



## **RANCANG BANGUN APLIKASI PENGENALAN WAJAH BERJERAWAT DI BEAUTEER GLOW AESTHETIC CLINIC**

**Cita Puspita Inayati<sup>1</sup>, Sri Lestanti<sup>2</sup>, Saiful Budiman<sup>3</sup>**

<sup>1,2,3</sup>Program Studi Teknik Informatika, Universitas Islam Balitar

Jln. Imam Bonjol No. 16, Sananwetan, Kecamatan Sananwetan, Kota Blitar, Jawa Timur, 66137

<sup>1</sup> [citapuspita882@gmail.com](mailto:citapuspita882@gmail.com), <sup>2</sup> [lestanti85@gmail.com](mailto:lestanti85@gmail.com), <sup>3</sup> [sync.saifulnb@gmail.com](mailto:sync.saifulnb@gmail.com)

---

### **Abstract**

*In carrying out consultations on the condition of acne-prone facial skin, it is still done by answering several questions directly, this is considered less effective considering that other beauty clinics have used real-time skin detection in their online/offline consultations. The purpose of this study is to design and build a simple application that can determine and identify the type of acne and provide real-time and effective acne recognition output. In this study, the author used 100 image datasets with 30 test images that have been validated by experts. As an effort to design and build a simple application that can determine and identify the type of acne and produce effective real-time output. The grouping of acne types consists of 6 types, including blackheads, whiteheads, papules, pustules, nodules and cysts. The results of the implementation of real-time acne facial recognition using the Python programming language with several supporting libraries in it produced a model accuracy with a percentage of 83%. Based on testing using blackbox testing from 33 image data obtained randomly for the type of acne, it showed a percentage of 78.79% whose accuracy results were according to the type of acne, indicating that the application was ready to run.*

**Keywords :** Pimple Type, SVM, Realtime, Blackbox Testing, Python

### **Abstrak**

Dalam menjalankan konsultasi kondisi kulit wajah konsumen yang berjerawat masih dilakukan dengan menjawab beberapa pertanyaan secara langsung, hal ini dianggap kurang efektif mengingat klinik kecantikan lain sudah menggunakan deteksi kulit secara realtime dalam konsultasi daring/luringnya. Tujuan penelitian ini diantaranya merancang dan membangun aplikasi sederhana yang dapat menentukan dan mengidentifikasi jenis jerawat serta menyediakan output hasil pengenalan jerawat secara real time dan efektif. Dalam penelitian ini penulis menggunakan 100 dataset gambar dengan 30 citra uji yang sudah divalidasi oleh pakar. Sebagai upaya untuk merancang dan membangun aplikasi sederhana yang dapat menentukan dan mengidentifikasi jenis jerawat dan menghasilkan output secara realtime yang efektif. Pengelompokan jenis jerawat terdiri dari 6 jenis, diantaranya blackhead, whitehead, papula, pustula, nodula dan kistik. Hasil implementasi pengenalan wajah berjerawat secara realtime menggunakan Bahasa pemrograman Python dengan beberapa library pendukung didalamnya menghasilkan akurasi model dengan persentase 83%. Berdasarkan pengujian menggunakan blackbox testing dari 33 data gambar yang diperoleh secara acak jenis jerawatnya, menunjukkan persentase 78,79% yang hasil akurasinya sesuai jenis jerawat, menunjukkan bahwa aplikasi siap untuk dijalankan.

**Kata kunci :** Jenis Jerawat, SVM, Realtime, Blackbox Testing, Python



## 1. PENDAHULUAN

Penampilan diri (grooming) merupakan penanda citra diri. Penampilan menarik mencerminkan kepribadian seseorang, dan akan dinilai sebagai orang yang berkepribadian baik dan memberikan kesan baik juga bagi orang lain. Begitu sebaliknya, jika tidak merawat penampilan, akan dinilai oleh sebagian orang sebagai individu yang tidak menarik [1]. Dalam situs [larocheposay.co.id](http://larocheposay.co.id) menyebutkan bahwa jerawat bisa menimbulkan dampak psikologis dan menyebabkan ketidakpercayaan diri seseorang [2]. Kepercayaan diri adalah kemampuan dalam meyakinkan diri pada apa yang kita miliki untuk mengembangkan penilaian positif untuk diri sendiri ataupun lingkungan sekitar. Terdapat beberapa faktor yang mempengaruhi tingkat kepercayaan diri seseorang, salah satunya adalah kondisi kulit wajah yang berjerawat [3].

Menurut situs resmi [halodoc.com](http://halodoc.com), jerawat merupakan penyakit kulit yang sering terjadi ketika lapisan luar kulit, terutama folikel rambut, terhambat oleh kotoran, debu, minyak, atau sel kulit mati. akhirnya menyebabkan radang pada pori-pori, yang diiringi dengan infeksi [4]. Untuk penanganan dan pengobatan jerawat tidak dapat dilakukan sembarangan, jika ditangani dengan tidak tepat akan memperburuk kondisi jerawat dan wajah seseorang. Jenis atau kategori jerawat perlu dikelompokkan agar mendapatkan pengobatan yang tepat untuk diberikan pada jerawat tersebut. Pengelompokan jerawat bisa berdasarkan jenis dan tingkat keparahannya [5].

*Beautee Glow Aesthetic Clinic* merupakan klinik kecantikan yang menyediakan jasa perawatan kecantikan dengan kualitas terbaik. Memiliki tenaga dokter dan profesional yang handal sehingga pengguna tidak ragu dalam berkonsultasi dan memakai jasa perawatannya. Melayani face treatment, body treatment, infus service, hair treatment radio frequency, eyes treatment, dental treatment, meso service. Sebagai klinik kecantikan yang sedang berkembang, menarik minat konsumen merupakan hal yang wajib dilakukan salah satunya dengan pengupdate produk terbaru sesuai dengan kebutuhan konsumen, discon di hari-hari tertentu serta memanfaatkan media sosial sebagai sarana mempromosikan produk. Dalam menjalankan konsultasi kondisi kulit wajah konsumen yang berjerawat masih dilakukan dengan menjawab beberapa pertanyaan secara langsung. Hal ini dianggap kurang efektif

mengingat klinik kecantikan yang lain sudah menggunakan deteksi kulit secara realtime dalam konsultasi daring/luringnya.

Berdasarkan permasalahan diatas, maka penulis mengadakan penelitian di *Beautee Glow Aesthetic Clinic* guna membangun suatu aplikasi yang dapat mengenali wajah berjerawat secara realtime dengan citra tersimpan sebelumnya dalam sebuah dataset dan nantinya akan membandingkan persamaan jerawat yang sesuai dengan data inputan sebelumnya dan bisa merekomendasi jenis skincare yang akan diberikan kepada konsumen tersebut.

Software Development Life Cycle (SDLC) merupakan tahapan-tahapan kerja yang bertujuan menghasilkan sistem berkualitas tinggi yang sesuai dengan tujuan sistem itu dibuat. Bisa berupa step by step yang harus dilakukan untuk berjalannya perangkat lunak [6]. Model pengembangan yang digunakan dalam penulisan ini menggunakan model waterfall karena sangat penting dalam membantu proses pengembangan perangkat lunak, dimana harus menyelesaikan satu tahap secara lengkap terlebih dahulu sebelum melangkah ke tahap berikutnya [7].

Sistem kerja pengenalan wajah yaitu menyamakan citra masukan (input) dengan citra yang sudah tersimpan di model yang sudah dibuat sebelumnya dan menemukan kecocokan wajah yang paling sesuai dengan data masukan yang ada sebelumnya [8]. Mengandung arti bahwa masalah pengenalan wajah dapat diselesaikan dengan menggunakan metode klasifikasi. *Support Vector Machine* adalah salah satu metode/algorithm yang mampu menyelesaikan masalah klasifikasi dengan baik [9].

Menurut Kamus Besar Bahasa Indonesia, klasifikasi merupakan pembagian secara sistematis kedalam golongan menurut aturan atau standar yang telah ditetapkan. Sedangkan pengelompokan merupakan proses pengelompokan, cara, atau kegiatan.

Selain SVM, algoritma pengklasifikasian yang paling banyak digunakan di dalam machine learning adalah *K-Nearest Neighbor* (KNN). KNN bekerja dengan mencari rentang terpendek antar data yang akan diestimasi dengan tetangganya di data latih dengan tujuan mengklasifikasikan objek baru berdasarkan atribut dan model data pelatihan serta hasilnya akan diklasifikasikan berdasarkan ketepatan dalam penyelesaian pembelajaran [10].



Dengan perbandingan dua metode pengklasifikasian diatas dan dari hasil studi literatur jurnal menunjukkan bahwa tingkat akurasi algoritma SVM yang lebih baik. Penelitian ini bertujuan untuk merancang dan membangun aplikasi sederhana yang dapat menentukan dan mengidentifikasi jenis jerawat serta menyediakan output hasil pengenalan jerawat secara realtime dengan efektif. Diharapkan aplikasi yang diciptakan lebih mudah digunakan, efektif, dan dapat dijadikan sebagai acuan secara realtime untuk rekomendasi pengobatan jerawat.

## 2. TINJAUAN PUSTAKA

Dalam penelitian ini mempelajari informasi dan penelitian sebelumnya sebagai pembanding. Beberapa penelitian yang menggunakan aplikasi untuk pendeteksian jerawat diantaranya :

Nur Masiqin Mat Isa dan Nur Nabila Abu Mangshor (2021) dari Universiti Teknologi MARA Cawangan Melaka Malaysia dengan judul "*Acne Type Recognition for Mobile-Based Application Using YOLO*". Penelitian ini menciptakan aplikasi *Mobile* yang menggunakan pendekatan Deep Learning untuk mendeteksi jerawat yang muncul di wajah secara *real time*. Aplikasi tersebut diciptakan dengan menggunakan algoritma YOLO (*You Look Only One*) yang mempunyai akurasi 91.25% setelah dilakukan uji coba terhadap 80 gambar [11].

Penelitian yang dilakukan oleh Hangzang dan Tianyi (2022) berjudul "*Acne Detection by Ensemble Neural Network*" dari Nanyang Technology University yang melakukan penelitian mengekstraksi informasi dari gambar dalam perawatan kesehatan untuk membantu dokter mendiagnosis penyakit, terutama penyakit kulit yang representasi visualnya lebih mudah dikenali. Pemeringkatan jerawat merupakan sebuah aplikasi spesifik dari klasifikasi gambar yang bertujuan untuk memperkirakan tingkat keparahan jerawat wajah berdasarkan gambar wajah. Penulis membuat jaringan ansambel untuk menilai tingkat keparahan dan jumlah jerawat dan untuk mendeteksi posisi jerawat berdasarkan kumpulan data publik ACNE04 [12].

Penelitian yang dilakukan oleh Riyan Latifahul Hasanah dan Muhamad Hasan (2022) yang berjudul "*Deteksi Lesi Acne Vulgaris Pada Citra Jerawat Wajah Menggunakan Metode K-Means Clustering*" dari Universitas Nusa Mandiri tentang pengolahan citra digital untuk mendeteksi jerawat *acne vulgaris* pada citra jerawat wajah

menggunakan metode *K-Means Clustering*. *Acne vulgaris* atau jerawat adalah penyakit kulit yang dialami hamper 85% dan juga dapat bertahan hingga bertahun-tahun yang mempengaruhi fisik dan psikologis yang berat pada penderita jerawat kronis. Dengan metode *K-Means Clustering* sebelum dilakukan segmentasi, citra asli dengan format RGB dikonversi menjadi format  $L^*a^*b$ . Hasil penelitian menunjukkan bahwa metode yang diusulkan dapat membedakan lesi jerawat dan lesi non-jerawat dan menandainya dalam *Region of Interest (ROI)* [13].

Haruno Sajati, Yuliani Indrianingsih, dan Puspa Ira Dewi Candra Wulan dari Sekolah Tinggi Teknologi Adisujipto Yogyakarta dengan penelitiannya berjudul "*Deteksi Jerawat Pada Wajah Menggunakan Metode Viola Jones*". Metode *Viola Jones* merupakan salah satu metode yang digunakan untuk mendeteksi suatu objek dengan tingkat keakuratan yang cukup tinggi dan kecepatan yang sangat tinggi. Penelitian ini menggunakan *OpenCV* untuk mentransformasi data dari citra diam atau kamera video ke salah satu keputusan atau representasi baru. Sistem dapat mengenali jerawat baik jerawat biasa maupun jerawat batu, namun metode *Viola Jones* kurang akurat diterapkan pada pendektasian jerawat karena ukuran jerawat hamper sama dengan *noise* di wajah seperti tahi lalat, flek hitam, bekas jerawat, maupun bekas jahitan [14].

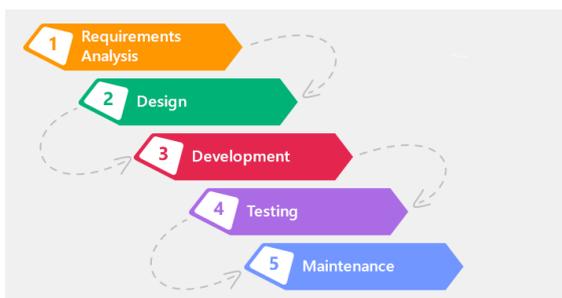
"Deteksi Penyakit Kulit Wajah Menggunakan Tensorflow Dengan Metode Convolutional Neural Network" merupakan penelitian yang dilakukan oleh Indah Widhi Prastika dan Eri Zuliarso dari Program Studi Teknik Informatika Stikubank Semarang. Kulit sendiri merupakan organ paling luar yang berfungsi melapisi dan melapisi seluruh organ tubuh pada manusia. Tentunya kulit sendiri juga bisa terjangkit penyakit kulit terutama di wajah yang dipengaruhi kebersihan, imunitas, kebiasaan, pola hubungan, pergaulan, makanan, seksual, mikrobiologi, factor fisik, bahan kimia, dan lingkungan. Dengan menggunakan bahasa pemrograman java dan python dengan environment aplikasi *Jupyter Notebook* serta *Convolutional Neural Network* peneliti membuat perancangan deteksi penyakit kulit wajah berbasis android. Pendektasian penyakit kulit wajah dilakukan dengan cara menghitung ketepatan pada sistem [15].

Pada penelitian yang berjudul "Pendeteksi Wajah Secara Realtime Menggunakan Metode *Eigenface*" oleh Oki Victoria dan Indra Permana

Solihin dari Fakultas Ilmu Komputer Universitas Pembangunan Nasional Veteran Jakarta dibahas tentang pendeteksi wajah secara realtime yang menggunakan metode *Eigenface*. Dasar dari algoritma *Eigenface* yaitu *Principal Component Analysis* (PCA) atau biasa disebut dengan transformasi *Karhunen-Loeve* yang diimplementasikan dalam bahasa pemrograman C# dengan modul CV dan CXCORE dari library OpenCV. Hasil dari aplikasi pendeteksi wajah secara realtime dibangun Ketika kamera webcam dinyalakan melalui aplikasi pendeteksi wajah [16].

### 3. METODOLOGI PENELITIAN

Penelitian ini menggunakan metode *waterfall* dalam penelitian dan pengembangannya dikarenakan bisa terkontrol antara tiap tahapan, pengembangannya berproses satu model dalam satu waktu sehingga bisa meminimalisir kesalahan. Dinamakan *waterfall* karena model ini berbentuk diagram alir satu arah kebawah. Keunggulan model ini adalah tahapan proses pengembangan jelas dan tepat, serta sistem tertata dengan baik sehingga tidak ada tahapan yang tumpang tindih, karena dalam prosesnya pengembang tidak dapat berpindah ke tahapan berikutnya jika salah satu tahapan tidak ada. Mudah digunakan terutama untuk perangkat dengan sistem yang tidak terlalu rumit [17].



Gambar 1. Jenis Model *Waterfall*

Berikut penjelasan Langkah-langkah penelitian yang dilakukan dengan menggunakan model *waterfall* :

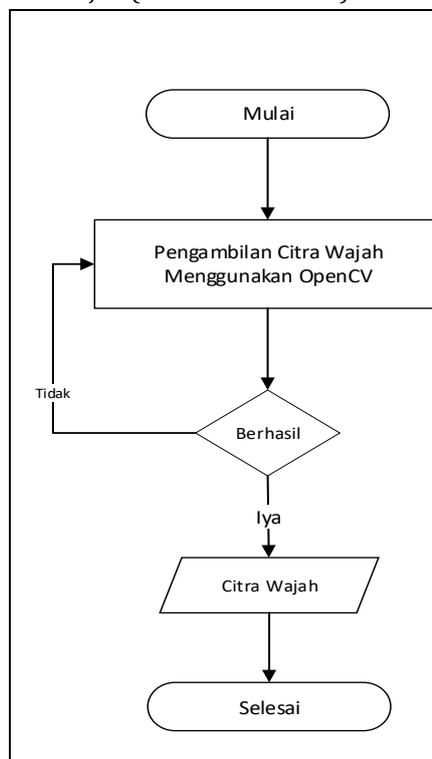
#### 1. Analisis Kebutuhan

Analisis kebutuhan dilakukan untuk kebutuhan pengguna dan kebutuhan system. Analisis kebutuhan berfokus pada perangkat lunak yang akan dibuat. Aplikasi pengenalan wajah berjerawat ini hanya mempunyai satu pengguna yaitu admin yang bisa mengelola

aplikasi ini untuk melakukan proses menambah, mengubah, menghapus, melihat data set wajah berjerawat serta melihat hasil dari pengenalan jerawat. Sementara itu untuk analisa kebutuhan *hardware* dan *software* yang dibutuhkan dalam penyusunan aplikasi ini antara lain perangkat keras (*hardware*) berupa laptop dengan sistem operasi Windows 11 64-bit, processor Intel Core i5, *memory* (RAM) 4096 MB, harddisk 512 GB, kamera HD Webcam dan untuk software berupa *Microsoft Office* 2019, *Microsoft Visio* Profesional 2015, dan *Phyton* 3.11.2 (64-bit).

#### 2. Desain Sistem

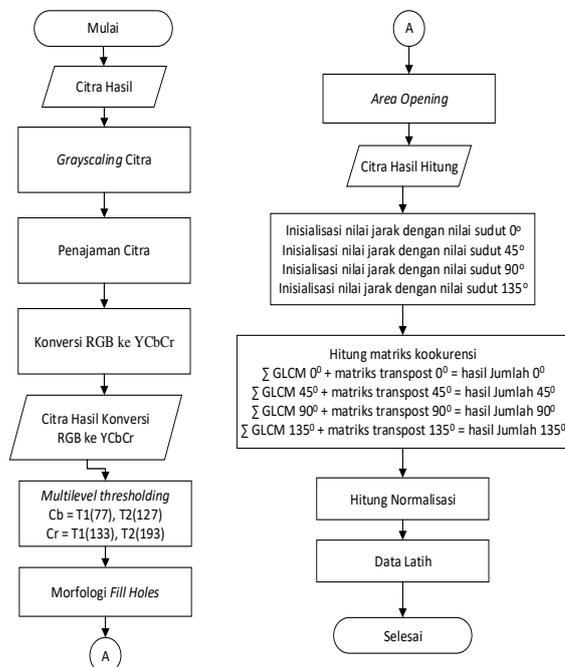
Dalam penelitian ini dibuat tiga desain sistem dengan menggunakan *flowchart* antara lain desain sistem deteksi wajah (*take foto webcam*), desain sistem alur proses mulai dari masukan hingga keluaran, dan desain sistem deteksi jerawat . Berikut adalah alur sistem deteksi wajah (*take foto webcam*) :



Gambar 2. Alur Sistem Deteksi Wajah

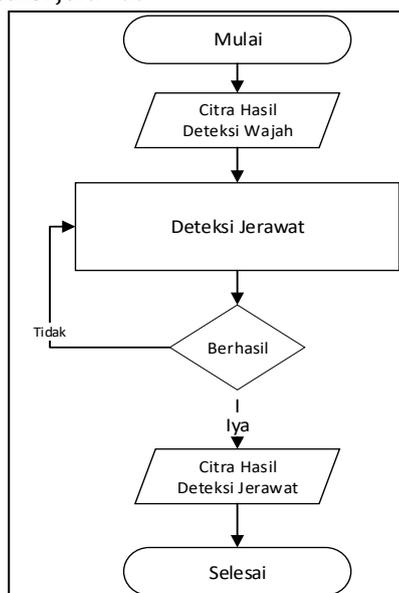
Selain itu untuk memudahkan sistem yang akan dirancang, maka dibuat *flowchart* untuk mendeskripsikan alur proses mulai dari

masukannya hingga keluarannya, seperti gambar berikut :



Gambar 3. Flowchart Alur Proses Masukan Sampai Keluaran

Desain sistem menggunakan flowchart diagram (diagram alir) untuk setiap tahapannya. Berikut adalah alur sistem deteksi jerawat :



Gambar 4. Alur Sistem Deteksi Jerawat

3. Sistem dan Desain Perangkat Lunak (Development)

Database yang digunakan dalam penelitian ini menggunakan database sederhana dengan 2 tabel yaitu Entity Relationship Diagram (ERD) merupakan table yang terdapat didalam database aplikasi wajah berjerawat dan tabel dataset (terdiri dari tabel file dan tabel file detail). Penelitian yang dilakukan dimodelkan dengan menggunakan analisis Unified Modeling Language (UML) yang meliputi identifikasi actor, usecase diagram, sequence diagram, activity diagram, dan scenario diagram.

4. Pengujian

Dalam pengujian sistem perangkat lunak yang dibuat, akan dilakukan menggunakan skema pengujian black box untuk menguji setiap fungsi yang terdapat pada perangkat lunak yang dibangun. Setelah didapatkan hasil dari mengklasifikasikan jerawat, selanjutnya akan dilakukan pengujian akurasi dengan confusion matrik untuk mengetahui tingkat akurasi untuk macam-macam jenis jerawat [18].

5. Pemeliharaan

Pemeliharaan yang dilakukan jika ada kesalahan/ debug sistem pada saat dipakai langsung oleh pengguna sistem yang tidak terdeteksi pada saat pengujian. Pemeliharaan yang dilakukan yaitu dengan menggunakan pemeliharaan korektif (corrective maintenance). Dengan cara deteksinya yaitu dengan mengecek fungsi, kinerja aplikasi, apakah ada yang tidak bekerja dengan baik selanjutnya diperbaiki aplikasi tersebut.

4. HASIL DAN PEMBAHASAN

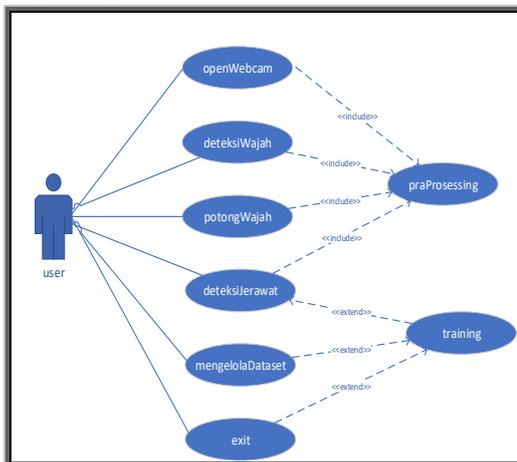
4.1. Perancangan Sistem

Berikut adalah gambaran desain sistem menggunakan Usecase Diagram, Skenario Diagram, Sequence Diagram, dan Activity Diagram.

a. Usecase Diagram

Usecase diagram merupakan rangkaian/uraian sekelompok yang saling terkait dan membentuk sistem secara teratur yang dilakukan atau diawasi oleh aktor. Aktor

yang terlibat dalam sistem yaitu pengguna (user). Pengguna disini merupakan orang yang menggunakan sistem.



Gambar 5. Usecase Diagram

b. Skenario Diagram

Skenario Diagram dibuat untuk berfungsi melakukan pengambilan citra wajah dari direktori untuk selanjutnya dilakukan preprocessing agar citra tersebut dapat diproses. Terdapat 5 tabel skenario antara lain Tabel Skenario openWebcam, Tabel Skenario deteksiWajah, Tabel Skenario potongWajah, Tabel Skenario deteksiJerawat, dan Tabel Skenario Mengelola Dataset. Tabel tersebut ditunjukkan sebagai berikut :

Tabel 1. Skenario openWebcam

<b>Identifikasi</b>	
<b>Nomor</b>	1
<b>Nama</b>	openWebcam
<b>Tujuan</b>	Melakukan open camera di aplikasi
<b>Deskripsi</b>	User dapat mengelola folder untuk yang hanya bisa dieksekusi untuk mengelompokkan citra hasil input webcam ataupun open file.
<b>Aktor</b>	User
<b>Skenario Utama</b>	
<b>Kondisi awal</b>	Dataset untuk kelompok wajah belum dibuat
<b>Aksi Aktor</b>	<b>Reaksi Sistem</b>

<ul style="list-style-type: none"> <li>- User memilih centang</li> <li>- User memilih button update/save</li> <li>- User memilih button exit</li> </ul>	<ul style="list-style-type: none"> <li>- Sistem memunculkan urutan nomor baru</li> <li>- Sistem mengubah nama folder yang dipilih</li> <li>- Sistem menghapus folder</li> </ul>
<b>Kondisi akhir</b>	Citra test berhasil ditampilkan dan dilakukan proses deteksi jerawat

Tabel 2. Skenario deteksiWajah

<b>Identifikasi</b>	
<b>Nomor</b>	2
<b>Nama</b>	deteksiWajah
<b>Tujuan</b>	Mengambil citra wajah dengan media webcam
<b>Deskripsi</b>	User mengambil citra wajah dengan webcam dengan jarak dan keadaan lingkungan yang telah ditentukan.
<b>Aktor</b>	User
<b>Skenario Utama</b>	
<b>Kondisi awal</b>	Directory telah ditentukan
<b>Aksi Aktor</b>	<b>Reaksi Sistem</b>
User meng-capture citra wajah yang akan digunakan sebagai training citra ataupun citra uji	Sistem menangkap citra wajah dari media webcam selanjutnya diproses.
<b>Kondisi akhir</b>	Image citra berhasil ditampilkan dan dilakukan preprocessing pada training citra. Sedangkan pada citra uji perlu dilakukan preprocessing.

Tabel 3. Skenario potongWajah

<b>Identifikasi</b>	
<b>Nomor</b>	3
<b>Nama</b>	potongWajah
<b>Tujuan</b>	Mengambil citra wajah dengan ukuran tertentu



<b>Deskripsi</b>	User mengambil citra wajah dengan <i>webcam</i> dengan jarak dan keadaan lingkungan yang telah ditentukan dan dengan ukuran tertentu
<b>Aktor</b>	User
<b>Skenario Utama</b>	
<b>Kondisi awal</b>	<i>Directory</i> telah ditentukan
<b>Aksi Aktor</b>	<b>Reaksi Sistem</b>
User meng- <i>capture</i> citra wajah dengan ukuran tertentu yang akan digunakan sebagai training citra ataupun citra uji	Sistem menangkap citra wajah dari media <i>webcam</i> dengan ukuran tertentu selanjutnya diproses.
<b>Kondisi akhir</b>	Image citra berhasil ditampilkan dan dilakukan crop, pada training citra.

Tabel 4. Skenario deteksi jerawat

<b>Identifikasi</b>	
<b>Nomor</b>	4
<b>Nama</b>	deteksiJerawat
<b>Tujuan</b>	Mendeteksi citra wajah berjerawat dengan media <i>webcam</i>
<b>Deskripsi</b>	User mengambil citra wajah berjerawat dengan <i>webcam</i> dengan jarak dan keadaan lingkungan yang telah ditentukan.
<b>Aktor</b>	User
<b>Skenario Utama</b>	
<b>Kondisi awal</b>	<i>Directory</i> telah ditentukan
<b>Aksi Aktor</b>	<b>Reaksi Sistem</b>
User meng- <i>capture</i> citra wajah berjerawat yang akan digunakan sebagai training citra ataupun citra uji	Sistem menangkap citra wajah berjerawat dari media <i>webcam</i> selanjutnya diproses.
<b>Kondisi akhir</b>	Image citra berjerawat berhasil ditampilkan dan dilakukan

	<i>preprocessing</i> pada training citra.
--	---

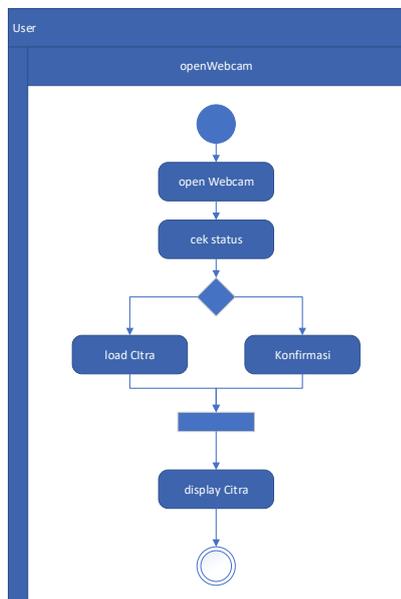
Tabel 5. Mengelola Dataset

<b>Identifikasi</b>	
<b>Nomor</b>	3
<b>Nama</b>	mengelolaData
<b>Tujuan</b>	Menambah, update, delete data file detail
<b>Deskripsi</b>	User dapat mengelola data (create, update, delete).
<b>Aktor</b>	User
<b>Skenario Utama</b>	
<b>Kondisi awal</b>	<i>Directory</i> belum dipilih
<b>Aksi Aktor</b>	<b>Reaksi Sistem</b>
- User memilih letak folder citra. - User memasukkan data - data yang dibutuhkan	- Sistem memunculkan nama folder terpilih - System melakukan manipulasi terhadap data masukan (add, update, delete)
<b>Kondisi akhir</b>	Data berhasil dimanipulasi (add, update, delete)

c. Activity Diagram

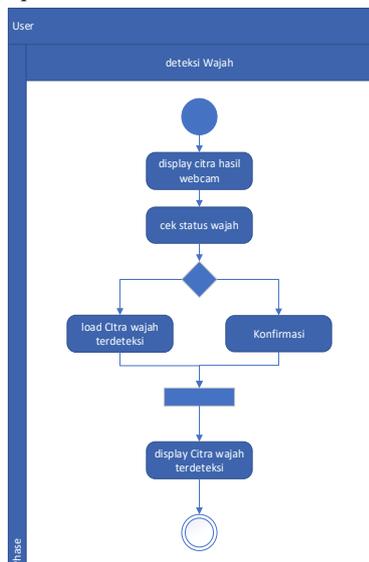
Terdapat 5 *activity* diagram yang dibuat antara lain :

- 1) *Activity* Diagram openWebcam, yang merupakan proses untuk membuka camera webcam, dimana setelah webcam terbuka akan dilakukan check status, apakah bisa untuk meload citra serta mendapatkan konfirmasi, selanjutnya ditampilkan citra tersebut.



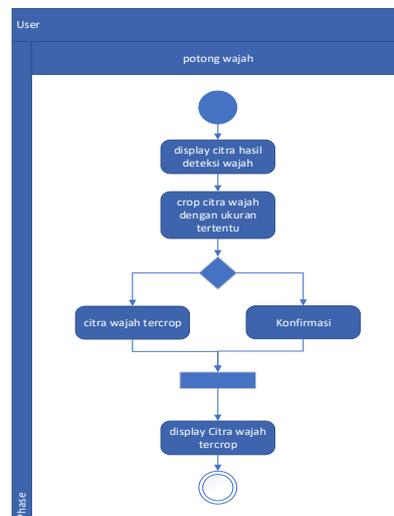
Gambar 6. Activity Diagram openWebcam

2) Activity Diagram deteksiWajah yang merupakan Proses untuk membuka camera webcam, dimana setelah webcam terbuka akan dilakukan check status, apakah bisa untuk meload citra serta mendapatkan konfirmasi, selanjutnya ditampilkan citra tersebut.



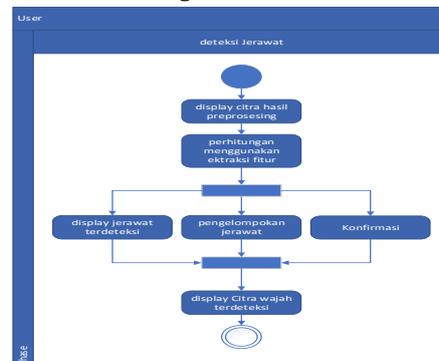
Gambar 7. Activity Diagram deteksiWajah

3) Activity Diagram potongWajah yang merupakan proses untuk menampilkan citra wajah dengan ukuran tertentu hasil display citra.



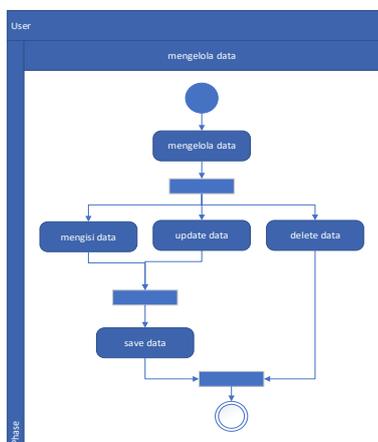
Gambar 8. Activity Diagram potongWajah

4) Activity Diagram deteksiJerawat yang merupakan proses untuk deteksi jerawat, dimana setelah input image terbuka akan dilakukan perhitungan dengan menggunakan ekstraksi fitur, apakah terdeteksi kedalam jenis jerawat yang sesuai serta mendapatkan konfirmasi, selanjutnya ditampilkan hasil proses deteksi dari image tersebut.



Gambar 9. Activity Diagram deteksiJerawat

5) Activity Diagram mengelola dataset merupakan proses untuk mengelola data jerawat, dimana sebelum dilakukan proses pendeteksian jenis jerawat dilakukan input data image sebagai rule ciri jenis jerawat dan mendapatkan model jenis jerawat dan dilakukan preprocessing untuk mendapatkan ekstraksi fitur dan didapatkan model .h5 (data set).



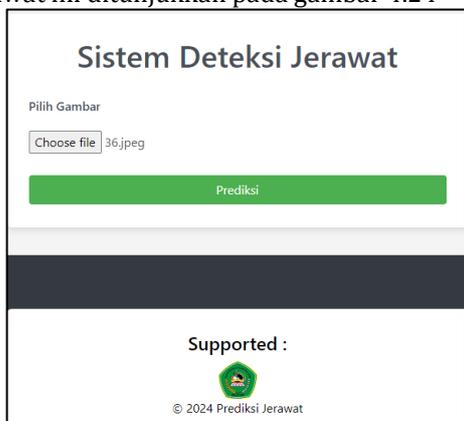
Gambar 10. Activity Diagram Mengelola Dataset

4.2. Implementasi Antarmuka

Berikut adalah implementasi antarmuka atau interface dari aplikasi pengalan wajah berjerawat yang dibuat :

a. Tampilan Home

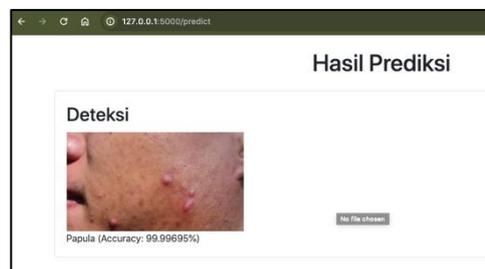
Untuk hasil tampilan mengenai aplikasi Deteksi Jerawat ini ditunjukkan pada gambar 4.24



Gambar 11. Tampilan Home

b. Tampilan Hasil Prediksi

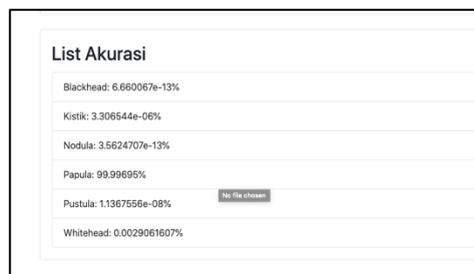
Setelah melakukan proses upload dan melakukan klik pada tombol prediksi sistem akan melakukan deteksi terkait jenis jerawat apa yang sesuai dengan gambar yang telah diupload. Hasil deteksi jenis jerawat ditunjukkan pada gambar 4.25. Selain hasil deteksi pada halaman ini juga menampilkan list prediksi terbesar dari kelima jenis jerawat yang telah ditentukan. Pada studi kasus ini prediksi terbesar ada pada jenis jerawat papula dengan persentase sebesar 86%.



Gambar 12. Hasil Prediksi Jenis Jerawat

c. Tampilan List Akurasi

Selain hasil deteksi pada aplikasi ini juga menampilkan hasil list persentase deteksi terbesar sesuai hasil deteksi jenis jerawat. Hasil list deteksi ditunjukkan pada gambar 4.26.



Gambar 13. Hasil Prediksi Jenis Jerawat

4.3 Pengujian Blackbox

Penelitian ini menggunakan pengujian blackbox dan close beta. Pengujian blackbox merupakan pengujian kualitas perangkat lunak yang memiliki fokus dalam pengujian fungsionalitas dari sebuah program. Penggunaan blackbox dalam penelitian ini digunakan untuk mengetahui input dan output dari aplikasi yang telah dikembangkan. Berikut hasil pengujian yang telah dilakukan :

Tabel 6. Pengujian Blacbox dan Close Beta

No	Fitur	Hasil yang diharapkan	Hasil yang diperoleh	Kesimpulan
1	Halaman Utama	Dapat menampilkan halaman terkait aplikasi deteksi jerawat	Sesuai	Berhasil
2	Klik proses choose file	Dapat melakukan proses	Sesuai	Berhasil



		pilih file gambar		
3	Klik <i>open</i>	Dapat menampilkan file yang dipilih pada gui	Sesuai	Berhasil
4	Klik prediksi tanpa klik <i>choose file</i> terlebih dahulu	Menampilkan peringatan untuk memasukkan gambar terlebih dahulu "please select a file"	Sesuai	Berhasil
5	Klik prediksi	Menampilkan file gambar yang sudah dipilih sebelumnya dan hasil prediksi jenis jerawatnya	Sesuai	Berhasil
6	Menampilkan hasil deteksi dari file gambar yang diuji (33 gambar)	Setelah file gambar diupload, jenis jerawat akan dideteksi dan menampilkan hasil persentase jenis jerawat yang dideteksi	25 Sesuai 8 Tidak Sesuai	25 Berhasil 8 Tidak Berhasil

Pada tabel diatas pengujian terkait halaman beranda, terdapat beberapa fitur yang akan diujikan yang mana dilakukan pengujian terkait

keberhasilan dan kegagalan dari fungsionalitas aplikasi.

Berikut hasil total skenario pengujian pada halaman beranda. Total skenario yang diujikan sebanyak 38 kali pengujian dengan fitur dan file gambar yang berbeda.

Skenario Berhasil = 30

Skenario Gagal = 8

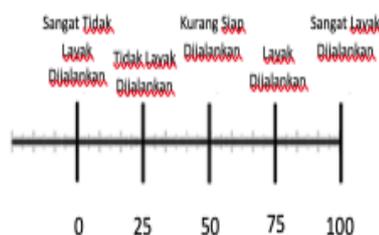
Selanjutnya dari pengujian *blackbox*, telah dilakukan proses *persentase* pengujian dengan rumus [19]:

$$\frac{\text{Hasil Pengujian berhasil}}{\text{Hasil Total Pengujian}} \times 100\%$$

$$= \frac{30}{38} \times 100\%$$

$$= 78,79\%$$

Skor yang telah didapat selanjutnya dimasukkan ke dalam bentuk interpretasi skor sebagai berikut :



Gambar 14. Interval Rating Scale

Berikut ini adalah hasil presentase hasil jawaban yang sudah dihitung nilainya dengan menggunakan rumus *blackbox*. Total yang didapat adalah sebesar 78,79%. Maka dapat disimpulkan bahwa aplikasi ini sudah dapat memenuhi tujuannya untuk aplikasi layak dijalankan.

5. KESIMPULAN DAN SARAN

Pengembangan aplikasi pengenalan wajah berjerawat secara realtime menggunakan metode SDLC jenis waterfall memiliki keunggulan proses pengembangannya jelas dan tepat, serta tertata dengan baik sehingga tidak ada tahapan yang tumpang tindih. Berdasarkan pengujian *blackbox* menunjukkan bahwa persentase yang dihasilkan sebesar 78,79%, menunjukkan bahwa aplikasi menurut interval rating scale layak untuk dijalankan.

Penggunaan Metode atau algoritma klasifikasi lain seperti decision tree untuk mengetahui apakah dapat digunakan untuk deteksi jenis jerawat. Memperbanyak citra dataset jerawat meningkatkan kemampuan deteksi



jerawat. Diperlukan pencahayaan yang merata dan bagu0s untuk hasil akurasi yang lebih maksimal. Diperlukan pengembangan selanjutnya yaitu rekomendasi penanganan pengobatan yang diperlukan untuk setiap jenis jerawat.

## 6. UCAPAN TERIMA KASIH

Ucapan terima kasih kami sampaikan kepada Prodi Teknik Informatika Universitas Islam Balitar atas bantuan dan dukungannya. Terima kasih kepada pemilik Beautee Glow Aesthetic Clinic yang telah membantu memberikan data dalam penelitian ini.

## DAFTAR PUSTAKA:

- [1] J. Arifianto and I. Muhimmah, "Aplikasi Web Pendeteksi Jerawat Pada Wajah Menggunakan Algoritma Deep Learning dengan TensorFlow," *J. Autom.*, pp. 21–29, 2021.
- [2] "Stress Hingga Depresi karena Jerawat? Ini Solusinya!" Accessed: Mar. 21, 2024. [Online]. Available: <https://www.larocheposay.co.id/article/s-tres-dan-depresi-akibat-jerawat>
- [3] M. R. Rais, "Kepercayaan Diri (Self Confidence) Dan Perkembangannya Pada Remaja," *Al-Irsyad*, vol. 12, no. 1, p. 40, 2022, doi: 10.30829/al-irsyad.v12i1.11935.
- [4] "Apa itu Jerawat? Gejala, Penyebab & Pengobatan - Halodoc." Accessed: Mar. 24, 2024. [Online]. Available: <https://www.halodoc.com/kesehatan/jerawat?srsltid=AfmBOoqHW07c4zMMZ1GOh5IrTYPPPKOaFp6Y3NHCuzDsh02HqO2Xqjgi>
- [5] "Mengenal Jenis-Jenis Jerawat dan Cara Mengatasinya." Accessed: Mar. 24, 2024. [Online]. Available: <https://www.siloamhospitals.com/informasi-siloam/artikel/jenis-jenis-jerawat>
- [6] K. Sutanto, D. Arisandi, and N. J. Perdana, "Aplikasi Pemesanan Berbasis Website Pada Life and Story Coffee," *J. Serina Sains, Tek. dan Kedokt.*, vol. 1, no. 2, pp. 259–266, 2023, doi: 10.24912/jsstk.v1i2.28512.
- [7] M. Badrul, "Penerapan Metode waterfall untuk Perancangan Sistem Informasi Inventory Pada Toko Keramik Bintang Terang," *PROSISKO J. Pengemb. Ris. dan Obs. Sist. Komput.*, vol. 8, no. 2, pp. 57–52, 2021, doi: 10.30656/prosisko.v8i2.3852.
- [8] T. Susim and C. Darujati, "Pengolahan Citra untuk Pengenalan Wajah (Face Recognition) Menggunakan OpenCV," *J. Syntax Admiration*, vol. 2, no. 3, pp. 534–545, 2021, doi: 10.46799/jsa.v2i3.202.
- [9] A. B. S. Suma'inna, and H. Maulana, "JURNAL TEKNIK INFORMATIKA VOL 9 NO. 2, OKTOBER 2016 | 166 Pengenalan Citra Wajah Sebagai Identifier Menggunakan Metode Principal Component Analysis (PCA)," *J. Tek. Inform.*, vol. 9, no. 2, pp. 166–175, 2016.
- [10] S. R. Cholil, T. Handayani, R. Prathivi, and T. Ardianita, "Implementasi Algoritma Klasifikasi K-Nearest Neighbor (KNN) Untuk Klasifikasi Seleksi Penerima Beasiswa," *IJCIT (Indonesian J. Comput. Inf. Technol.)*, vol. 6, no. 2, pp. 118–127, 2021, doi: 10.31294/ijcit.v6i2.10438.
- [11] N. A. M. Isa and N. N. A. Mangshor, "Acne Type Recognition for Mobile-Based Application Using YOLO," *J. Phys. Conf. Ser.*, vol. 1962, no. 1, 2021, doi: 10.1088/1742-6596/1962/1/012041.
- [12] H. Zhang and T. Ma, "Acne Detection by Ensemble Neural Networks," *Sensors*, vol. 22, no. 18, pp. 1–16, 2022, doi: 10.3390/s22186828.
- [13] R. L. Hasanah and M. Hasan, "Deteksi Lesi Acne Vulgaris pada Citra Jerawat Wajah Menggunakan Metode K-Means Clustering," *Indones. J. Softw. Eng.*, vol. 8, no. 1, pp. 46–51, 2022, doi: 10.31294/ijse.v8i1.12966.
- [14] Y. Indrianingsih, H. Sajati, and P. I. D. C. Wulan, "Deteksi Jerawat Pada Wajah Menggunakan Metode Viola Jones," *Compiler*, vol. 5, no. 1, pp. 53–64, 2016.
- [15] I. W. Prastika and E. Zuliarso, "Deteksi Penyakit Kulit Wajah Menggunakan Tensorflow Dengan Metode Convolutional Neural Network," *J. Manaj. Inform. dan Sist. Inf.*, vol. 4, no. 2, pp. 84–91, 2021, doi: 10.36595/misi.v4i2.418.
- [16] O. Victoria and I. P. Solihin, "Pendeteksi Wajah Secara Realtime Menggunakan Metode Eigenface," *SEINASI-KESI (Seminar Nas. Inform. Sist. Inf. Dan Keamanan Siber)*, no. metode eigenface, pp. 126–131, 2018.
- [17] F. H. T. Ilham, A. Yaafi, R. Nurahmat, "PENCARIAN ANGKUTAN KOTA MAJALENGKA DENGAN METODE," *J. Manaj. Inform. dan Sist. Inf.*, vol. 7, no. 2, pp.



- 143–149, 2024, doi: 10.36595/misi.v5i2.
- [18] F. Wazir, “Rancang bangun sistem pengenalan wajah dengan metode,” vol. 1, no. 2, pp. 59–75, 2016.
- [19] I. Ernawati, “Uji Kelayakan Media Pembelajaran Interaktif Pada Mata Pelajaran Administrasi Server,” *Elinvo*

*(Electronics, Informatics, Vocat. Educ., vol. 2, no. 2, pp. 204–210, 2017, doi: 10.21831/elinvo.v2i2.17315.*