daun yaitu daun basil, daun kari, daun katuk, daun kelor, daun kemangi, daun kunyit, daun salam, dan daun temulawak. Dataset yang digunakan awalnya berjumlah 100 gambar per jenis daun, yang berarti ada 800 gambar total. Kemudian, dilakukan augmentasi data yang membuat dataset bertambah menjadi 2400 gambar Dimana pembagiannya adalah 300 gambar per folder.

```
Folder /content/gdrive/MyDrive/Herbal/DataTraining/Daun Basil sudah memenuhi target.  
Folder /content/gdrive/MyDrive/Herbal/DataTraining/Daun Kelor sudah memenuhi target.  
Folder /content/gdrive/MyDrive/Herbal/DataTraining/Daun Salam sudah memenuhi target.  
Folder /content/gdrive/MyDrive/Herbal/DataTraining/Daun Kari sudah memenuhi target.  
Folder /content/gdrive/MyDrive/Herbal/DataTraining/Daun Kemangi sudah memenuhi target.  
Folder /content/gdrive/MyDrive/Herbal/DataTraining/Daun Temulawak sudah memenuhi target.  
Folder /content/gdrive/MyDrive/Herbal/DataTraining/Daun Katuk sudah memenuhi target.  
Folder /content/gdrive/MyDrive/Herbal/DataTraining/Daun Kunyit sudah memenuhi target.
```

Gambar 4. 1. Target Dataset

Hal pertama yang dilakukan kepada dataset adalah augmentasi data. Gambar 4.4. memperlihatkan bahwa setiap dataset sudah memenuhi target yang diinginkan sesuai permintaan pada kode augmentasi.



Gambar 4. 2. Jumlah Dataset Tiap Folder

Gambar 4.3. menunjukkan bahwa setiap folder memiliki jumlah dataset yang sama, dan hasil nya ditampilkan dalam bentuk grafik batang.

4.3. Implementasi Model

Setelah tahap pra-pemrosesan data, selanjutnya adalah tahap membangun model. Tahap ini dilakukan menggunakan library TensorFlow dan Keras dengan arsitektur umum jaringan saraf tiruan yang terdiri dari tiga lapisan: lapisan masukan, lapisan keluaran, dan lapisan tersembunyi. Arsitektur model terdiri dari sebuah lapisan masukan (input layer), lapisan tersembunyi (hidden layer) dengan 3 buah dense layer dan juga 3 buah dropout layer masing-masing setelah dense layer, serta sebuah lapisan keluaran (output layer). Detail arsitektur lapisan jaringan saraf tiruan, jumlah unit neuron, dan fungsi aktivasi yang digunakan dapat dilihat pada Gambar 4.4.

Layer (type)	Output Shape	Param #
resnet50 (Functional)	(None, 7, 7, 2048)	23587712
global_average_pooling2d (GlobalAveragePooling2D)	(None, 2048)	0
flatten (Flatten)	(None, 2048)	0
dense (Dense)	(None, 1024)	2098176
dense_1 (Dense)	(None, 512)	524800
classification (Dense)	(None, 8)	4104
Total params: 26214792 (100.00 MB)		
Trainable params: 26161672 (99.80 MB)		
Non-trainable params: 53120 (207.50 KB)		

Gambar 4. 3. Implementasi Model

Dari gambar 4.4. dapat dilihat bahwa model memiliki parameter dimana :

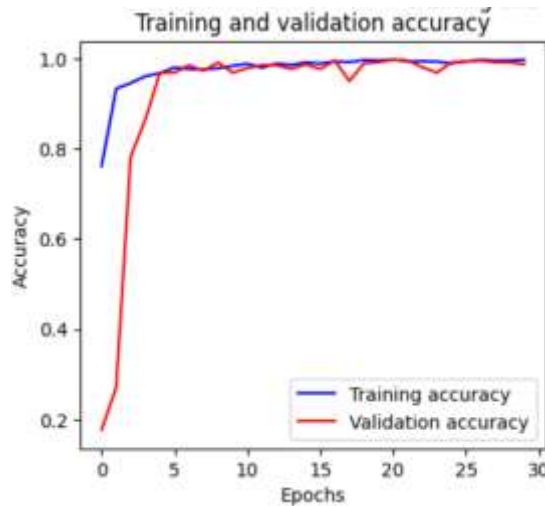
- **Total Parameter ResNet** = 23587712.
- **Dense(1024, activation="relu"):**
 Jumlah parameter: (jumlah input + 1) * jumlah neuron
 Jumlah parameter: (2048 + 1) * 1024 = 2098176
- **Dense(512, activation="relu"):**
 Jumlah parameter: (1024 + 1) * 512 = 524800
- **Dense(8, activation="softmax", name="classification"):**
 Jumlah parameter: (512 + 1) * 8 = 4,104
- **Total parameter** = 23,587,712 + 2,098,176 + 524,800 + 4,104 = 26214792

4.4. Implementasi Tahap Evaluasi dan Pengujian Model

Implementasi Tahap Evaluasi dan Pengujian Model adalah tahapan yang dilakukan setelah tahapan implementasi model berhasil dilakukan. Tujuannya adalah untuk menunjukkan bahwa sistem yang dikembangkan dapat berfungsi dengan baik, dalam mengidentifikasi daun tanaman herbal. Pada tahap ini, terdapat jenis daun herbal. Jenis daun herbal yang digunakan adalah, Daun Basil, Daun Kelor, Daun Salam, Daun Kari, Daun Kemangi, Daun Temulawak, Daun Katuk dan Daun Kunyit. Berikut adalah tabel yang akan menampilkan Daftar Daun Herbal.

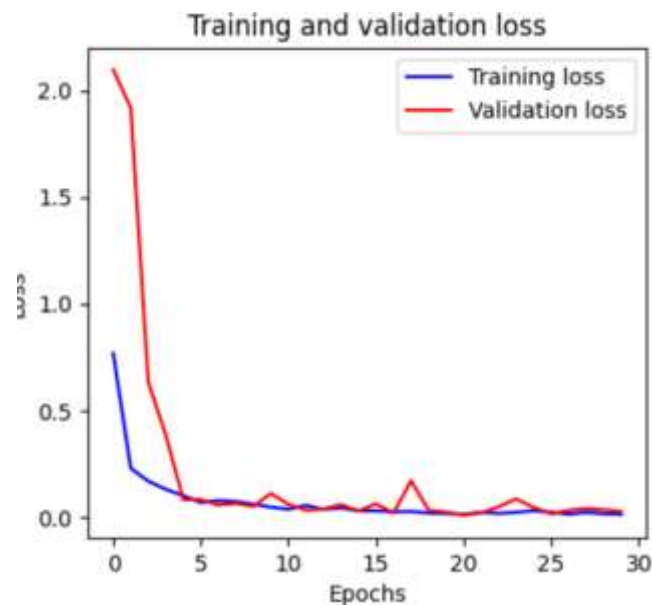
Tabel 4. 1. Daftar Daun Herbal

<p>Daun Basil</p> 	<p>Daun Kunyit</p> 	<p>Daun Kari</p> 	<p>Daun Katuk</p> 
<p>Daun Kelor</p> 	<p>Daun Salam</p> 	<p>Daun Temulawak</p> 	<p>Daun Kemangi</p> 



Gambar 4. 4. Grafik Training dan Validation Accuracy

Pada gambar 4.5. terdapat grafik yang dihasilkan berdasarkan akurasi latihan dan validasi data. Kedua garis terlihat saling berdampingan pada bagian atas grafik. Hal ini menunjukkan bahwa model dapat melakukan identifikasi yang baik dengan data latihan serta data yang baru pertama kali diperlihatkan.











Gambar 4. 5. Grafik Training dan Validation Loss

Pada gambar 4.6. terdapat grafik yang dihasilkan berdasarkan *loss* latihan dan validasi. Hal ini mengindikasikan bahwa model mempelajari pola data dengan baik pada saat latihan dan dapat mempelajari data baru dengan baik pula pada validasi.

4.5. Hasil Pengujian Sistem

Setelah dilakukan pengujian sistem, berikut hasil yang didapat dengan mencoba semua daun yang telah dipaparkan diatas:

Tabel 4. 2. Daftar Hasil Percobaan

<i>Inputan</i>	<i>Output</i>
	 Daun Basil  <small>Klik di sini untuk informasi lebih lanjut</small> 
	 Daun Kari  <small>Klik di sini untuk informasi lebih lanjut</small> 



Daun Temulawak



[Klik di sini untuk informasi lebih lanjut](#)

Ok









Daun Salam

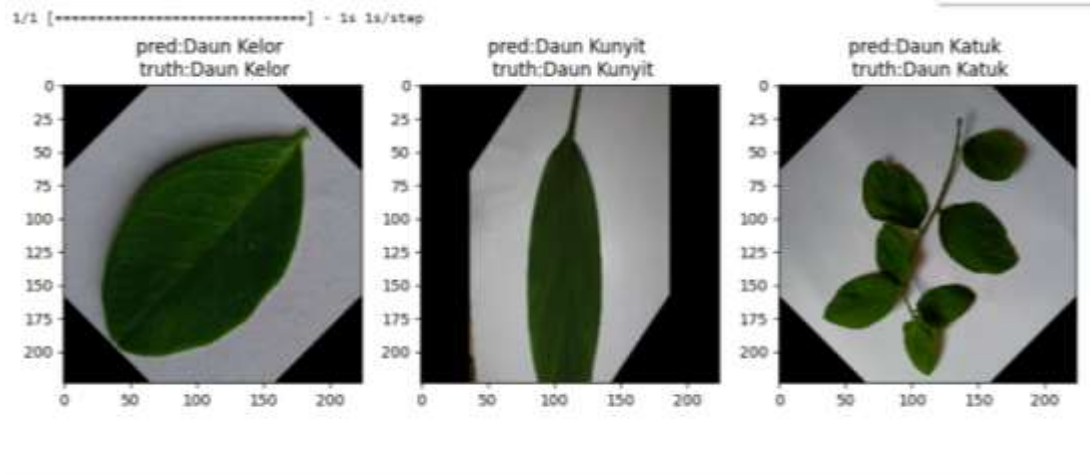


[Klik di sini untuk informasi lebih lanjut](#)

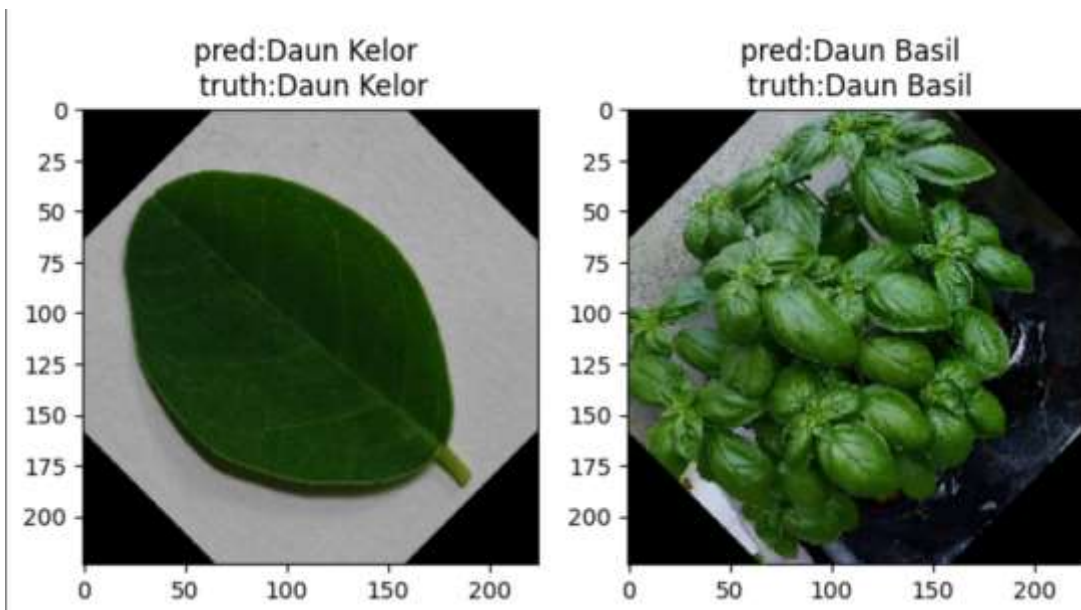
Ok

	 <p>Daun Katuk</p>  <p>Klik di sini untuk informasi lebih lanjut</p> <p>Ok</p>
	 <p>Daun Kemangi</p>  <p>Klik di sini untuk informasi lebih lanjut</p> <p>Ok</p>

Selain dilakukan pengujian pada *website*, ada dilakukan pengujian pada model juga yang menghasilkan hasil sebagai berikut :



Gambar 4. 6. Tampilan keluaran model



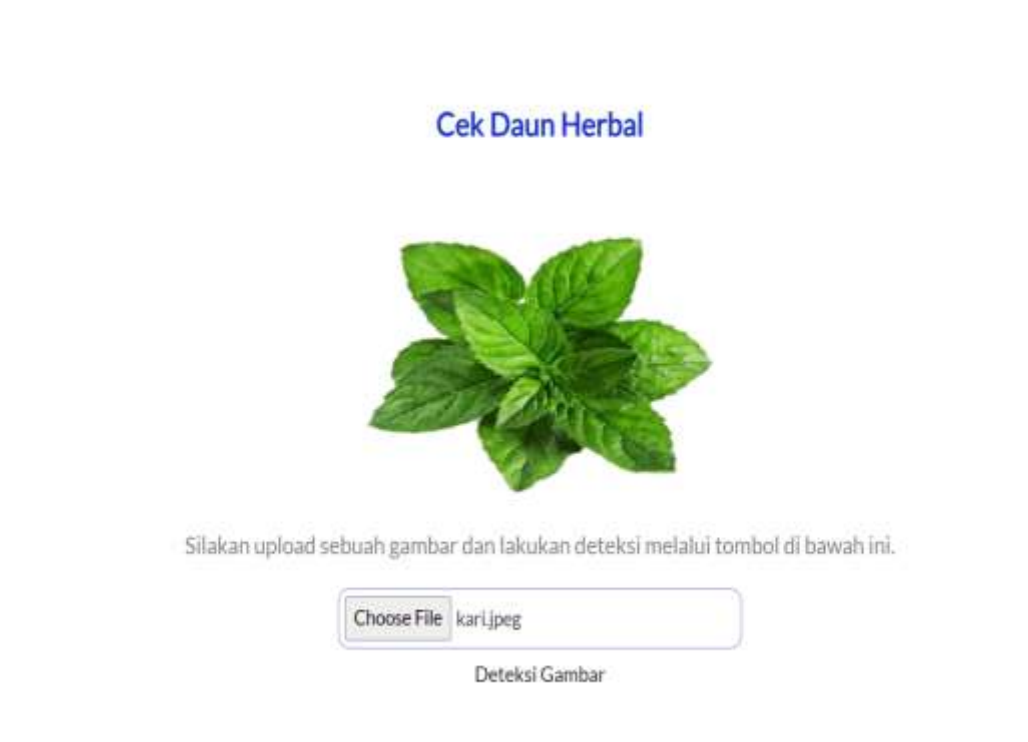
Gambar 4. 7. Tampilan keluaran model

Pada gambar 4.6. terdapat hasil keluaran identifikasi gambar yang dilakukan. Dapat dilihat, model berhasil mengidentifikasi data baru yang diberikan dengan akurasi yang tinggi. Hal ini berarti model telah dilatih dengan baik dan siap melakukan identifikasi untuk data baru yang belum pernah dilihat lainnya.

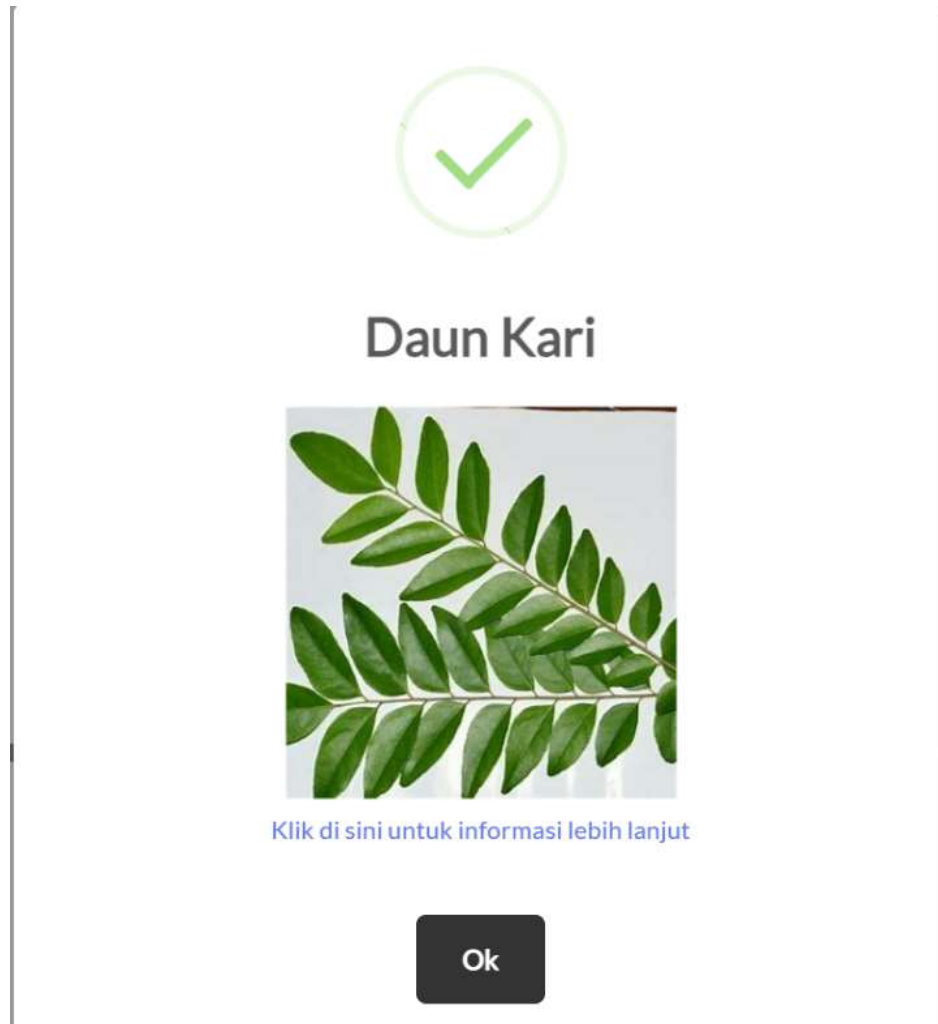
4.6. Implementasi Pada Website



Gambar 4. 8. Beranda



Gambar 4. 9. Halaman Identifikasi dengan Gambar



Gambar 4. 10. Halaman Hasil Identifikasi dengan Gambar



Gambar 4. 11. Halaman Info Daun Herbal

4.7. Faktor Pengaruh Algoritma *ResNet50*

Algoritma *ResNet50* adalah salah satu arsitektur deep learning yang terkenal untuk tugas-tugas pengenalan gambar dan klasifikasi. Kompleksitas dan kemampuan *ResNet50* dalam mempelajari pola pada gambar dipengaruhi oleh sejumlah elemen utama. Berikut adalah faktor-faktor tersebut:

1. **Arsitektur Jaringan :** *ResNet50* adalah jaringan yang dalam, terdiri dari 50 lapisan (layers). Jaringan ini menggunakan "residual blocks" yang membantu mengatasi masalah vanishing gradients dalam jaringan yang sangat dalam. Jumlah lapisan dan penggunaan residual blocks menambah kompleksitas perhitungan namun meningkatkan kemampuan jaringan dalam mempelajari pola yang lebih kompleks.
2. **Jumlah Parameter :** Jumlah parameter dalam *ResNet50* sangat besar, mencakup jutaan bobot dan bias yang harus dipelajari selama pelatihan. Banyaknya parameter ini memungkinkan jaringan untuk memiliki kapasitas yang tinggi untuk mempelajari representasi yang kompleks, tetapi juga menambah beban komputasi dan memori.
3. **Ukuran Input :** Ukuran input gambar yang digunakan untuk melatih *ResNet50* mempengaruhi kompleksitas perhitungan. Ukuran standar input untuk *ResNet50* adalah 224x224 piksel. Gambar dengan resolusi lebih tinggi memerlukan lebih banyak perhitungan pada setiap lapisan jaringan.
4. **Data Latih :** Jumlah dan variasi data latih yang digunakan sangat mempengaruhi kemampuan *ResNet50*. Dataset besar dan beragam memungkinkan jaringan untuk mempelajari fitur-fitur yang lebih umum dan kompleks. Namun, dataset besar juga membutuhkan lebih banyak waktu dan sumber daya komputasi untuk pelatihan.
5. **Kecepatan dan Ketersediaan Perangkat Keras :** Kecepatan pelatihan dan kompleksitas perhitungan sangat dipengaruhi oleh perangkat keras yang digunakan. GPU (Graphics Processing Unit) sangat meningkatkan kecepatan pelatihan dibandingkan dengan CPU (Central Processing Unit). Ketersediaan perangkat keras yang kuat memungkinkan pelatihan jaringan dalam yang lebih efisien.

6. **Teknik Regulasi** : Teknik regularisasi seperti dropout, batch normalization, dan data augmentation digunakan untuk menghindari overfitting dan memperbaiki generalisasi model. Teknik ini menambah beberapa lapisan perhitungan tambahan selama pelatihan, tetapi meningkatkan kualitas model yang dihasilkan.
7. **Hyperparameter Tuning** : Pemilihan hyperparameter seperti learning rate, batch size, dan jumlah epochs sangat mempengaruhi kompleksitas dan hasil pelatihan. Hyperparameter yang diatur dengan baik dapat mengoptimalkan kinerja model, sedangkan hyperparameter yang tidak sesuai dapat menyebabkan pelatihan menjadi lebih lambat dan kurang efektif.

BAB 5

KESIMPULAN DAN SARAN

5.1. Kesimpulan

Berdasarkan hasil pengujian dan implementasi yang dijabarkan dalam Bab 4, Ada beberapa kesimpulan yang dapat penulis buat, khususnya :

1. Pada penelitian kali ini, model dapat melakukan identifikasi terhadap 8 jenis daun herbal dengan baik.
2. Augmentasi dilakukan pada penelitian kali ini membuat adanya peningkatan pada jumlah data gambar yang akan ditraining dan juga di validasi.
3. Model mencapai akurasi yang sangat tinggi, dengan nilai akurasi validasi mencapai lebih dari 98% pada epoch terakhir. Hal ini menunjukkan bahwa model *ResNet-50* sangat efektif dalam mengidentifikasikan daun-daun herbal yang dapat bermanfaat dalam bidang kesehatan dan penelitian herbal.

5.2. Saran

Berikut beberapa saran dan rekomendasi yang penulis dapat berikan dari hasil penelitian ini untuk pengembangan sistem selanjutnya, yaitu:

1. Disarankan untuk memperbanyak dataset yang digunakan dalam pelatihan model dengan lebih banyak jenis daun herbal dan variasi gambar dalam berbagai kondisi pencahayaan, sudut pandang, dan latar belakang. Hal ini akan membantu meningkatkan generalisasi model dan akurasinya dalam situasi nyata.
2. Menambahkan fitur tambahan seperti pencarian berdasarkan gambar, rekomendasi daun herbal berdasarkan gejala kesehatan, dan pelaporan serta analisis data pengguna untuk memperkaya fungsi aplikasi.
3. Mengembangkan versi *mobile* dari aplikasi untuk meningkatkan aksesibilitas dan kenyamanan penggunaan oleh masyarakat luas, terutama bagi pengguna yang membutuhkan identifikasi daun herbal di lapangan.

DAFTAR PUSTAKA

- Awaludin, I., Fadhil, M., Andhika Zaini Zulfikor, M., Negeri Bandung Jl Gegerkalong Hilir, P., Parongpong, K., Bandung Barat, K., Barat, J., & Artikel, I. (2022). Analisis Kinerja ResNet-50 dalam Klasifikasi Penyakit pada Daun Kopi Robusta. *JURNAL INFORMATIKA*, 9(2). <http://ejournal.bsi.ac.id/ejurnal/index.php/ji>
- Bella Dwi Mardiana, Wahyu Budi Utomo, Ulfah Nur Oktaviana, Galih Wasis Wicaksono, & Agus Eko Minarno. (2023). Herbal Leaves Classification Based on Leaf Image Using CNN Architecture Model VGG16. *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, 7(1), 20–26. <https://doi.org/10.29207/resti.v7i1.4550>
- Çınar, A., Yıldırım, M., & Eroğlu, Y. (2021). Classification of pneumonia cell images using improved ResNet50 model. *Traitement Du Signal*, 38(1), 165–173. <https://doi.org/10.18280/TS.380117>
- Dali, A. D. A., Omar, N. A., & Mustapha, A. (2018). Data mining approach to herbs classification. *Indonesian Journal of Electrical Engineering and Computer Science*, 12(2), 570–576. <https://doi.org/10.11591/ijeecs.v12.i2.pp570-576>
- Dewantoro, R. W. , A. S. Y. N. , & R. R. A. (2022). *Dewantoro, R. W., Analysis Of Right And Wrong Use Of Mask Based On Deep Learning*.
- Eva Sulistiany, dan, Keperawatan Poltekkes Kemenkes Aceh Desa Paya Bujuk Beuromo, J., Langsa Barat Kota langsa, K., Kunci, K., & Kari, D. (n.d.). Efektifitas Pemberian Ekstrak Daun Kari (*Murraya Koenigii*) Terhadap Penurunan Tekanan Darah Pada Penderita Hipertensi di Desa Seuriget Kecamatan Langsa Barat Langsa Tahun 2016 *The Effectiveness Of Giving Curry Leaves Extract (murraya koenigii) To Lower Blood Pressure Of Hypertensive People In Seuriget Village, West Langsa District, Aceh In 2016*.
- Fulton, L. V., Dolezel, D., Harrop, J., Yan, Y., & Fulton, C. P. (2019). *Classification of alzheimer's disease with and without imagery using gradient boosted machines and resnet-50*. *Brain Sciences*, 9(9). <https://doi.org/10.3390/brainsci9090212>

- Hendrianto, D. E. (2013). Pembuatan sistem informasi perpustakaan berbasis website pada sekolah menengah pertama negeri 1 donorojo kabupaten pacitan.
- Homepage, J., Putra Hafi Puja Gusti, G., Haerani, E., Syafria, F., Yanto, F., Kurnia Gusti, S., Studi Teknik Informatika, P., & Sains dan Teknologi, F. (2024). *MALCOM: Indonesian Journal of Machine Learning and Computer Science Implementation of Convolutional Neural Network Algorithm (ResNet-50) for Benign and Malignant Skin Cancer Classification Implementasi Algoritma Convolutional Neural Network (Resnet-50) untuk Klasifikasi Kanker Kulit Benign dan Malignant*. 4(3), 984–992. <https://doi.org/10.57152/malcom.v4i3.1398>
- Muneer, A., & Fati, S. M. (2020). Efficient and automated herbs classification approach based on shape and texture features using deep learning. *IEEE Access*, 8, 196747–196764. <https://doi.org/10.1109/ACCESS.2020.3034033>
- Naeem, S., Ali, A., Chesneau, C., Tahir, M. H., Jamal, F., Sherwani, R. A. K., & Hassan, M. U. (2021). The classification of medicinal plant leaves based on multispectral and texture feature using machine learning approach. *Agronomy*, 11(2). <https://doi.org/10.3390/agronomy11020263>
- Prof. Dr.Ir. Urip Santoso, M. Sc. (2016). Katuk, Tumbuhan Multi Khasiat.
- Purwanto, & Sumardi. (2022). 328-1170-1-PB.
- Ridhovan, A., Suharso, A., Fakultas,), Komputer, I., Karawang, S., Ronggo Waluyo, J. H., Timur, T., & Karawang, K. (2022). Penerapan Metode *Residual Network (ResNet)* Dalam Klasifikasi Penyakit Pada Daun Gandum.
- Riska, A., Purnawansyah, Darwis, H., & Astuti, W. (2023). *IJCS_12_3+Riska*.
- Setya Hardhani, A. (n.d.). *The Effect of Orally Administered Bay Leaves (Eugenia polyantha) extract on Triglyceride Serum Level in Hyperlipidemic Male Wistar Rats*.
- Sugimoto, G. (2019). *Introduction to Populating a Website with API Data*.
- Thalib, T., Menne, F., Purwasetiawatik, Titin. F., Ningsi, F., Primanthoro, R., & Zahra, F. (2023). PKM Peningkatan Pemahaman Olahan Daun Basil Sebagai Produk Pangan Alternatif Bernilai Ekonomis di Moncobalang Desa Bontosunggu . *Jurnal Hasil Pengabdian Masyarakat Vol. 4, No. 2 (2023)* .

- Warta, L., Harismah, K., Chusniatun, D., Kimia, J. T., & Tarbiyah, J. (n.d.).
Pemanfaatan Daun Salam (*eugenia polyantha*) Sebagai Obat Herbal dan
Rempah Penyedap Makanan.
- Yulianto, S. (2017). Penggunaan Tanaman Herbal Untuk Kesehatan *Susilo
Yulianto*.

LAMPIRAN KODINGAN

1. Model.ipynb

```
from google.colab import drive
drive.mount('/content/gdrive')

Mounted at /content/gdrive

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from keras.preprocessing.image import ImageDataGenerator
from tensorflow.keras.utils import load_img
from tensorflow.keras.utils import img_to_array
from tensorflow.keras.utils import array_to_img
from tensorflow.keras.layers import Input, Dense, Flatten, GlobalAveragePooling2D, Dropout, BatchNormalization
from tensorflow.keras.models import Sequential

import glob, os, random, zipfile
import tensorflow as tf

train_data_directory = '/content/drive/MyDrive/Herbal/DataTraining'
```

```
base_path = os.path.join('/content/gdrive/MyDrive/Herbal/DataTraining')

#img_list = glob.glob(os.path.join(base_path, '**/*.jpg' + '**/*.jpeg'))
#print(len(img_list))

from os.path import join
from glob import glob

img_list = []
for ext in ('**/*.jpg', '**/*.jpeg', '**/*.png', '**/*.JPG'):
    img_list.extend(glob(join(base_path, ext)))

print(len(img_list))
```

```
plt.figure(figsize=(12, 8))
for i, img_path in enumerate(random.sample(img_list, 5)):
    if not os.path.exists(img_path):
        continue

    img = load_img(img_path)
    img = img_to_array(img, dtype=np.uint8)

    plt.subplot(2, 3, i+1)
    plt.imshow(img.squeeze())
```

```

import os
import random
from PIL import Image

# Path ke direktori dataset
base_path = '/content/gdrive/MyDrive/Herbal/DataTraining'

# Fungsi untuk melakukan augmentasi data pada suatu gambar
def augment_image(image, save_path):
    # Rotasi gambar sebesar 45 derajat
    rotated_image = image.rotate(45)
    # Simpan gambar yang sudah dirotasi
    rotated_image.save(save_path)

# Daftar nama folder kelas
class_folders = [
    'Daun Basil',
    'Daun Kelor',
    'Daun Salam',
    'Daun Kari',
    'Daun Kemangi',
    'Daun Temulawak',
    'Daun Katuk',
    'Daun Kunyit'
]

```

```

# Jumlah target gambar per folder
target_count_per_folder = 300

# Augmentasi data untuk setiap folder (kelas)
for folder_name in class_folders:
    folder_path = os.path.join(base_path, folder_name)
    image_files = [f for f in os.listdir(folder_path) if os.path.isfile(os.path.join(folder_path, f))]

    # Jumlah gambar dalam folder saat ini
    current_count = len(image_files)

    # Jumlah tambahan gambar yang diperlukan
    additional_count = target_count_per_folder - current_count
    if additional_count < 0:
        additional_count = 0

    # Jika jumlah gambar dalam folder sudah mencapai atau melebihi target, lanjut ke folder berikutnya
    if additional_count == 0:
        print(f"Folder {folder_path} sudah memenuhi target.")
        continue

    # Augmentasi data dengan menambahkan gambar baru
    for i in range(additional_count):
        # Pilih file gambar secara acak dari yang sudah ada
        random_image = random.choice(image_files)
        image_path = os.path.join(folder_path, random_image)

```

```

# Baca gambar
image = Image.open(image_path)

# Buat nama baru untuk gambar augmented
new_image_name = f"augmented_{i + 1}_{random_image}"
new_image_path = os.path.join(folder_path, new_image_name)

# Augmentasi gambar dan simpan
augment_image(image, new_image_path)

print(f"Folder {folder_path} ditambahkan {additional_count} gambar.")

```

```

train_datagen = tf.keras.preprocessing.image.ImageDataGenerator(
    rescale=1./255,
    shear_range=0.2,
    zoom_range=0.2,
    rotation_range=0.2,
    width_shift_range=0.2,
    height_shift_range=0.2,
    horizontal_flip=True,
    vertical_flip=True,
    validation_split=0.2
)

test_datagen = ImageDataGenerator(
    rescale=1./255,
    validation_split=0.05
)

train_generator = train_datagen.flow_from_directory(
    base_path,
    target_size=(224, 224),
    batch_size=8,
    class_mode='categorical',
    subset='training',
    seed=0
)

```

```

validation_generator = train_datagen.flow_from_directory(
    base_path,
    target_size=(224, 224),
    batch_size=8,
    class_mode='categorical',
    subset='validation',
    seed=0
)

test_generator = test_datagen.flow_from_directory(
    base_path,
    target_size=(224, 224),
    batch_size=8,
    class_mode='categorical',
    subset='validation',
    seed=0
)

labels = (train_generator.class_indices)
labels = dict((v,k) for k,v in labels.items())

print(labels)

```

```

import tensorflow as tf
from tensorflow.keras.applications import ResNet50
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import GlobalAveragePooling2D, Flatten, Dense

# Load pre-trained ResNet50 model
resnet = ResNet50(weights='imagenet', include_top=False, input_shape=(224, 224, 3))

# Create a new model
def make_model():
    model = Sequential()
    model.add(resnet)
    model.add(GlobalAveragePooling2D())
    model.add(Flatten())
    model.add(Dense(1024, activation="relu"))
    model.add(Dense(512, activation="relu"))
    model.add(Dense(8, activation="softmax", name="classification"))

    return model

model = make_model()

```

```

# Compile the model
model.compile(optimizer=tf.keras.optimizers.SGD(learning_rate=0.0005, momentum=0.9),
              loss='categorical_crossentropy',
              metrics=['accuracy'])

model.summary()

```



```
history = model.fit(train_generator, epochs=30, validation_data=validation_generator)
```

```
model.evaluate(validation_generator)

accuracy = history.history['accuracy']
val_accuracy = history.history['val_accuracy']

loss = history.history['loss']
val_loss = history.history['val_loss']

epochs = range(len(accuracy))

fig, (ax1, ax2) = plt.subplots(1, 2, figsize=(10, 4))

ax1.plot(epochs, accuracy, 'b', label='Training accuracy')
ax1.plot(epochs, val_accuracy, 'r', label='Validation accuracy')
ax1.set_title('Training and validation accuracy')
ax1.set_xlabel('Epochs')
ax1.set_ylabel('Accuracy')
ax1.legend()

ax2.plot(epochs, loss, 'b', label='Training loss')
ax2.plot(epochs, val_loss, 'r', label='Validation loss')
ax2.set_title('Training and validation loss')
ax2.set_xlabel('Epochs')
ax2.set_ylabel('Loss')
ax2.legend()
```

```
fig.suptitle('Training and validation metrics', fontsize=12)
plt.show()
```

```
test_x, test_y = test_generator.__getitem__(1)

preds = model.predict(test_x)

plt.figure(figsize=(12, 16))
for i in range(8):
    plt.subplot(3, 3, i+1)
    plt.title('pred:%s \n truth:%s' % (labels[np.argmax(preds[i])], labels[np.argmax(test_y[i])]))
    plt.imshow(test_x[i])
```

```
model.save_weights("model_daun.h5")
```

2. App.py

```

from flask import Flask, render_template, request, jsonify
from werkzeug.utils import secure_filename
import os
import numpy as np
from PIL import Image
from fungsi import make_model

app = Flask(__name__, static_url_path='/static')

app.config['MAX_CONTENT_LENGTH'] = 1624 * 1024
app.config['UPLOAD_EXTENSIONS'] = ['.jpg', '.png', '.jpeg', '.png']
app.config['UPLOAD_PATH'] = './static/images/uploads/'

model = None

NUM_CLASSES = 8
daun_class = ["Daun Basil", "Daun Kari", "Daun Katuk", "Daun Kelor", "Daun Kemangi", "Daun Kumit", "Daun Salam", "Daun Tewelawak"]

def predict(input_data):
    global model
    if model is not None:
        predictions = model.predict(input_data)
        predicted_class_index = np.argmax(predictions)
        predicted_class = daun_class[predicted_class_index]
        return predicted_class
    else:
        return 'Daun Default'

```

```

@app.route("/")
def beranda():
    return render_template('index.html')

@app.route("/api/deteksi", methods=['POST'])
def apiDeteksi():
    uploaded_file = request.files['file']
    filename = secure_filename(uploaded_file.filename)

    if filename != '':
        file_ext = os.path.splitext(filename)[1]
        if file_ext.lower() in app.config['UPLOAD_EXTENSIONS']:
            uploaded_file.save(os.path.join(app.config['UPLOAD_PATH'], filename))

            test_image = Image.open(os.path.join(app.config['UPLOAD_PATH'], filename))
            test_image_resized = test_image.resize((224, 224))
            image_array = np.array(test_image_resized)
            test_image_x = (image_array / 255.0) - 0.5
            test_image_x = np.expand_dims(test_image_x, axis=0)

            hasil_prediksi = predict(test_image_x)

```

```
# Daftar tautan untuk masing-masing kategori
category_links = {
    "Daun Basil": "https://www.halodoc.com/artikel/ini-6-manfaat-daun-basil-bagi-kesehatan-tubuh",
    "Daun Kari": "https://umsu.ac.id/berita/manfaat-daun-kari/#:~:text=Berikut%20beberapa%20manfaat%20daun%20kar",
    "Daun Katuk": "https://bnp.jambiprov.go.id/17-manfaat-daun-katuk-untuk-kesehatan-dan-ibu-menyusui/",
    "Daun Kelor": "https://www.halodoc.com/artikel/13-manfaat-daun-kelor-untuk-kesehatan",
    "Daun Kemangi": "https://umsu.ac.id/health/manfaat-daun-kemangi-untuk-kesehatan-tubuh/",
    "Daun Kunyit": "https://rri.co.id/padang/ragan/gaya-hidup/1308104/manfaat-daun-kunyit-untuk-kesehatan-dan-ef",
    "Daun Salam": "https://bnp.jambiprov.go.id/12-manfaat-daun-salam-untuk-kesehatan/",
    "Daun Temulawak": "https://www.alodokter.com/menilik-manfaat-temulawak"
}

# Dapatkan tautan berdasarkan hasil prediksi
prediction_link = category_links.get(hasil_prediksi, "https://example.com/default")

return jsonify({
    "prediksi": hasil_prediksi,
    "gambar_prediksi": os.path.join(app.config['UPLOAD_PATH'], filename),
    "prediction_link": prediction_link
})
else:
    return jsonify({
        "error": "Ekstensi file tidak didukung"
    })
}
```

```
if __name__ == '__main__':
    model = make_model()
    model.load_weights("model_daun.h5")
    app.run(host="localhost", port=5000, debug=True)
```

3. Fungsi.py

```
import tensorflow as tf
from tensorflow.keras.layers import GlobalAveragePooling2D, Flatten, Dense, Dropout
from tensorflow.keras.models import Sequential
from tensorflow.keras.applications import ResNet50

resnet = ResNet50(weights='imagenet', include_top=False, input_shape=(224, 224, 3))

# Create a new model
def make_model():
    model = Sequential()
    model.add(resnet)
    model.add(GlobalAveragePooling2D())
    model.add(Flatten())
    model.add(Dense(512, activation="relu"))
    model.add(Dropout(0.2)) # Mengurangi dropout rate
    model.add(Dense(256, activation="relu"))
    model.add(Dropout(0.2)) # Mengurangi dropout rate
    model.add(Dense(8, activation="softmax", name="classification"))

    return model
```