

LAPORAN PRAKTIKUM TEKNOLOGI CLOUD COMPUTING

**IMPLEMENTASI DEPLOYMENT APLIKASI CATATAN
MENGUNAKAN DOCKER DAN GOOGLE CLOUD RUN
PRAKTIKUM TEKNOLOGI CLOUD COMPUTING IF - G**



Disusun oleh

Nama : Gustansyah Dwi Putra Sujanto

NIM : 123220210

**PROGRAM STUDI INFORMATIKA
JURUSAN INFORMATIKA
FAKULTAS TEKNIK INDUSTRI
UNIVERSITAS PEMBANGUNAN NASIONAL “VETERAN”
YOGYAKARTA
2025**

HALAMAN PENGESAHAN

LAPORAN PRAKTIKUM

MIGRASI DATABASE KE CLOUD MEMANFAATKAN CLOUD STORAGE *DAN CLOUD SQL*

PRAKTIKUM TEKNOLOGI CLOUD COMPUTING IF - G

Disusun Oleh :

Gustansyah Dwi Putra Sujanto 123220210

Telah diperiksa dan disetujui oleh Asisten Praktikum teknologi Cloud Computing

Pada tanggal :

Menyetujui.

Asisten Praktikum

Asisten Praktikum

Muhammad Rafli
NIM 123210078

Raditya Haikal
NIM 123210062

KATA PENGANTAR

Puji syukur penulis panjatkan ke hadirat Tuhan Yang Maha Esa atas rahmat-Nya sehingga laporan praktikum berjudul **“Implementasi Deployment Aplikasi Catatan Menggunakan Docker dan Google Cloud Run”** ini dapat diselesaikan dengan baik.

Laporan ini merupakan bentuk penerapan materi mengenai containerisasi aplikasi menggunakan Docker serta deployment melalui layanan Cloud Run pada Google Cloud Platform (GCP). Fokus utama dalam tugas ini adalah membangun dan menjalankan aplikasi catatan berbasis frontend dan backend secara cloud-native.

Penulis menyadari bahwa laporan ini masih memiliki kekurangan. Oleh karena itu, saran dan masukan yang membangun sangat diharapkan. Semoga laporan ini dapat memberikan manfaat bagi pembaca sebagai referensi dalam memahami implementasi Docker dan Cloud Run.

Yogyakarta, 30 April 2025

Penulis

DAFTAR ISI

HALAMAN PENGESAHAN	2
KATA PENGANTAR	iii
DAFTAR ISI	iv
DAFTAR GAMBAR	v
BAB I	
PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Tujuan	2
1.4 Manfaat	2
BAB II	
TINJAUAN LITERATUR	3
2.1 Web Service dan Restful API	3
2.2 Express.js sebagai Framework Backend	3
2.3 Sequelize sebagai ORM untuk MySQL	3
BAB III	
METODOLOGI	4
3.1 Analisis Permasalahan	4
3.2 Perancangan Solusi	4
BAB IV	
HASIL DAN PEMBAHASAN	11
4.1 Hasil	11
4.2 Pembahasan	12
BAB V	
PENUTUP	13
5.1 Kesimpulan	13
5.2 Saran	13
DAFTAR PUSTAKA	14

DAFTAR GAMBAR

Gambar 3.1 – Export Database dari phpMyAdmin.

Gambar 3.2 – Create Instance.

Gambar 3.3 – Memilih DB Engine.

Gambar 3.4 – Menentukan Spesifikasi Instance.

Gambar 3.5 – Menentukan Spesifikasi Instance.

Gambar 3.6 – Membuat Database Baru.

Gambar 3.7 – Membuat Bucket.

Gambar 3.8 – Menentukan Spesifikasi Bucket.

Gambar 3.9 – Unggah File DB “notes.sql”.

Gambar 3.11 – Pilih File yang Akan Diimpor.

Gambar 3.12 – Pilih Database yang Akan Diisi File yang Diimpor.

Gambar 3.13 – database.js dengan IP Instance.

Gambar 4.1 – Menampilkan Isi Tabel Setelah Database di CloudShell.

Gambar 4.2 – Backend Tersambung ke Database Google Cloud.

BAB I

PENDAHULUAN

1.1 Latar Belakang

Latar Belakang Perkembangan teknologi komputasi awan (cloud computing) telah mendorong kemudahan dalam pengembangan, deployment, dan skala aplikasi modern. Salah satu platform yang memudahkan hal ini adalah Google Cloud Platform (GCP) dengan berbagai layanan seperti Cloud Run dan Cloud Build. Docker juga menjadi teknologi penting dalam dunia pengembangan perangkat lunak karena memungkinkan developer membuat, mengemas, dan menjalankan aplikasi dalam container yang portabel dan konsisten di berbagai lingkungan. Tugas ini disusun untuk mengimplementasikan proses deployment backend menggunakan Cloud Run serta menjalankan frontend melalui Docker di Cloud Shell.

1.2 Rumusan Masalah

1. Bagaimana cara membuat image backend dan frontend aplikasi catatan?
2. Bagaimana proses deployment backend ke Cloud Run?
3. Bagaimana cara menjalankan frontend menggunakan Docker di Cloud Shell?

1.3 Tujuan

Tujuan dari praktikum ini adalah:

1. Mengimplementasikan konsep containerisasi menggunakan Docker.
2. Menerapkan deployment backend ke layanan Cloud Run.
3. Menjalankan frontend melalui container di Cloud Shell.

1.4 Manfaat

Manfaat yang diharapkan dari implementasi aplikasi ini antara lain:

1. Mahasiswa memahami proses containerisasi dan deployment aplikasi cloud-native.
2. Meningkatkan keterampilan penggunaan GCP dan Docker.
3. Memberikan pengalaman praktis membangun dan menjalankan aplikasi berbasis microservice.

BAB II

TINJAUAN LITERATUR

2.1 Cloud Computing

Komputasi Awan Cloud computing adalah model layanan komputasi berbasis internet yang menyediakan sumber daya komputasi secara dinamis dan sesuai kebutuhan.

2.2 Google Cloud Platform (GCP)

Platform Layanan Cloud Google GCP adalah layanan cloud publik dari Google yang menyediakan berbagai layanan untuk komputasi, penyimpanan, pembelajaran mesin, serta pengembangan dan deployment aplikasi.

2.3 Docker

Teknologi Containerisasi Docker adalah platform open-source untuk mengotomatisasi proses deployment aplikasi dalam container. Container memungkinkan aplikasi berjalan secara konsisten di berbagai lingkungan.

2.4 DCloud Run

Layanan Hosting Container Cloud Run adalah layanan dari GCP yang memungkinkan deployment container berbasis HTTP secara otomatis, terkelola penuh, dan dapat diskalakan secara otomatis.

2.5 Frontend dan Backend

Arsitektur Aplikasi Frontend adalah bagian dari aplikasi yang berinteraksi langsung dengan pengguna, sedangkan backend menangani logika bisnis, database, dan komunikasi server.

2.6 App Engine

Platform Aplikasi Terkelola App Engine adalah platform PaaS dari GCP yang memungkinkan developer mengembangkan dan menghosting aplikasi tanpa mengelola infrastruktur secara langsung.

2.7 Cloud Build

Layanan Build Otomatis Cloud Build memungkinkan developer membuat dan menguji container dari kode sumber secara otomatis serta mendukung CI/CD untuk proses integrasi dan deployment.

BAB III

METODOLOGI

3.1 Analisis Permasalahan

Permasalahan utama yang dianalisis adalah bagaimana membangun dan menjalankan sistem aplikasi catatan menggunakan container dan layanan cloud. Aplikasi harus dapat dijalankan secara terpisah untuk frontend dan backend dengan teknologi Docker dan GCP.

3.2 Perancangan Solusi

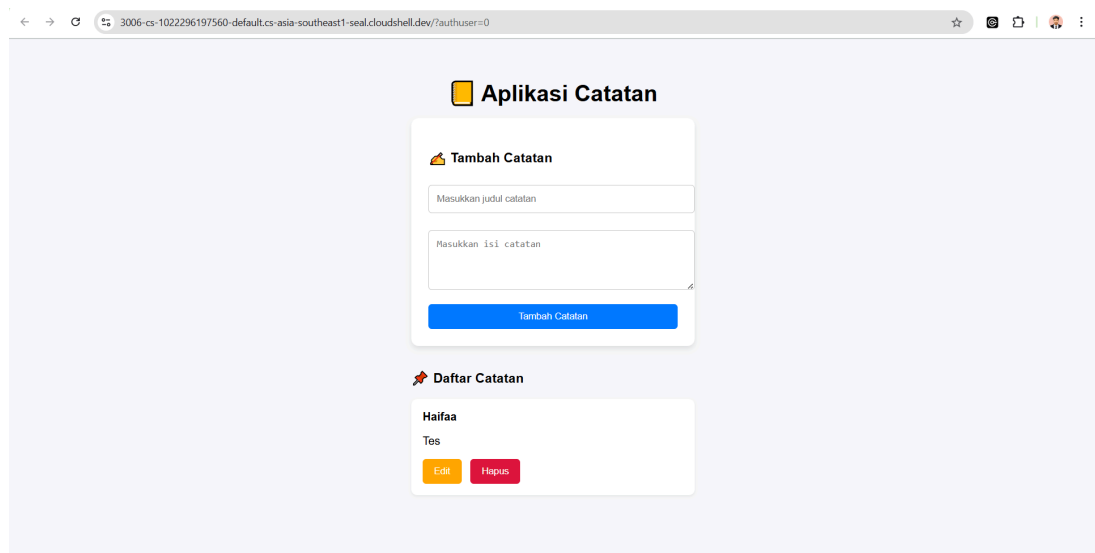
1. Membuat direktori proyek dengan struktur `backend/` dan `frontend/`.
2. Backend dibuat menggunakan Node.js dan Express, disertai Dockerfile.
3. Frontend dibuat dengan HTML, CSS, dan JavaScript, serta dijalankan menggunakan Express server dalam Docker.
4. Backend di-deploy ke Cloud Run:
 - Login ke GCP
 - Build image: `gcloud builds submit --tag gcr.io/[PROJECT_ID]/notes-backend`
 - Deploy: `gcloud run deploy notes-backend --image gcr.io/[PROJECT_ID]/notes-backend --platform managed --region asia-southeast2 --allow-unauthenticated --port 8080`
5. Frontend dijalankan via Docker di Cloud Shell:
 - `docker build -t notes-frontend .`
 - `docker run -d -p 3006:3000 notes-frontend`
6. Web Preview dibuka pada port 3006.

BAB IV

HASIL DAN PEMBAHASAN

4.1 Hasil

Hasil Aplikasi backend berhasil di-deploy ke Cloud Run dan dapat diakses melalui URL publik. Frontend berhasil dijalankan di container Docker dan ditampilkan melalui fitur Web Preview Cloud Shell. Antarmuka aplikasi memungkinkan pengguna menambahkan, menampilkan, mengedit, dan menghapus catatan.



Gambar 4.1 Tampilan antarmuka aplikasi catatan

```
PS C:\Users\Gustansyah Dwi Putra\notes-app\backend> gcloud builds submit --tag gcr.io/prak-g-kelompok12/notes-backend
Creating temporary archive of 3 file(s) totalling 697 bytes before compression.
Uploading tarball of [...] to [gs://prak-g-kelompok12_cloudbuild/source/1749079426.346219-5a4862199465472eba4461786aa11d86.tgz]
Created [https://cloudbuild.googleapis.com/v1/projects/prak-g-kelompok12/locations/global/builds/1b88c704-f92a-413c-8b49-3daa29574621].
Logs are available at [ https://console.cloud.google.com/cloud-build/builds/1b88c704-f92a-413c-8b49-3daa29574621?project=462458321784 ].
Waiting for build to complete. Polling interval: 1 second(s).
----- REMOTE BUILD OUTPUT -----
starting build "1b88c704-f92a-413c-8b49-3daa29574621"

FETCHSOURCE
Fetching storage object: gs://prak-g-kelompok12_cloudbuild/source/1749079426.346219-5a4862199465472eba4461786aa11d86.tgz#1749079426594768
Copying gs://prak-g-kelompok12_cloudbuild/source/1749079426.346219-5a4862199465472eba4461786aa11d86.tgz#1749079426594768...
/ [1 files][ 710.0 B/ 710.0 B]
Operation completed over 1 objects/710.0 B.
BUILD
```

Gambar 4.2 Proses build image backend dengan Cloud Build

```

PS C:\Users\Gustansyah Dwi Putra\notes-app\backend> gcloud run deploy notes-backend --image gcr.io/prak-g-kelompok12/notes-backend --platform managed --region asia-southeast2 --allow-unauthenticated --port 8080
Deploying container to Cloud Run service [notes-backend] in project [prak-g-kelompok12] region [asia-southeast2]
OK Deploying... Done.
OK Creating Revision...
OK Routing traffic...
OK Setting IAM Policy...
Done.
Service [notes-backend] revision [notes-backend-00003-x26] has been deployed and is serving 100 percent of traffic.
Service URL: https://notes-backend-462458321784.asia-southeast2.run.app
PS C:\Users\Gustansyah Dwi Putra\notes-app\backend>
PS C:\Users\Gustansyah Dwi Putra\notes-app\backend>

```

Gambar 4.3 Proses deployment backend ke Cloud Run

4.2 Pembahasan

Pembahasan Proses deployment menunjukkan keunggulan container dalam membuat aplikasi portabel dan mudah dijalankan lintas platform. Cloud Run sangat mempermudah deployment backend tanpa perlu konfigurasi server manual. Penggunaan Docker di Cloud Shell juga memberikan fleksibilitas dalam menjalankan frontend tanpa hosting tambahan. Tantangan yang muncul terkait alokasi port dan error port binding berhasil diatasi dengan menggunakan port alternatif dan pembersihan container mati.

BAB V

PENUTUP

5.1 Kesimpulan

Proses migrasi database dari phpMyAdmin ke Google Cloud SQL telah berhasil diselesaikan dengan memanfaatkan Cloud Storage sebagai media penyimpanan sementara. Dengan langkah ini, database kini dapat diakses dengan mudah dan aman melalui berbagai perangkat yang terhubung ke backend aplikasi.

5.2 Saran

Untuk pengembangan selanjutnya, beberapa aspek yang dapat diperbaiki dan ditingkatkan antara lain:

1. Optimasi Performa Cloud SQL dengan menerapkan indeks pada tabel yang sering diakses serta menggunakan strategi partisi database untuk mengurangi beban pada satu instance.

DAFTAR PUSTAKA