

**PENGGUNAAN INFRASTRUKTUR GOOGLE CLOUD
DALAM PENGEMBANGAN APLIKASI REKOMENDASI
PAKAIAN “TRENIN”**

PRAKTEK KERJA LAPANGAN



Oleh :

AGUS BISANA PUTRA

NPM 21081010125

**PROGRAM STUDI INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS PEMBANGUNAN NASIONAL “VETERAN”
JAWA TIMUR**

2024

**LEMBAR PENGESAHAN
PRAKTEK KERJA LAPANGAN**

Judul : **PENGGUNAAN INFRASTRUKTUR GOOGLE CLOUD
DALAM PENGEMBANGAN APLIKASI REKOMENDASI
PAKAIAN “TRENIN”**

Oleh : **AGUS BISANA PUTRA**

NPM : **21081010125**

Telah Diseminarkan Dalam Ujian PKL, pada :
Hari Kamis, Tanggal 18 Januari 2024

Menyetujui

Dosen Pembimbing

Pembimbing Lapangan


bangkt!

Henni Endah Wahanani, ST. M.Kom.

Deti Anggraini Ekawati

NIP. 19780922 2021212 005

NIP. 02022018017

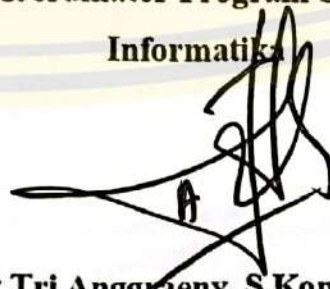
Mengetahui

Dekan

Koordinator Program Studi

Fakultas Ilmu Komputer

Informatika



Prof. Dr. Ir. Noyirina Hendrasarie, M.T. Fetty Tri Anggraeny, S.Kom., M.Kom

NIP. 19681126 199403 2 001

NIP 19820211 2021212 005

SURAT PERNYATAAN

Saya yang bertanda tangan dibawah ini :

Nama : Agus Bisana Putra

NPM : 21081010125

Menyatakan bahwa kegiatan PKL yang saya lakukan memang benar-benar telah kami lakukan di Perusahaan/instansi :

Nama Perusahaan/Instansi : Yayasan Dicoding Indonesia

Alamat : Dicoding Space, Jalan Batik Kumeli No 50,
Kecamatan : Cibeunying Kaler, Kelurahan :
Sukaluyu, RT : 10, RW : 07, Kota Bandung, Jawa
Barat, 40123

Valid, dan perusahaan/instansi tempat kami PKL benar adanya dan dapat dibuktikan kebenarannya. Jika saya menyalahi surat pernyataan yang saya buat maka saya siap menepatkan konsekuensi akademik maupun non-akademik. Berikut surat pernyataan saya buat sebagai syarat laporan PKL di prodi informatika, FIK, UPN "Veteran" Jawa Timur.

Hormat Saya,



Agus Bisana Putra

21081010125

Judul	: Penggunaan Infrastruktur Google Cloud dalam Pengembangan Aplikasi Rekomendasi Pakaian “TrenIn”
Studi Kasus	: Yayasan Dicoding Indonesia
Penulis	: Agus Bisana Putra
Pembimbing	: Henni Endah Wahanani, ST. M.Kom.

Abstrak

Dalam studi empiris yang melibatkan 57 partisipan berusia 19-25 tahun dari berbagai latar belakang agama dan pekerjaan, termasuk mahasiswa dan karyawan, kami bertujuan untuk menangkap demografi yang luas. Studi ini mengungkapkan bahwa 82% responden memberikan prioritas pada pakaian untuk nilai estetikanya, diikuti oleh pertimbangan kognitif (14%) dan implikasi sosial (4%). Trend ini di kalangan populasi muda menekankan perlunya aplikasi fashion yang sejalan dengan preferensi ekspresi diri mereka. Faktor penentu dalam pemilihan pakaian meliputi kombinasi warna (64%), kenyamanan (22%), dan kesesuaian untuk aktivitas atau acara tertentu (14%). Temuan ini membuka jalan untuk pengembangan aplikasi fashion yang dapat membantu dengan membuat rekomendasi pakaian berdasarkan preferensi individual. Dengan demikian, hadirnya TrenIn sebagai sebuah solusi dalam memberikan rekomendasi pakaian sehingga faktor keterbatasan waktu dan pengetahuan dalam pemilihan pakaian dapat diminimalisir.

Aplikasi TrenIn menggunakan infrastruktur Google Cloud sebagai wadah back-end aplikasi. Salah satu layanan Google Cloud yang dimanfaatkan yaitu Cloud Run yang digunakan untuk proses deploy API. Pemilihan Cloud Run sebagai layanan komputasi didasarkan pada kemudahan serta efisiensi biaya. Selain Cloud Run, aplikasi TrenIn juga menggunakan Firebase Cloud Storage sebagai wadah penyimpanan serta Firebase Authenticator untuk mengatur proses login pengguna. Penggunaan infrastruktur Cloud memungkinkan tim untuk mengatur semua layanan yang digunakan sesuai dengan kebutuhan aplikasi.

Dari infrastruktur yang sudah ditetapkan, aplikasi TrenIn menggunakan dua API yang di-deploy menggunakan Cloud Run. API pertama bernama storage-api digunakan untuk mengatur penyimpanan dan pengambilan data atau gambar dari firebase cloud storage. Gambar yang disimpan dalam firebase cloud storage meliputi, foto profil pengguna, pakaian-pakaian pengguna, dan hasil rekomendasi pakaian pengguna. API kedua bernama image-predict-api bertugas untuk menjalankan logika rekomendasi pakaian, API ini akan berjalan ketika pengguna telah menginputkan satu gambar yang akan dijadikan patokan kemudian patokan tadi akan dideteksi klasifikasi jenis dan warnanya lalu proses rekomendasi dapat berjalan berdasarkan data jenis dan warna dari gambar patokan tadi. Hasil dari rekomendasi akan disimpan dalam folder outfits.

Kata Kunci: Google Cloud, API, efisiensi biaya, fashion, rekomendasi.

Abstract

In an empirical study involving 57 participants aged 19-25 from diverse religious and occupational backgrounds, including students and employees, we aimed to capture a broad demographic. The study revealed that 82% of respondents prioritize clothing for its aesthetic value, followed by cognitive considerations (14%) and social implications (4%). This trend among the young population emphasizes the need for fashion applications that align with their self-expression preferences. Determining factors in clothing selection include color combinations (64%), comfort (22%), and suitability for specific activities or events (14%). These findings pave the way for the development of fashion applications that can provide clothing recommendations based on individual preferences. Hence, TrenIn emerges as a solution to minimize time and knowledge limitations in clothing selection.

The TrenIn application utilizes Google Cloud infrastructure as the backend container. One of the Google Cloud services utilized is Cloud Run, employed for deploying the API processing. The choice of Cloud Run as a computing service is based on its ease of use and cost efficiency. In addition to Cloud Run, the TrenIn application also leverages Firebase Cloud Storage as a storage container and Firebase Authenticator to manage the user login process. The use of Cloud infrastructure enables the team to configure all services according to the application's needs.

From the established infrastructure, the TrenIn application uses two APIs deployed using Cloud Run. The first API, named storage-api, manages the storage and retrieval of data or images from Firebase Cloud Storage. Images stored in Firebase Cloud Storage include user profile photos, user clothing, and the results of clothing recommendations. The second API, named image-predict-api, is responsible for running the logic of clothing recommendations. This API is triggered when a user inputs an image, which serves as a reference. The reference image is then classified in terms of type and color, and the recommendation process is carried out based on this data. The results of the recommendations are stored in the outfits folder.

Keywords: *Google Cloud, API, cost efficient, fashion, recommendation.*

KATA PENGANTAR

Dengan penuh rasa syukur, penulis ingin mengungkapkan terima kasih kepada Tuhan Yang Maha Esa atas berkah, rahmat, dan petunjuk-Nya yang senantiasa membimbing langkah penulis. Dengan izin-Nya, penulis berhasil menyelesaikan program Studi Independen Bersertifikat di Bangkit Academy 2023 Batch 2 oleh Google, GoTo, dan Traveloka - Cloud Computing Path yang diselenggarakan oleh Yayasan Dicoding Indonesia. Tak lupa, ucapan terima kasih juga disampaikan kepada Ibu Henni Endah Wahanani, ST, M.Kom, yang telah menjadi dosen wali dan pembimbing selama perjalanan di Bangkit Academy, tim Bangkit Academy 2023 Batch 2, dan Kak Ellyas Immanuel Sinaga sebagai mentor dari CC-04.

Dengan selesainya program studi independen Bangkit Academy 2023, penulis berharap bahwa aplikasi dan ilmu yang diperoleh dapat memberikan kontribusi positif dalam perkembangan teknologi di Indonesia.

DAFTAR ISI

KATA PENGANTAR.....	i
DAFTAR ISI.....	ii
DAFTAR TABEL	iii
DAFTAR GAMBAR.....	iv
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Tujuan Praktek Kerja Lapangan.....	2
1.4 Manfaat.....	3
BAB II GAMBARAN UMUM TEMPAT MBKM.....	4
BAB III PELAKSANAAN	8
3.1 Waktu dan Tempat	8
3.2 Pelaksanaan	11
3.2.1 Tinjauan Pustaka	11
3.2.2 Pelaksanaan PKL MBKM.....	30
BAB IV HASIL DAN PEMBAHASAN	34
BAB V PENUTUP.....	60
5.1 Kesimpulan.....	60
5.2 Saran	60
DAFTAR PUSTAKA	61

DAFTAR TABEL

<i>Tabel 3.1.1</i> Jadwal pelaksanaan kegiatan Bangkit.....	9
<i>Tabel 3.2.2.1</i> Anggota tim	30

DAFTAR GAMBAR

<i>Gambar 2.1</i> Struktur organisasi Bangkit Academy	3
<i>Gambar 3.2.2.1</i> Use case TrenIn	31
<i>Gambar 3.2.2.2</i> Figma TrenIn	33
<i>Gambar 4.1</i> Arsitektur Google Cloud TrenIn.....	34
<i>Gambar 4.2</i> image-predict-api pada cloud run	34
<i>Gambar 4.3</i> storage-api pada cloud run.....	35
<i>Gambar 4.4</i> Folder output machine learning pada firebase cloud storage	35
<i>Gambar 4.5</i> Folder pengguna pada firebase cloud storage.....	35
<i>Gambar 4.6</i> Flowchart algoritma API rekomendasi pakaian.....	36
<i>Gambar 4.7</i> Source code inisialisasi firebase admin	37
<i>Gambar 4.8</i> Source code konfigurasi express dan multer	37
<i>Gambar 4.9</i> Source code persiapan multer, fileFilter, dan validasi password.....	38
<i>Gambar 4.10</i> Source code fungsi deleteStorageFolder.....	38
<i>Gambar 4.11</i> Source code endpoint upload foto profil.....	39
<i>Gambar 4.12</i> Source code endpoint pengambilan foto profil.....	40
<i>Gambar 4.13</i> Source code endpoint upload gambar	41
<i>Gambar 4.14</i> Source code endpoint pengambilan semua gambar	42
<i>Gambar 4.15</i> Source code pengambilan gambar dengan parameter kategori	43
<i>Gambar 4.16</i> Souce code endpoint upload outfit	45
<i>Gambar 4.17</i> Source code endpoint pengambilan outfit	46
<i>Gambar 4.18</i> Source code endpoint deleteaccount.....	47
<i>Gambar 4.19</i> Fungsi tambahan	47
<i>Gambar 4.20</i> Menjalankan server express.....	48
<i>Gambar 4.21</i> Library dan inisialisasi flask.....	48
<i>Gambar 4.22</i> Inisialisasi firebase admin dan kredensial	48
<i>Gambar 4.23</i> Load model dan inisialisasi label warna dan jenis.....	49
<i>Gambar 4.24</i> Fungsi preprocess, prediksi, dan remove background.....	49
<i>Gambar 4.25</i> Fungsi matching outfit, pengambilan data warna, dan membuat rekomendasi	50
<i>Gambar 4.26</i> Fungsi parse filename, mencari gambar, mengunduh gambar, dan proses gambar.....	51
<i>Gambar 4.27</i> Fungsi upload ke firebase dan mencari outfit.....	52

<i>Gambar 4.28</i> Fungsi token_required dan endpoint utama.....	53
<i>Gambar 4.29</i> Endpoint upload.....	54
<i>Gambar 4.30</i> Endpoint mix-match	55
<i>Gambar 4.31</i> Menjalankan flask.....	55
<i>Gambar 4.32</i> GitHub TrenIn.....	56
<i>Gambar 4.33</i> Detail image-predict-api	58
<i>Gambar 4.34</i> Detail storage-api.....	58