

LAPORAN PROYEK MIDTERM
KOMPUTASI AWAN



Kelompok 2

Anggota:

Athillah Syamsuddin	162012233012
Alfico Permana	162012233050
Amadeus Gavriel Tataming	162012233070
Safa Saviera Sulaiman	162112233012
Alfa Noora Fithria	162112233029

TEKNIK ROBOTIKA DAN KECERDASAN BUATAN
FAKULTAS TEKNOLOGI MAJU DAN MULTIDISIPLIN
UNIVERSITAS AIRLANGGA
2024

I. Latar Belakang

Dalam era modern ini, kesehatan kulit menjadi perhatian yang semakin penting bagi banyak orang, terutama wanita. Banyak dari mereka sangat mendambakan kulit wajah yang cerah, sehat, bersih, dan terawat dengan baik. Namun, seringkali dalam upaya merawat kulit wajah, mereka tidak mempertimbangkan jenis kulit yang mereka miliki. Setiap orang memiliki jenis kulit wajah yang berbeda, sehingga permasalahan yang muncul pada kulit wajah pun bermacam-macam seperti jerawat, kulit kering, dan gangguan kulit lainnya.

Masalah kulit, seperti jerawat, dapat secara signifikan mempengaruhi kepercayaan diri seseorang, yang pada akhirnya dapat berdampak pada kehidupan sosial, karir, dan kesejahteraan mental secara keseluruhan. Oleh karena itu, banyak orang menggunakan skincare sebagai salah satu cara untuk meningkatkan kepercayaan diri mereka (Aladwan, dkk. 2023).. Dengan demikian, penggunaan skincare atau kosmetik yang tepat dapat meningkatkan kualitas hidup dengan memperbaiki citra dan kompetensi diri, baik pada wanita maupun pria (Bouslimani, A., dkk. 2019; Byrne, A.)

Untuk dapat melakukan perawatan kulit yang tepat dan efektif, diperlukan pemahaman yang memadai tentang karakteristik dan kebutuhan khusus kulit seseorang. Melakukan konsultasi dengan dokter kulit atau ahli kecantikan profesional menjadi pilihan yang dapat diambil untuk mendapatkan saran dan rekomendasi produk perawatan kulit yang sesuai. Akan tetapi, biaya konsultasi dengan tenaga ahli tersebut terkadang cukup tinggi, sehingga banyak orang tidak melakukannya (Sirapanji S. 2014). Selain itu, terbilang cukup banyak individu merasa malu atau tidak nyaman untuk berkonsultasi langsung dengan dokter kulit atau ahli kecantikan tentang kondisi kulit wajah mereka (Oktaviani, E., & Budiyanto, U. 2023)

Penggunaan skincare menjadi langkah mandiri untuk merawat kesehatan kulit. Namun, memilih produk skincare yang tepat juga menjadi tantangan tersendiri. Pemilihan produk perawatan kulit yang salah seringkali berujung pada kondisi kulit yang memburuk, terutama bagi mereka yang memiliki kulit sensitif yang rentan terhadap bahan kimia berbahaya (Liang W. 2020)

Untuk mengatasi masalah-masalah tersebut, dibutuhkan sebuah solusi yang dapat membantu orang-orang memilih produk skincare yang tepat, tanpa harus menanggung biaya konsultasi yang tinggi. Solusi yang diusulkan adalah "Skincare Recommendation with Face Detection AI" yang memangkas dan membuat segmentasi pada foto wajah untuk memudahkan ahli dalam menganalisis kondisi kulit dan memberikan rekomendasi produk skincare yang sesuai. Dengan mesin ini, pengguna dapat melakukan pemindaian wajah

mereka secara mandiri dan mesin akan mengekstrak gambar wajah dan membuat segmentasinya. Kemudian, gambar yang sudah diolah akan dikirim ke ahli untuk dianalisis karakteristik kulit seperti jenis kulit dan masalah kulit. Lalu, ahli kulit memberikan rekomendasi produk skincare yang paling sesuai untuk kebutuhan kulit pengguna. Solusi ini diharapkan dapat membantu orang-orang memilih produk skincare yang tepat, meningkatkan kepercayaan diri, dan mengatasi kendala biaya.

II. Metode

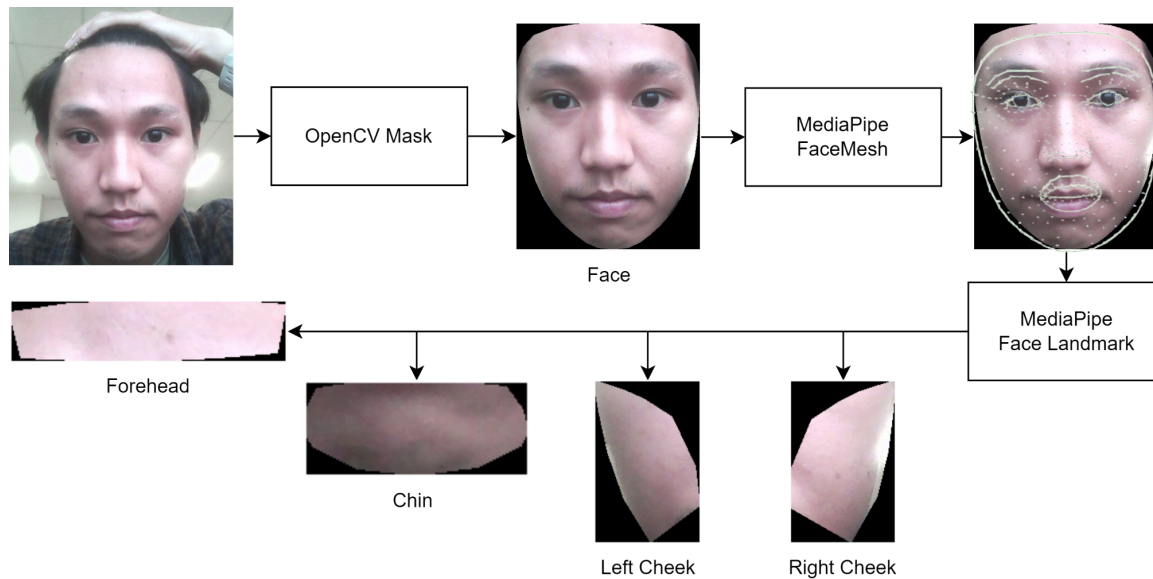
Pengembangan Visi Komputer

User yang menginginkan rekomendasi *skincare* diperlukan untuk mengunggah foto wajahnya yang kemudian akan disegmentasi dan dikirimkan kepada *user* ahli untuk menentukan rekomendasi *skincare*. Dalam mengekstraksi gambar wajah dan berbagai fitur yang disegmentasi, digunakan *library* OpenCV dan MediaPipe.

Library OpenCV merupakan *library* yang *open source* yang dikembangkan dengan bahasa C++ dan berfokus pada penggunaan secara *real-time*. Salah satu tujuan dari OpenCV adalah membantu orang-orang dalam membangun infrastruktur visi komputer dengan menyediakan berbagai fungsi dalam aspek-aspek pengolahan gambar dan video serta modul-modul *Machine Learning* (ML) yang berfokus pada pengenalan pola dan *clustering*. (Kaehler, A., & Bradski, G. 2016).

Sedangkan, MediaPipe atau MediaPipe Solutions merupakan sebuah *library* yang menyediakan berbagai model *machine learning* dan *artificial intelligence* yang siap untuk diaplikasikan. Pada MediaPipe, terdapat sebuah fitur bernama Face Mesh. Hanya dengan 1 gambar sebagai masukan, Face Mesh dapat menentukan 478 *landmark* dalam bentuk 3D. Selain itu, segmentasi pada gambar wajah dapat dilakukan dengan memotong berbagai *landmark* yang diinginkan (MediaPipe)

Dengan menggunakan OpenCV dan MediaPipe, gambar dari wajah *user* dapat diekstraksi beserta segmentasi pada bagian-bagian wajah tertentu. Pada Gambar 1 di bawah menggambarkan garis besar bagaimana proses segmentasi dari gambar wajah.



Gambar 1. Proses Ekstraksi dan Segmentasi Wajah

Pengembangan Front-End

Javascript adalah bahasa yang digunakan untuk mengembangkan *front-end*. Selain itu, pengembangan *front-end* dilengkapi oleh berbagai *library* yang antara lain: ReactJS, Chakra UI, React Router, dan Nodejs. ReactJS merupakan *library* yang populer dikarenakan alur pengerjaan yang simpel untuk pengembangan web dan aplikasinya (Vipul, A. M., & Sonpatki, P. 2016). Tidak hanya itu, ReactJS dilengkapi oleh Chakra UI yang komprehensif dalam mengembangkan komponen-komponen *user interface* yang kompleks namun tetap menjaga stabilitasnya (Adebayo, Segun, dkk. 2023). Namun, ReactJS membutuhkan NodeJS yang juga populer dalam membangun *framework* Javascript yang asinkronus (Thompson, M. 2011). Sedangkan, React Router digunakan untuk navigasi *user* dalam sebuah web (Ganatra, S. 2018).

Aplikasi web didesain memiliki empat halaman. Halaman pertama adalah halaman *home* serta *landing* di mana terdapat judul dari proyek midterm ini beserta motto kelompok. Pada halaman kedua, atau halaman *scan*, *user* diberi pilihan untuk mengunggah foto wajahnya atau menjepret langsung melalui *webcam*. Kemudian, *user* diperlukan untuk mengisi formulir yang berisikan nama, email, dan kondisi wajah secara keseluruhan pada halaman ketiga atau halaman *form*. Pada halaman terakhir atau halaman *result*, *user* disajikan foto wajahnya yang telah tersegmentasi.

Pengembangan Back-End

Metode pengembangan back-end dimulai dengan mempersiapkan Virtual Private Server (VPS) menggunakan sistem operasi Ubuntu Server melalui layanan Azure Virtual Machine. Pendekatan ini memberikan lingkungan server yang terisolasi, aman, dan dapat diakses dari mana saja melalui internet. Setelah VPS siap, akses ke server dilakukan melalui protokol SSH (Secure Shell) menggunakan aplikasi PuTTY. SSH memfasilitasi komunikasi aman antara dua sistem dengan arsitektur klien-server, memungkinkan remote sistem, forwarding port TCP, tunnelling, dan koneksi X11 (Tohirin, 2020). Setelah terhubung dengan server melalui SSH, konfigurasi server dilakukan di shell sesuai dengan kebutuhan, meliputi pengaturan firewall, pembaruan sistem operasi, instalasi paket tambahan, dan lainnya untuk memastikan lingkungan server yang aman dan optimal bagi aplikasi web.

Web server Nginx diinstal pada server Ubuntu sebagai salah satu web server populer yang dikenal karena kinerjanya yang tinggi dalam menangani permintaan HTTP. Nginx merupakan web server yang ringan dan memiliki performa yang cepat, Nginx dapat memproses beberapa *request* dengan baik. Nginx dapat diakses oleh banyak klien diwaktu yang bersamaan yang artinya server Nginx mampu menerima banyak traffic dengan menggunakan load balancing Nginx (Apriliansyah, dkk. 2020). File-file aplikasi web kemudian ditransfer ke server menggunakan FTP FileZilla yang memungkinkan unggah dan unduh file secara aman melalui protokol FTP (File Transfer Protocol), meliputi kode sumber, aset statis seperti gambar, CSS, JavaScript, dan file konfigurasi.

Metode ini memberikan fondasi yang kuat untuk pengembangan back-end yang efisien dan terkelola dengan baik, di mana VPS Ubuntu Server menyediakan lingkungan server yang terisolasi dan dapat diakses dari mana saja melalui internet, konfigurasi server melalui shell memungkinkan penyesuaian lingkungan sesuai kebutuhan aplikasi web, sementara web server Nginx menangani permintaan HTTP dari klien dan FileZilla memfasilitasi transfer file aplikasi web ke server melalui FTP.

Langkah selanjutnya adalah menginstal Docker pada server Ubuntu. Docker merupakan proyek *open source* yang memanfaatkan berbagai teknologi yang telah lama dikenal dalam dunia penelitian sistem operasi antara lain kontainer LXC, virtualisasi OS, pengelolaan perubahan berbasis hash, dll. Docker menggabungkan teknologi-teknologi ini untuk menyediakan solusi yang lengkap (Boettiger, C. 2014). Menginstal Docker pada server Ubuntu bertujuan untuk menjalankan program AI yang telah dibuat dengan lebih mudah dan

fleksibel. Program AI akan dikemas dalam sebuah kontainer Docker, yang memastikan bahwa program tersebut akan berjalan dalam lingkungan yang terisolasi dan konsisten.

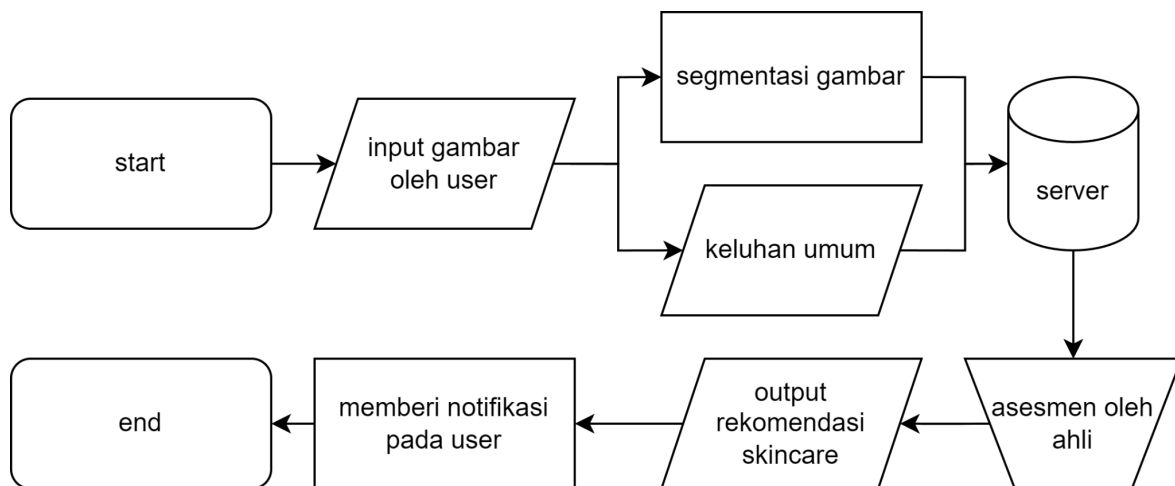
Membangun image Docker untuk program AI dengan menggunakan Dockerfile. Dockerfile ini berisi instruksi untuk mengkonfigurasi lingkungan yang dibutuhkan oleh program AI, seperti menginstal dependensi, menyalin file program, dan menentukan perintah untuk menjalankan program tersebut. Setelah image Docker berhasil dibangun, dapat menjalankannya sebagai sebuah kontainer Docker pada server Ubuntu. Kontainer ini akan menjalankan program AI dalam lingkungan terisolasi, sehingga tidak akan mengganggu aplikasi lain yang berjalan pada server yang sama. Docker juga memudahkan proses deployment program AI ke lingkungan lain, seperti lingkungan produksi, karena kontainer Docker menjamin konsistensi lingkungan dimanapun kontainer tersebut dijalankan.

Selain menjalankan program AI dalam lingkungan terisolasi menggunakan Docker, pengembangan back-end juga melibatkan pengelolaan database untuk menyimpan dan mengakses data sistem website. Dalam hal ini, Firebase Cloud Storage digunakan sebagai solusi penyimpanan objek yang aman dan terkelola untuk menyimpan konten seperti gambar, file, dan data lain yang digunakan oleh sistem website. *Cloud Storage for Firebase* adalah layanan penyimpanan objek yang canggih, sederhana, dan hemat biaya yang dibuat untuk skala Google. *Cloud Storage for Firebase* menyimpan file di *bucket* Google Cloud Storage, sehingga dapat diakses melalui Firebase dan Google Cloud. Hal ini memberi fleksibilitas untuk mengupload dan mendownload file dari klien seluler melalui Firebase SDK untuk Cloud Storage.(Firebase).

Alur Kerja

Pada Gambar 2, disajikan bagaimana alur dari “Skincare Recommendation with Face Detection AI” bekerja. Saat membuka halaman utama, *user* diminta untuk memasukkan foto dari wajahnya, baik dengan mengunggahnya dari *personal computer* atau menjepretnya melalui *webcam*. Setelah itu, foto yang telah diambil akan diproses oleh *support AI* untuk memangkas foto wajah dan membuat segmentasi pada bagian dahi, dagu, pipi kanan dan kiri. Selain itu, *user* juga diminta untuk mengisi formulir yang berisikan nama, email, dan keluhan umum. Kemudian, gambar dan formulir yang diperoleh akan dikirim ke *server* dan menunggu dinilai oleh *user* yang ahli dalam bidang wajah. Di sisi lain, *user* mendapatkan foto wajahnya yang telah dipangkas dan tersegmentasi. Setelah dinilai oleh ahli, ahli akan

menyusun rekomendasi *skincare* dan mengirimkannya kembali pada *user*. Proses ini berakhir dengan *user* yang mendapatkan notifikasi.



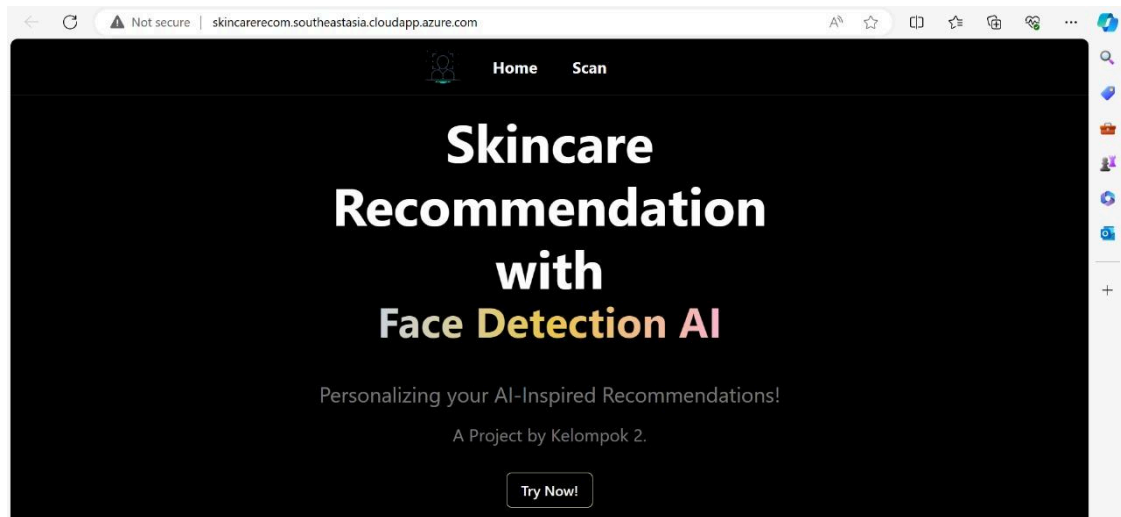
Gambar 2. Diagram Alir Kerja Web

III. Hasil

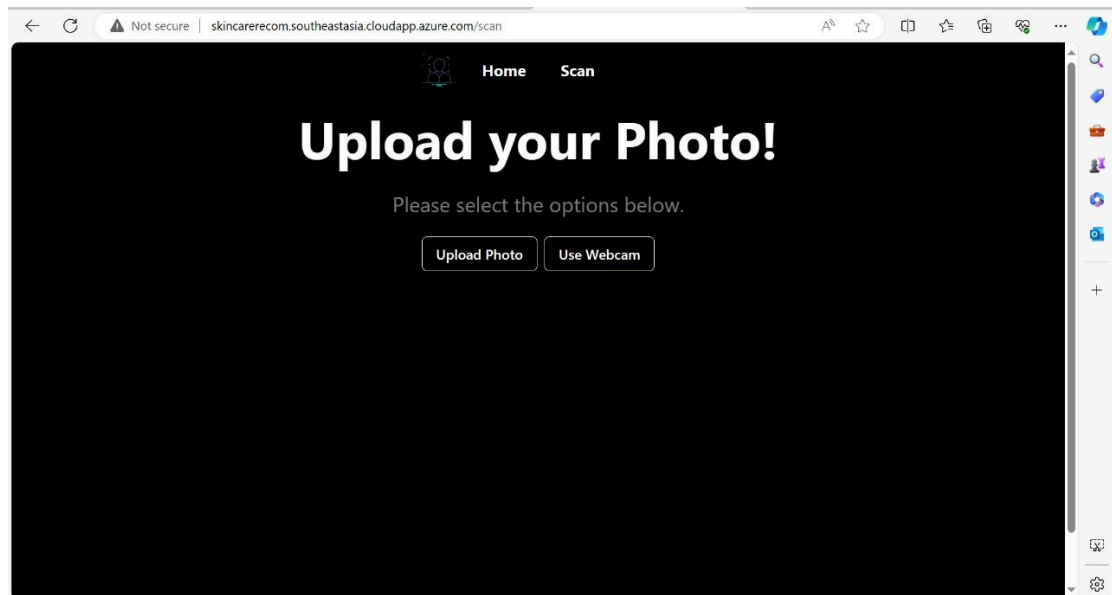
Dalam penelitian ini, dikembangkan sebuah website yang mampu memberikan rekomendasi produk *skincare* yang sesuai dengan kondisi kulit pengguna. Website ini dilengkapi dengan fitur untuk mengupload foto wajah atau mengambil gambar secara langsung melalui webcam. Setelah gambar wajah berhasil diperoleh, data gambar tersebut akan diolah oleh model Artificial Intelligence (AI) yang telah dilatih untuk memangkas dan membuat segmentasi wajah.

Melalui proses deteksi, pemangkas, dan segmentasi foto wajah, sistem akan memberikan hasil foto yang diolah kepada ahli untuk identifikasi karakteristik kulit seperti jenis kulit (kering, berminyak, atau kombinasi), adanya masalah kulit seperti jerawat, noda hitam, atau kerutan. Berdasarkan hasil analisis oleh ahli tersebut, website akan menampilkan informasi tentang kondisi kulit pengguna secara spesifik yang diberikan oleh ahli kulit.

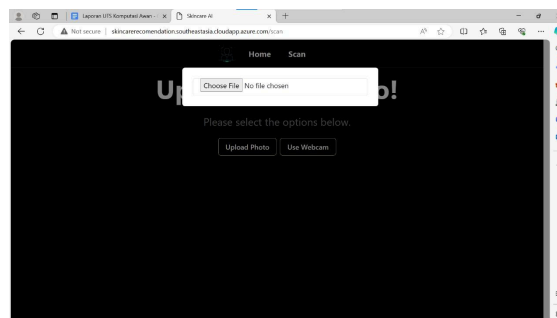
Selanjutnya, website akan memberikan rekomendasi produk *skincare* yang sesuai dengan kondisi kulit pengguna yang direkomendasikan oleh ahli. Rekomendasi ini mencakup jenis produk seperti pembersih wajah, pelembab, serum, masker wajah, dan produk perawatan kulit lainnya dari berbagai merek. Setiap rekomendasi disertai dengan deskripsi singkat mengenai manfaat dan fungsi produk tersebut untuk membantu pengguna dalam memilih produk yang tepat.



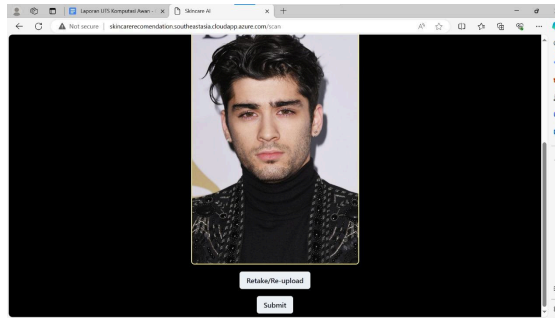
Gambar 3. Halaman Landing Page atau Home



Gambar 4. Halaman Scan



Gambar 5. Halaman Scan: Upload Photo



Gambar 6. Halaman Scan: Submit Photo

Gambar 7. Halaman Form

Tampilan website didesain dengan antarmuka yang menarik dan mudah digunakan, sehingga pengguna dapat dengan mudah meng-upload atau mengambil gambar, menerima informasi kondisi kulit, serta mengakses rekomendasi produk skincare yang sesuai. Gambar 3 menunjukkan halaman awal, dan dengan menekan tombol ‘Try Now!’ maka user diarahkan ke halaman scan. Halaman Scan, seperti pada Gambar 4, memberikan pilihan pada user untuk mengunggah foto atau menjepret langsung wajahnya. Untuk mengunggah foto, user disajikan seperti pada Gambar 5. Setelah mengunggah, user tinggal menekan ‘Submit’ seperti pada Gambar 6. Setelah di-submit, user diminta untuk mengisi formulir seperti pada Gambar 7. Dengan demikian, pengguna dapat memperoleh rekomendasi produk skincare yang sesuai dengan kebutuhan kulit mereka secara personal.

Daftar Pustaka

- Adebayo, Segun, dkk. (2023). Chakra: Build Accessible React Apps with Speed. GitHub. Diambil pada tanggal 26 Maret 2024 di <https://github.com/chakra-ui/chakra-ui/>
- Apriliansyah, Fahmi & Fitri, Iskandar & Iskandar, Agus. (2020). Implementasi Load Balancing Pada Web Server Menggunakan Nginx. *Jurnal Teknologi dan Manajemen Informatika*. 6. 10.26905/jtmi.v6i1.3792.

- Boettiger, C. (2014). An introduction to Docker for reproducible research. *ACM SIGOPS Oper. Syst. Rev.*, 49, 71-79.
- Bouslimani, A., da Silva, R., Kosciolk, T., Janssen, S., Callewaert, C., Amir, A., ... & Dorrestein, P. C. (2019). The impact of skin care products on skin chemistry and microbiome dynamics. *BMC biology*, 17, 1-20.
- Byrne, A. Skincare, Selfies, Self-esteem and Masculinity.
- Firebase(2024). Cloud Storage for Firebase. Diambil pada tanggal 26 Maret 2024 di <https://firebase.google.com/docs/storage>
- Ganatra, S. (2018). *React Router Quick Start Guide: Routing in React Applications Made Easy*. Packt Publishing Ltd.
- Kaehler, A., & Bradski, G. (2016). *Learning OpenCV 3: computer vision in C++ with the OpenCV library*. " O'Reilly Media, Inc."
- Liang, W. (2020, June). Toxicity and effect of chemicals in skin care products on human health. In *IOP Conference Series: Earth and Environmental Science* (Vol. 512, No. 1, p. 012081). IOP Publishing.
- MediaPipe (2024). Face landmark detection guide. Diambil pada tanggal 24 Maret 2024 di https://developers.google.com/mediapipe/solutions/vision/face_landmarker
- MediaPipe (2023). MediaPipe Solutions guide. Diambil pada tanggal 24 Maret 2024 di <https://developers.google.com/mediapipe/solutions/guide>
- Oktaviani, E., & Budiyanto, U. (2023, October). SISTEM PAKAR DIAGNOSA PENYAKIT KULIT BERBASIS WEB MENGGUNAKAN METODE FORWARD CHAINING. In *Prosiding Seminar Nasional Mahasiswa Fakultas Teknologi Informasi (SENAFTI)* (Vol. 2, No. 2, pp. 1119-1126).
- Sirapanji, S., & Hansun, S. (2014). Rancang bangun aplikasi sistem pakar untuk menangani masalah kecantikan pada wajah menggunakan metode decision tree. *Ultimatics: Jurnal Teknik Informatika*, 6(1), 9-14.
- Thompson, M. (2011). *Getting Started with GEO, CouchDB, and Node.js*. " O'Reilly Media, Inc."
- Tohirin. (2020). PENERAPAN KEAMANAN REMOTE SERVER MELALUI SSH DENGAN KOMBINASI KRIPTOGRAFI ASIMETRIS DAN AUTENTIKASI DUA LANGKAH. *Jurnal Teknologi Informasi*. 4. 133-138.
- Vipul, A. M., & Sonpatki, P. (2016). *ReactJS by Example-Building Modern Web Applications with React*. Packt Publishing Ltd.

Lampiran

website: <http://skincarerecomendation.southeastasia.cloudapp.azure.com>

artificial intelligence: <https://github.com/Alficopermana/FaceExtract>

front-end: <https://github.com/deusgvrl/AISkincare/tree/dev>