

BAB V

IMPLEMENTASI DAN PENGUJIAN SISTEM

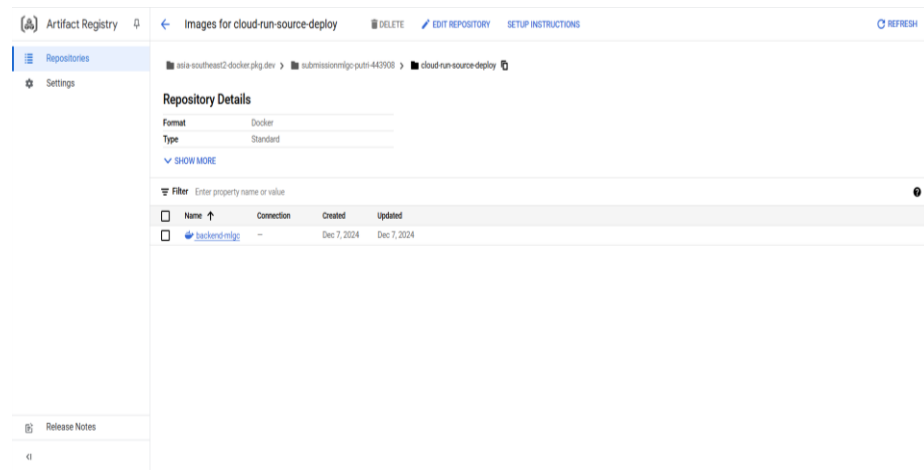
5.1 HASIL IMPLEMENTASI

Implementasi aplikasi scancer mencakup pengintegrasian berbagai layanan *google cloud platform* untuk mendukung proses prediksi kanker kulit berbasis *cloud*. Berikut adalah hasil implementasi yang dicapai :

5.1.1 Model Deployment

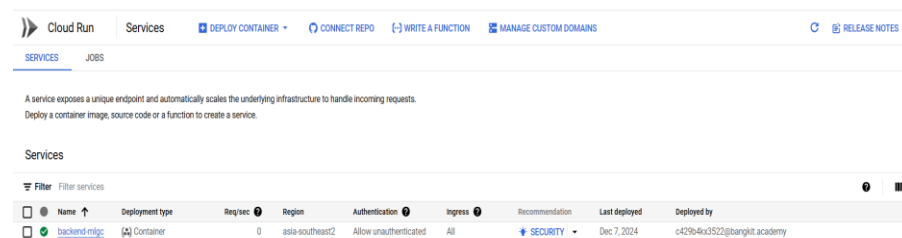
Model *machine learning* yang dikembangkan dalam aplikasi dikembangkan oleh tim *ML*. model prediksi kanker kulit yang mengidentifikasi berdasarkan gambar. Langkah – langkah dalam proses ini mencakup :

- Pembuatan image aplikasi dengan *Docker*: Aplikasi dan model *machine learning* dikemas ke dalam *container Docker* untuk memastikan konsistensi lingkungan antara pengembangan lokal dan produksi.
- Mengunggah image ke *Google Container Registry*: *Image Docker* yang telah dibuat diunggah ke *Google Container Registry*, layanan penyimpanan *image container* dari *Google Cloud*.



Gambar 5.1 Artifact Registry

- *Deploy ke Google Cloud Run:* Aplikasi *container* kemudian di *deploy* ke *Google Cloud Run*. *Cloud Run* memungkinkan aplikasi dijalankan dalam *container* secara *serverless*, memastikan skalabilitas dan pengelolaan beban secara otomatis.

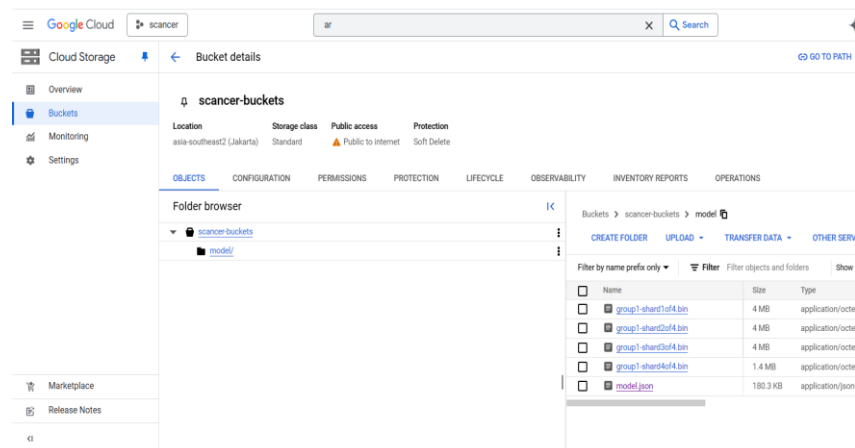


Gambar 5.2 Cloud Run

5.1.2 Penyimpanan

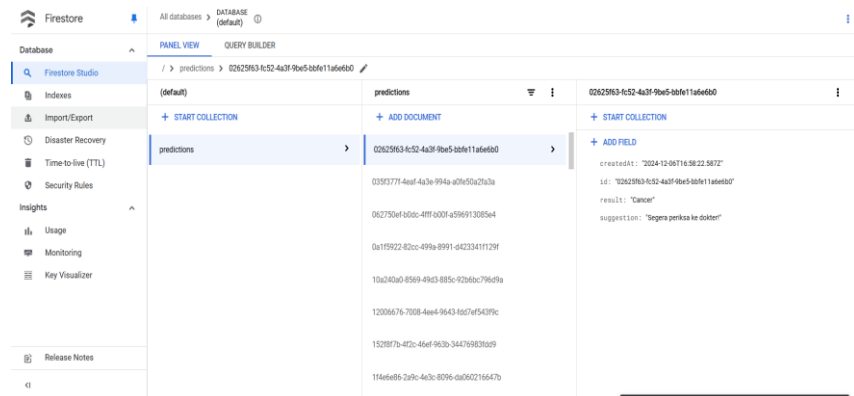
Penyimpanan data dalam aplikasi sancer dilakukan dengan menggunakan dua layanan utama dari *google cloud* :

- *Google Cloud Storage* : Model disimpan di *Google Cloud Storage*, layanan penyimpanan objek yang menawarkan skalabilitas tinggi dan mudah diakses yang mudah. Penyimpanan model di *cloud storage* memungkinkan model sudah diakses dengan cepat dan aman oleh aplikasi.



Gambar 5.3 Cloud Storage

- *Firestore* : Data pengguna disimpan menggunakan *firestore*, layanan database *NoSQL* dari *google cloud* yang dirancang untuk menangani data secara *real-time*. *Firestore* mendukung sinkronisasi data dengan cepat, memastikan data pengguna dapat diakses dan dikelola dengan efisien.

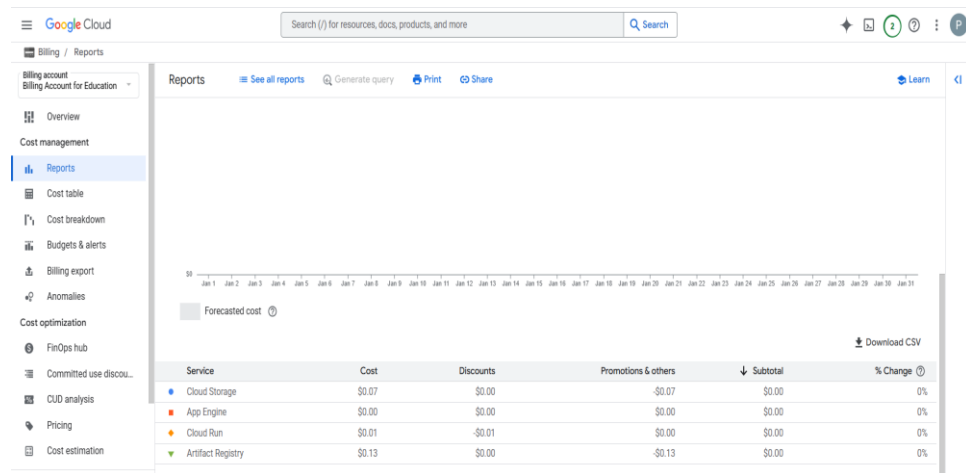


Gambar 5.4 Firestore

Dengan memanfaatkan berbagai layanan google cloud seperti *cloud run*, *cloud storage*, dan *firestore*, aplikasi scancer dapat berjalan dengan efisien. Ini memastikan bahwa fitur aplikasi berfungsi dengan baik dan mampu memenuhi kebutuhan pengguna dengan performa yang optimal.

Setelah *deployment backend* dan model *machine learning* di *google cloud*, integrasi dengan *frontend* aplikasi scancer dilakukan melalui *URL endpoint* yang dihasilkan dari *google cloud run*. *URL endpoint* prediksi model *machine learning* dari *Google Cloud Run* akan digunakan dalam *frontend* untuk mengirim gambar dan menerima hasil prediksi. Dengan mengintegrasikan *URL endpoint* ini, *frontend* aplikasi dapat berkomunikasi dengan *backend* yang di *deploy* di *Google Cloud*, memastikan semua fungsi aplikasi berjalan dengan baik dan memberikan pengalaman pengguna yang mulus dan responsif.

5.1.3 Budget



Gambar 5.5 Budget

Untuk aplikasi scanner, total anggaran biaya yang dikeluarkan didukung oleh kredit sebesar \$21 yang diberikan oleh *Bangkit Academy* untuk penggunaan *google cloud platform* secara keseluruhan. Pengeluaran terbesar berasal dari penggunaan *artifact registry* dan *cloud storage*, yang merupakan layanan utama untuk *image docker* dan menyimpan data. Selain itu, penggunaan *cloud run* menghabiskan biaya sebesar \$0.01.

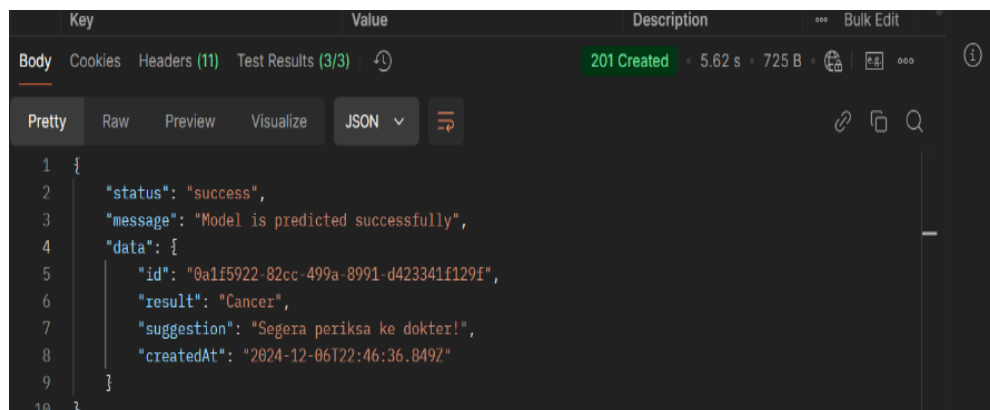
5.2 PENGUJIAN SISTEM

Pada bagian ini, akan dijelaskan secara rinci mengenai proses pengujian yang dilakukan terhadap *endpoint-endpoint API* menggunakan Postman. Pengujian ini bertujuan untuk memastikan bahwa setiap *endpoint* dapat merespons permintaan dengan benar dan menghasilkan output yang sesuai dengan yang diharapkan. Selain itu, pengujian ini juga dilakukan untuk mendeteksi kemungkinan kesalahan atau bug yang dapat memengaruhi kinerja *API*.

Table 5.1 Pengujian API

HTTP Method	Endpoint	Fungsi
POST	/predict	Untuk memulai prediksi jenis kanker
GET	/predict/histories	Untuk mendapatkan riwayat prediksi berdasarkan ID tertentu.
GET	/load-model	Untuk memuat model TensorFlow yang digunakan untuk prediksi.

Berikut adalah hasil pengujian sistem *API* yang dilakukan dengan menggunakan postman untuk memastikan *endpoint* berfungsi dengan baik.

**Gambar 5.6 Pengujian dengan Postman**

1. Pengujian *POST /predict* : *Endpoint* ini digunakan untuk memulai prediksi jenis kanker berdasarkan gambar yang di unggah. Dalam pengujian, gambar yang valid dikirim melalui *request*. Sistem berhasil memproses gambar dan memberikan respon yang sesuai. Hasil respon menunjukkan status *201 Created* yang berarti prediksi berhasil dilakukan. Respon juga mencakup hasil prediksi.

2. Pengujian *GET /predict/histories* : *Endpoint* ini digunakan untuk mendapatkan riwayat prediksi berdasarkan ID tertentu. Dalam pengujian, ID yang valid dikirimkan sebagai parameter untuk mengambil riwayat prediksi. Respon yang diterima berisi informasi mengenai hasil prediksi.
3. Pengujian *GET /load-model* : *Endpoint* ini digunakan untuk memuat model *TensorFlow* yang digunakan dalam prediksi. Status "*success*" mengindikasikan bahwa tidak ada masalah saat model dimuat ke dalam sistem.

5.3 ANALISIS SISTEM YANG DICAPAI OLEH SISTEM

Setelah selesai melakukan implementasi dan pengujian, adapun analisis hasil yang di capai sistem ini adalah sebagai berikut :

1. Kelebihan sistem

Setelah melakukan pengujian terhadap sistem yang dibuat dapat dijabarkan mengenai kelebihan sistem yang dapat dilihat sebagai berikut :

- a. Sistem dapat memproses gambar dengan efisien dan memberikan hasil prediksi.
- b. Arsitektur berbasis *cloud* mendukung kemudahan pengolahan data aplikasi.
- c. Penggunaan layanan *google cloud platform* meningkatkan keandalan sistem.

2. Kekurangan sistem

Setelah melakukan pengujian terhadap sistem yang dibuat dapat dijabarkan mengenai kelebihan sistem yang dapat dilihat sebagai berikut :

- a. Ketergantungan pada koneksi internet untuk mengakses layanan *cloud*.
- b. Keterbatasan pada jenis kanker yang dipredik**