

MAKALAH
DEPLOYMENT DAN DISTRIBUSI APLIKASI RESEPKU
BERBASIS DOCKER DAN DOCKER HUB



Disusun oleh:
Muhammad Rafli Aldian Jamil
(32602300044)

UNIVERSITAS ISLAM SULTAN AGUNG SEMARANG
FAKULTAS TEKNOLOGI INDUSTRI
PRODI TEKNIK INFORMATIKA
2025

BAB I PENDAHULUAN

2.1 Latar Belakang

Di era digital, banyak orang ingin mengelola resep masakan secara digital agar mudah diakses dan tidak hilang. Namun, deployment aplikasi web kerap terkendala perbedaan environment dan konfigurasi. Docker hadir untuk mengemas aplikasi beserta dependensinya dalam container sehingga deployment lebih mudah, konsisten, dan dapat direplikasi. Docker Hub mempermudah distribusi image ke server atau tim. Rumusan Masalah

2.2 Tujuan

1. Membangun aplikasi manajemen resep berbasis web.
2. Melakukan deployment aplikasi menggunakan Docker.
3. Mendistribusikan aplikasi ke Docker Hub.
4. Mendemonstrasikan aplikasi secara lokal.

BAB II LANDASAN TEORI

2.1 Docker

Platform open-source untuk mengemas aplikasi dan dependensinya dalam container yang berjalan konsisten di berbagai environment.

2.2 Docker Hub

Registry online untuk menyimpan dan mendistribusikan image Docker yang dapat diakses dari mana saja.

2.3 MariaDB

Sistem manajemen basis data relasional turunan MySQL dengan performa dan kompatibilitas tinggi.

2.4 Flask

Framework web Python yang ringan, modular, dan fleksibel untuk membangun aplikasi web.

2.5 PhpMyAdmin

Aplikasi web untuk mengelola database MySQL/MariaDB secara visual tanpa perintah SQL manual.

BAB III METODOLOGI

Penelitian ini menggunakan metode kualitatif dengan pendekatan rekayasa perangkat lunak untuk menggambarkan proses pengembangan, deployment, dan distribusi aplikasi secara deskriptif.

3.1 Analisis Kebutuhan

1. Fitur CRUD resep.
2. Aplikasi berjalan dalam container.
3. Database terintegrasi via phpMyAdmin.
4. Deployment dan distribusi mudah.

3.2 Perancangan Sistem

1. Arsitektur multi-container dengan Docker Compose.
2. Struktur folder proyek.
3. Dockerfile untuk aplikasi Flask.
4. docker-compose.yml untuk mengatur semua service.

3.3 Implementasi

1. Membuat kode aplikasi Flask.

2. Membuat Dockerfile dan docker-compose.yml.
3. Build image, testing lokal, push ke Docker Hub.

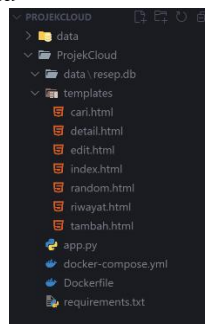
3.4 Pengujian

Aplikasi diuji secara lokal dan dengan image Docker Hub untuk memastikan semua fitur berjalan dan service terintegrasi. Hasil didokumentasikan dengan narasi dan screenshot.

BAB IV IMPLEMENTASI

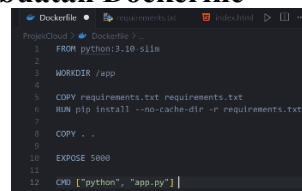
4.1 Striktur Folder Proyek

Struktur folder aplikasi Resepku



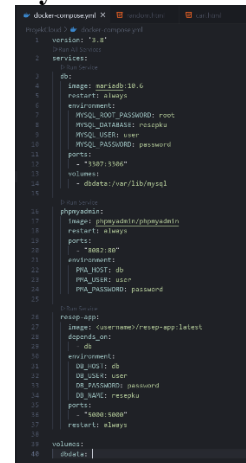
Gambar 4. 1 Struktur folder proyek

4.2 Pembuatan Dockerfile



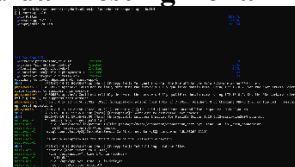
Gambar 4. 2 Isi file Dockerfile

4.3 Pembuatan docker-compose.yml



Gambar 4. 3 Isi file docker-compose.yml

4.4 Build dan Testing Lokal



Gambar 4. 4 Hasil build di terminal



Gambar 4. 5 Tampilan home aplikasi



Gambar 4. 6 Tampilan phpMyAdmin

4.5 Build dan Push ke Docker Hub



Gambar 4. 7 Proses Build Image di Terminal

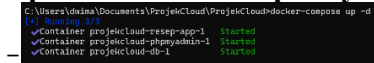


Gambar 4. 8 Proses push image ke Docker Hub



Gambar 4. 9 Repository Aplikasi di Docker Hub

4.6 Update docker-compose.yml

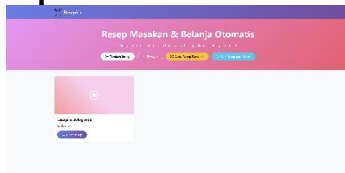


Gambar 4. 10 Tampilan Saat Menjalankan Aplikasi

4.7 Penjelasan Fitur Aplikasi

Aplikasi Resepku memiliki beberapa fitur utama, yaitu:

4.7.1 Tampilan Home



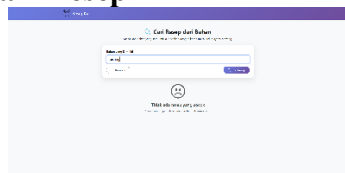
Gambar 4. 11 Tampilan Home Aplikasi

4.7.2 Tambah Resep

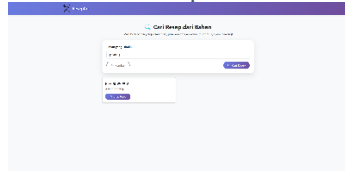


Gambar 4. 12 Tampilan Tambah Resep

4.7.3 Cari Resep



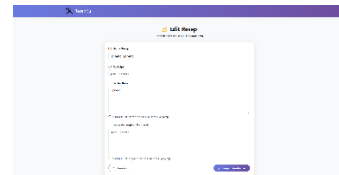
Gambar 4. 13 Tampilan Cari Resep



Gambar 4. 14 Tampilan Hasil Pencarian Resep



Gambar 4. 15 Tampilan Detail Resep



Gambar 4. 16 Form Edit Resep



Gambar 4. 17 Tampilan Hapus Resep



Gambar 4. 18 Tampilan Random Resep



Gambar 4. 19 Tampilan Riwayat Pencarian

BAB V HASIL DAN PEMBAHASAN

5.1 Hasil

Aplikasi berhasil dibuild, diuji lokal dengan Docker Compose, dan dipush ke Docker Hub. Seluruh fitur seperti CRUD resep, pencarian, random, riwayat pencarian, dan pengelolaan database via phpMyAdmin berjalan baik.

5.2 Pembahasan

Proses Deployment: Konsisten di berbagai OS, distribusi mudah lewat Docker Hub, dan manajemen multi-container praktis.

Fitur Aplikasi: Semua fitur berjalan sesuai kebutuhan pengguna.

Kendala & Solusi: Error build/push, port bentrok, dan integrasi service diatasi dengan koreksi konfigurasi.

Evaluasi: Deployment efisien, cepat, konsisten, dengan seluruh fitur berfungsi baik.

BAB VI KESIMPULAN

Berdasarkan hasil implementasi, pengujian, dan pembahasan, diperoleh kesimpulan sebagai berikut:

1. Aplikasi Resepku berhasil dibangun dengan Flask, MariaDB, dan phpMyAdmin, serta seluruh fitur berjalan sesuai kebutuhan.
2. Docker dan Docker Compose mempermudah deployment dengan setup cepat, konsisten, dan minim error.
3. Distribusi melalui Docker Hub efektif untuk membagikan image

aplikasi ke server lain tanpa konfigurasi ulang.

4. Manajemen multi-container berjalan lancar dan efisien dengan Docker Compose.
5. Pengujian menunjukkan semua fitur berfungsi baik, termasuk CRUD resep, pencarian, random, dan riwayat pencarian.
6. phpMyAdmin mempermudah administrasi database secara visual.
7. Dokumentasi detail penting untuk memastikan proses deployment dapat direplikasi oleh siapa pun.
8. Kendala teknis dapat diatasi dengan troubleshooting dan pemahaman tools yang baik.
9. Teknologi container relevan untuk pengembangan modern berbasis DevOps dan CI/CD.
10. Proyek ini membuktikan bahwa deployment aplikasi web bisa dilakukan secara efisien, terstandarisasi, dan siap dikembangkan lebih lanjut.

DAFTAR PUSTAKA

Docker Documentation.
<https://docs.docker.com/>
Docker Hub. <https://hub.docker.com/>
Flask Documentation.
<https://flask.palletsprojects.com/>
MariaDB Documentation.
<https://mariadb.com/kb/en/documentation/>
phpMyAdmin Documentation.
<https://www.phpmyadmin.net/docs/>
[Link GitHub Proyek]
https://github.com/Fxlyn/WebApp_Resepku/tree/main/ProjekCloud