

# Laporan Tugas Besar Mata Kuliah Komputasi Awan

## Identitas Diri

- Nama Lengkap: Hasbi Ash Shiddiqi
- Nomor Induk Mahasiswa (NIM): 2311523014
- Kelas: B
- ID Proyek Google Cloud: komputasi-awan-dsi-unand-2510

## 1. Aplikasi Frontend

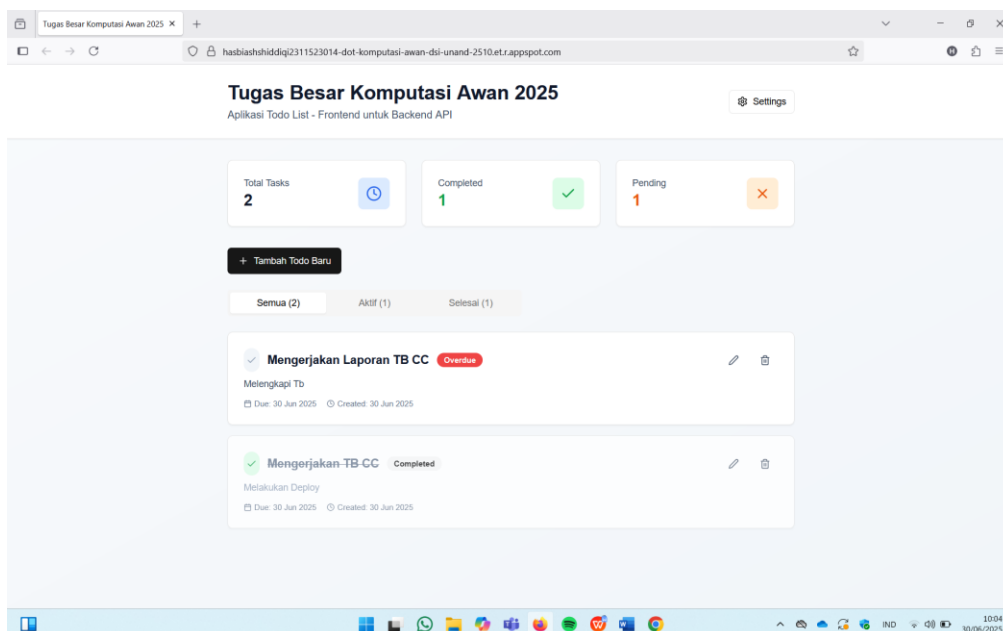
### 1.1. Tautan Aplikasi Frontend

URL Aplikasi: <https://hasbiashshiddiqi2311523014-dot-komputasi-awan-dsi-unand-2510.et.r.appspot.com/>

### 1.2. Layanan Google Cloud yang Digunakan

- Nama Layanan Google Cloud: Google App Engine
- Nama Layanan pada GCP: hasbiashshiddiqi2311523014

### 1.3. Tampilan Hasil Deployment Frontend



#### 1.4. Justifikasi Pemilihan Layanan Frontend

Jelaskan alasan Anda memilih layanan Google Cloud [Layanan yang Digunakan (GCA/GCE/GCR)] untuk mendeploy aplikasi *frontend* dengan menggunakan bahasa sendiri.

Saya memilih Google Cloud App Engine karena layanan ini sangat mudah digunakan dan memudahkan saya dalam meng-online-kan aplikasi frontend. App Engine adalah platform tanpa server (serverless), jadi saya tidak perlu repot mengurus server sendiri—semuanya, mulai dari pengelolaan server, penyesuaian kapasitas, hingga perawatan sistem, ditangani oleh Google. Ini membuat saya bisa lebih fokus mengembangkan aplikasi tanpa pusing memikirkan hal teknis di balik layar.

Selain itu, App Engine mendukung berbagai bahasa pemrograman seperti JavaScript/Node.js, Python, dan lainnya, jadi saya bisa memakai bahasa yang sudah saya kuasai. Proses upload aplikasi ke internet juga mudah, cukup atur beberapa baris di file `app.yaml`, aplikasi saya sudah bisa diakses secara online.

#### 1.5. Tahapan Deployment Frontend

Jelaskan langkah-langkah yang Anda lakukan untuk mendeploy aplikasi *frontend* ke layanan Google Cloud yang dipilih menggunakan bahasa sendiri.

Langkah-Langkah mendeploy *frontend*:

1. Buka Google Cloud Console, lalu pilih App Engine  
Pertama, buka layanan App Engine di Google Cloud agar kita bisa mengatur dan mendeploy aplikasi.
2. Masuk ke bagian Service  
Di sini kita bisa melihat layanan yang sedang berjalan dan nantinya akan muncul aplikasi yang kita deploy.
3. Clone repository frontend dari link yang diberikan  
Buka terminal atau command prompt, lalu clone repo frontend dari GitHub menggunakan perintah `git clone <link-repo>`.

4. (Opsional) Ubah nama folder hasil clone  
Jika perlu, ganti nama folder supaya lebih mudah dikenali.
5. Masuk ke folder proyek dengan perintah `cd <nama-folder>`  
Ini dilakukan agar kita bisa mulai menjalankan perintah-perintah di dalam folder proyek tersebut.
6. Edit file `app.yaml`  
Di file ini, ubah bagian `service`: menjadi nama sesuai dengan nama dan NIM agar tidak bentrok dengan layanan lain.
7. Install semua package yang dibutuhkan  
Jalankan perintah `npm install --legacy-peer-deps` agar semua dependensi terpasang dengan benar.
8. Build aplikasi frontend  
Jalankan perintah `npm run build` untuk membangun proyek, agar siap dideploy.
9. Deploy aplikasi ke App Engine  
Gunakan perintah `gcloud app deploy` untuk mulai proses upload aplikasi ke server Google.
10. Konfirmasi proses deploy  
Saat diminta konfirmasi, ketik `y` lalu tekan Enter.
11. Selesai!  
Aplikasi frontend sudah berhasil dideploy. Selanjutnya kita tinggal menunggu backend-nya juga selesai dideploy agar semuanya bisa berjalan dengan baik.

## **2. Aplikasi Backend (API)**

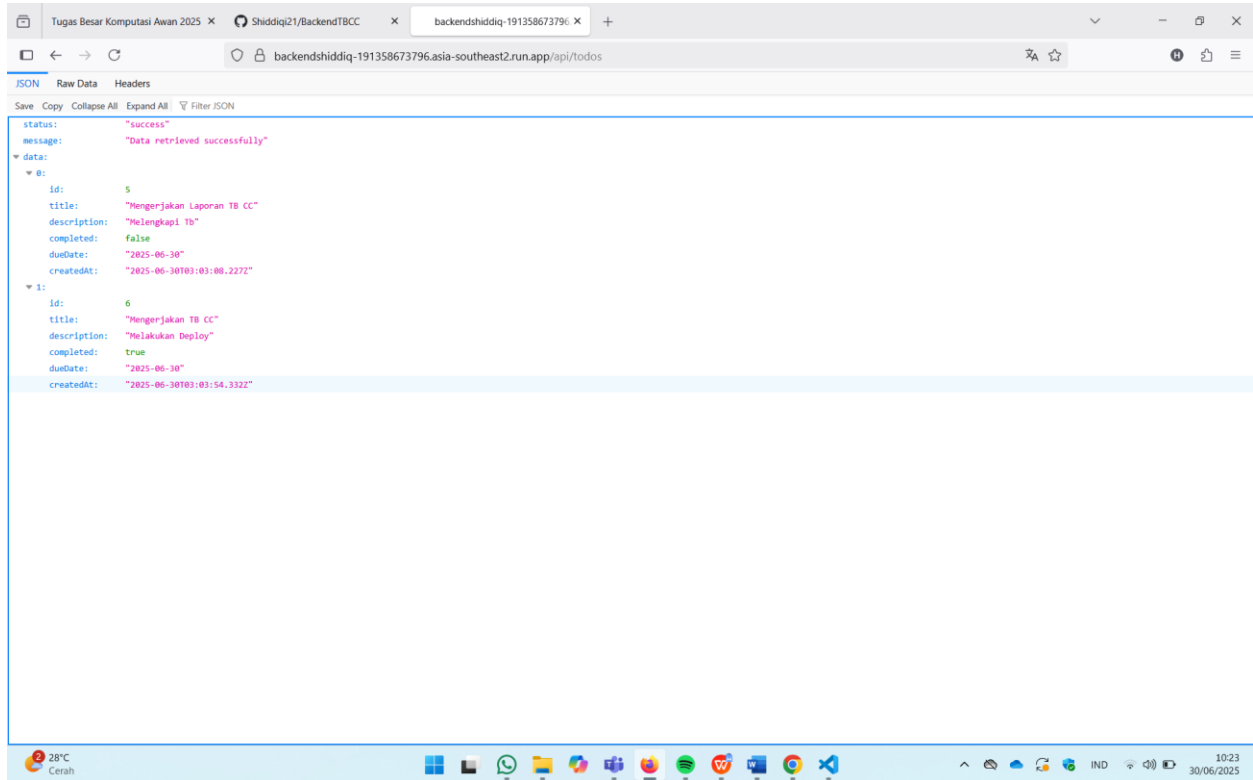
### **2.1. Tautan Aplikasi Backend**

- URL Aplikasi: <https://backendshiddiqi-191358673796.asia-southeast2.run.app>
- Link Repositori Aplikasi Backend (GitHub):  
<https://github.com/Shiddiqi21/BackendTBCC.git>

## 2.2. Layanan Google Cloud yang Digunakan

- Nama Layanan Google Cloud: Google Cloud Run
- Nama Layanan pada GCP: backendshiddiq

## 2.3. Tampilan Hasil Deployment Backend



## 2.4. Justifikasi Pemilihan Layanan Backend

Jelaskan secara singkat alasan Anda memilih layanan Google Cloud [Layanan yang Digunakan (GCA/GCE/GCR)] untuk mendeploy aplikasi *backend* Anda. Gunakan bahasa Anda sendiri untuk menyampaikan pemahaman Anda.

Saya memilih Google Cloud Run untuk mendeploy aplikasi backend karena layanannya fleksibel dan gampang dipakai. Dengan Cloud Run, saya bisa menjalankan aplikasi backend dalam bentuk container tanpa harus ribet mengelola server secara manual. Cukup dengan membungkus (mengemas) aplikasi saya, seperti aplikasi yang dibuat dengan Node.js atau Express, ke dalam Docker container, lalu saya bisa langsung menjalankannya di Cloud Run. Prosesnya cepat, dan saya tidak perlu pusing mikirin urusan infrastruktur.

## 2.5. Tahapan Deployment Backend

Jelaskan langkah-langkah yang Anda lakukan untuk mendeploy aplikasi *backend* ke layanan Google Cloud yang dipilih menggunakan bahasa sendiri.

Langkah-langkah mendeploy *backend*

1. Upload proyek backend ke GitHub  
Pertama, pastikan kode backend sudah diunggah ke GitHub agar bisa diakses lewat Google Cloud.
2. Clone repositori dari GitHub ke terminal Google Cloud  
Buka terminal (Cloud Shell) di Google Cloud, lalu jalankan `git clone <link-repo>` untuk mengambil kode backend.
3. (Opsional) Ubah nama folder proyek  
Bisa di ganti namanya agar lebih mudah dikenali.
4. Masuk ke folder proyek dengan perintah `cd <nama-folder>`
5. Buat file bernama Dockerfile  
File ini berisi instruksi cara menjalankan aplikasi backend dalam bentuk container. Isinya disesuaikan dengan bahasa yang di pakai, misalnya Node.js.
6. Ubah port server menjadi 8080  
Cloud Run hanya menerima aplikasi yang berjalan di port 8080, jadi pastikan server backend kita menggunakan port ini.
7. Buka tab baru, lalu masuk ke layanan Artifact Registry  
Ini tempat menyimpan image dari aplikasi kita.

8. Buat repository baru di Artifact Registry  
Klik "Create Repository".
9. Beri nama repository-nya  
Misalnya: backendshiddiq
10. Ganti lokasi (region) ke asia-southeast2 (Jakarta)  
Setelah itu klik "Create".
11. Kembali ke terminal proyek backend
12. Cek ID Project dengan perintah:  
`gcloud projects list`
13. Atur project aktif dengan perintah:  
`gcloud config set project komputasi-awan-dsi-unand-2510`
14. Atur region default ke Jakarta dengan perintah:  
`gcloud config set run/region asia-southeast2`
15. Build dan push image ke Artifact Registry dengan perintah:  
`gcloud builds submit --tag asia-southeast2-docker.pkg.dev/komputasi-awan-dsi-unand-2510/backendshiddiq/backendshiddiq`  
Ini akan membangun Docker image dari proyek dan mengunggahnya ke repository yang tadi dibuat.
16. Buka tab baru, lalu buka layanan Cloud Run
17. Klik "Deploy Service" untuk mulai proses deploy
18. Pada bagian Container image URL, klik "Select", lalu pilih image yang tadi di-push
19. Ganti region ke asia-southeast2 (Jakarta)
20. Pilih "Allow unauthenticated invocations"  
Agar backend bisa diakses secara publik, misalnya oleh frontend.
21. Klik "Create" untuk memulai proses deploy
22. Setelah proses selesai, akan muncul link dari backend kita
23. Salin link tersebut dan gunakan di bagian frontend (misalnya di file .env)  
Ini agar frontend bisa terhubung dengan backend yang sudah online.

### 3. Video Penjelasan

- Link Video YouTube: <https://youtu.be/ToZqW0hIM6M?feature=shared>

#### **4. Tantangan**

Jelaskan tantangan atau kendala yang Anda alami selama pengerjaan tugas besar ini, baik dalam pengembangan maupun proses deployment.

Selain kebingungan awal terkait alur pengerjaan dan langkah-langkah deployment, saya juga menghadapi beberapa tantangan teknis lainnya selama proses pengembangan dan deployment aplikasi ini.

Salah satu kendala yang cukup terasa adalah kurangnya pemahaman saya terhadap Docker. Karena ini adalah pengalaman pertama saya menggunakan Docker, saya perlu mempelajari bagaimana cara membuat Dockerfile, mengatur port, serta memahami konsep containerization secara umum agar backend dapat dijalankan dengan baik di Google Cloud Run.

Proses build dan deploy ke Cloud Run juga cukup memakan waktu, terutama saat mengunggah image ke Artifact Registry. Beberapa kali saya harus mengulang proses karena konfigurasi yang belum benar.

#### **5. Kesimpulan**

Ringkaslah pembelajaran dan pencapaian utama Anda dari tugas besar ini.

Melalui tugas besar ini, saya mendapatkan pemahaman yang lebih dalam tentang Google Cloud Platform (GCP), khususnya dalam hal cloud computing dan proses deployment aplikasi. Sebelumnya saya belum familiar dengan GCP, namun setelah menyelesaikan tugas ini, saya jadi lebih mengerti cara kerja berbagai layanan seperti App Engine dan Cloud Run, serta fungsi masing-masing.

Saya juga belajar bagaimana memilih layanan GCP yang tepat untuk kebutuhan frontend dan backend, serta memahami proses teknis seperti membuat Dockerfile, konfigurasi app.yaml, hingga penggunaan Artifact Registry. Tugas ini tidak hanya meningkatkan kemampuan teknis saya, tapi juga melatih saya untuk belajar mandiri, bekerja sama dalam tim, dan menyelesaikan masalah secara bertahap.

