

Perancangan dan Implementasi Sistem Informasi Inventaris Barang Berbasis Web Menggunakan Framework CodeIgniter 4

Diajukan untuk memenuhi tugas mata kuliah Cloud Computing

Dosen pengampu : Sam Farisa Chaerul Haviana, ST., M.Kom



DISUSUN OLEH

NAMA :Ahmad Laits Naufal

NIM : 32602300057

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS ISLAM SULTAN AGUNG SEMARANG
TAHUN 2025**

BAB I

Pendahuluan

1.1 Latar Belakang

Inventarisasi barang adalah kegiatan penting dalam manajemen aset suatu organisasi. Pada sistem manual, pencatatan data barang sering kali memakan waktu dan rawan kesalahan. Untuk itu, dibutuhkan sistem informasi berbasis web yang dapat membantu pengelolaan data barang secara efektif dan efisien. Pada proyek ini, penulis merancang dan membangun sistem informasi inventaris menggunakan framework CodeIgniter 4 dan database MariaDB, serta dikemas dalam lingkungan Docker agar mudah di-deploy di berbagai sistem.

1.2 Rumusan Masalah

- Bagaimana merancang sistem informasi inventaris barang yang dinamis dan mudah digunakan?
- Bagaimana proses CRUD dapat diimplementasikan dengan framework CodeIgniter 4
- Bagaimana menghubungkan sistem dengan database menggunakan Docker?

1.3 Tujuan

- Membangun sistem informasi inventaris berbasis web.
- Mengimplementasikan fitur CRUD (Create, Read, Update, Delete) barang.
- Menerapkan CodeIgniter 4 dan Docker sebagai teknologi utama.

Bab II

PEMBAHASAN

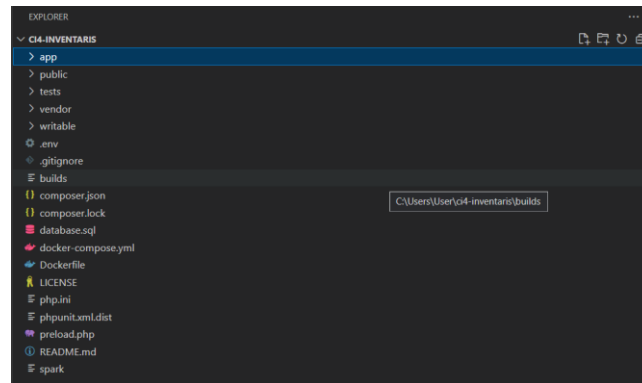
2.1 Persiapan Lingkungan

Install software yang dibutuhkan:

- Docker & Docker Compose
- Code Editor (seperti VS Code)
- Git (jika menggunakan GitHub)

2.2 Membuat struktur

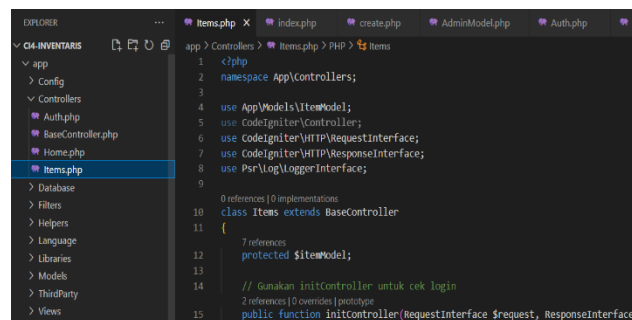
1. proyek aplikasi inventaris berbasis CodeIgniter 4 terdiri dari beberapa bagian penting. Folder `app/` menyimpan logika utama aplikasi, termasuk Controllers, Models, dan Views untuk mengatur alur, pengolahan data, dan tampilan.



Gambar 2. 1 Struktur Proyek

Pada gambar 2.1 struktur proyek aplikasi inventaris berbasis CodeIgniter 4 terdiri dari beberapa bagian penting. Folder app/ menyimpan logika utama aplikasi, termasuk Controllers, Models, dan Views untuk mengatur alur, pengolahan data, dan tampilan.

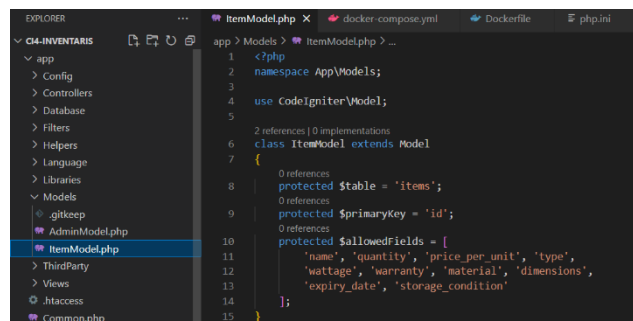
2. Membuat Items.php didalam folder app/Controllers



Gambar 2. 2 app/Controllers/Items.php

Pada gambar 2.2 Items.php mengatur logika aplikasi terkait barang, seperti menampilkan, menambah, mengedit, dan menghapus data.

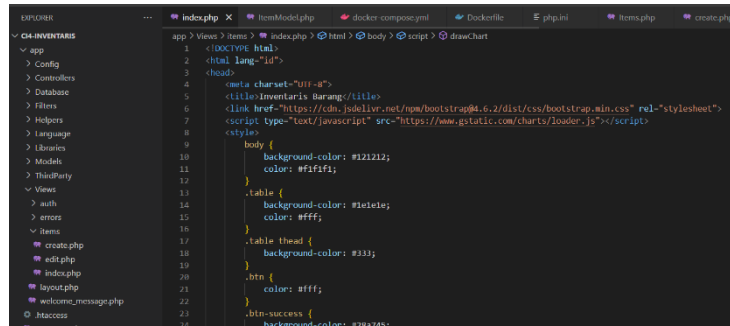
3. Membuat ItemsModel.php didalam folder app/Models



Gambar 2. 3 app/Models/ItemModel.php

Pada gambar 2.3 ItemModel.php bertugas untuk mengakses dan memanipulasi data barang dari database.

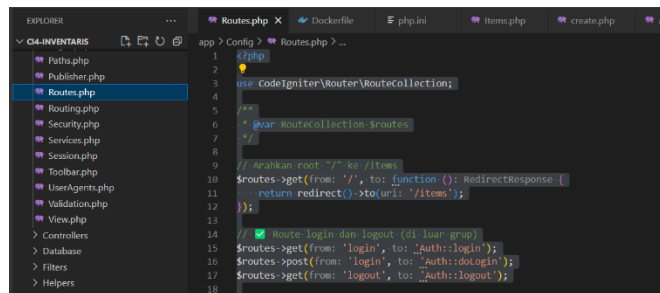
4. Membuat file index.php didalam folder app/Views/items/



Gambar 2. 4 app/Views/items/index.php (berserta create.php, edit.php)

Pada gambar 2.4 index.php Menyediakan tampilan antarmuka pengguna untuk berinteraksi dengan data barang

5. Membuat Routes.php didalam folder app/Config

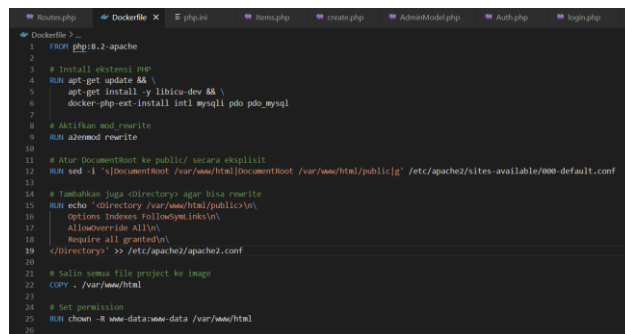


Gambar 2. 5 app/Config/Routes.php

Pada gambar 2.5 routes.php mengatur rute (URL) aplikasi agar permintaan diarahkan ke controller yang sesuai.

2.3 Setup Docker

a. Dockerfile



Gambar 2. 6 Dockerfile

Pada gambar 2.6 Dockerfile adalah file yang berisi instruksi untuk membangun image Docker. File ini digunakan untuk mengatur lingkungan aplikasi, seperti menginstal PHP, Apache, dan ekstensi yang dibutuhkan. Dengan Dockerfile, proses setup server menjadi otomatis dan konsisten di berbagai lingkungan.

b. Docker-compose.yml

```

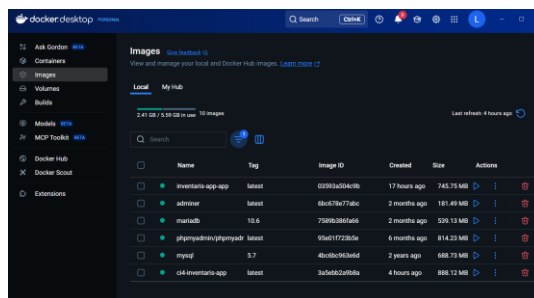
1 version: '3.8'
2
3 services:
4   app:
5     build: .
6     container_name: inventaris_app
7     ports:
8       - "8080:80"
9     depends_on:
10      - db
11
12   db:
13     image: mariadb:10.6
14     container_name: inventaris_db
15     restart: always
16     environment:
17       MYSQL_ROOT_PASSWORD: rootpass
18       MYSQL_DATABASE: inventaris
19       MYSQL_USER: user
20       MYSQL_PASSWORD: userpass
21     ports:
22       - "3306:3306"
23     volumes:
24       - dbdata:/var/lib/mysql
25
26   phpmyadmin:
27     image: phpmyadmin/phpmyadmin

```

Gambar 2. 7 docker-compose.yml

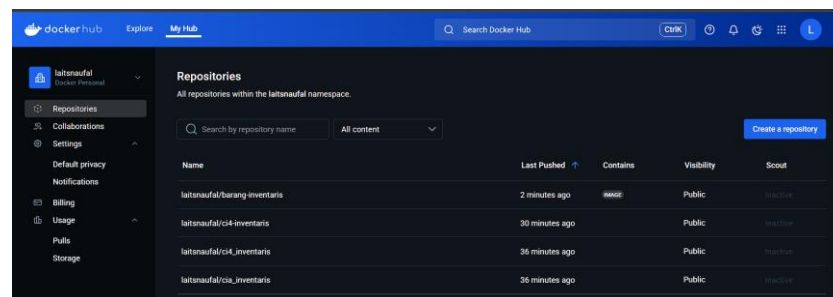
File docker-compose.yml berfungsi untuk mengatur dan menjalankan beberapa container Docker secara bersamaan. Dalam proyek aplikasi, file ini digunakan untuk menghubungkan layanan seperti web server, database, dan phpMyAdmin dalam satu konfigurasi yang mudah dijalankan dengan satu perintah (docker-compose up).

2.4 Docker image dan deploy ke docker hub



Gambar 2. 8 tampilan setelah berhasil di hubungkan ke Docker desktop

Gambar tersebut menunjukkan tampilan Docker Desktop pada tab Images, yang menampilkan daftar image lokal yang tersedia di sistem. Terlihat beberapa image seperti inventaris-app, adminer, mariadb, phpmyadmin, mysql, dan ci4-inventaris-app, yang masing-masing digunakan untuk membangun dan menjalankan aplikasi inventaris berbasis CodeIgniter 4 beserta layanan pendukungnya (seperti database dan phpMyAdmin). Setiap image ditampilkan dengan informasi tag, ID, waktu dibuat, ukuran, serta opsi untuk menghapusnya.



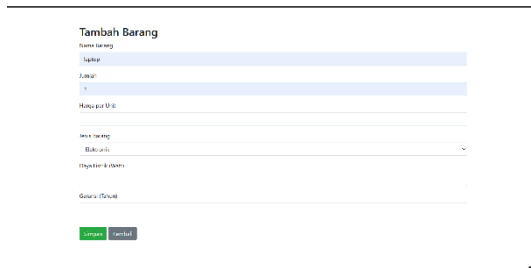
Gambar 2. 9 docker hub

Gambar diatas menampilkan bahwa image barang-inventaris telah berhasil diunggah ke Docker Hub pada akun laitsnaufal dan bersifat public.



Gambar 2. 10 Tampilan halaman login

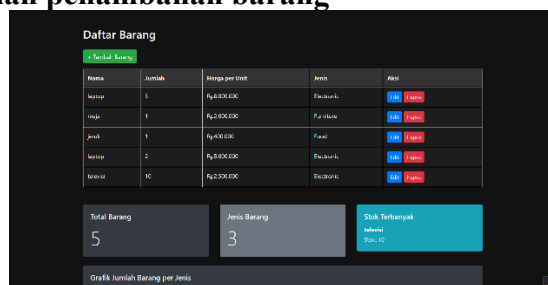
Pada gambar 2.9 merupakan tampilan halaman login saat mengakses port 8080 dengan Username “admin” dan password “admin1234”.



Gambar 2. 11 Tampilan awal aplikasi

Gambar diatas menampilkan form tambah barang pada aplikasi inventaris berbasis web. Form ini memungkinkan pengguna untuk menginput data barang seperti Nama Barang, Jumlah, Harga per Unit, Jenis Barang, Daya Listrik, dan Garansi. Di bagian bawah terdapat tombol Simpan untuk menyimpan data dan tombol Kembali untuk kembali ke halaman sebelumnya.

2.5 Output aplikasi setelah penambahan barang



Gambar 2. 12 Tampilan aplikasi setelah barang ditambahkan

Gambar diatas menunjukkan halaman daftar barang yang menampilkan tabel data inventaris, tombol edit/hapus, ringkasan statistik, dan grafik jenis barang.

3. Penutup

Pembuatan aplikasi inventaris barang berbasis web ini bertujuan untuk memudahkan pengelolaan data barang secara efisien dan terstruktur. Dengan menggunakan framework CodeIgniter 4 serta Docker sebagai lingkungan pengembangan, aplikasi dapat berjalan dengan stabil dan mudah di-deploy. Diharapkan aplikasi ini dapat menjadi solusi praktis dalam pencatatan dan monitoring data barang di lingkungan instansi atau perusahaan. Semoga makalah ini dapat memberikan gambaran yang jelas dan bermanfaat bagi pembaca. Link github <https://github.com/tutelnaufal/CI4-INVENTARIS.git>