

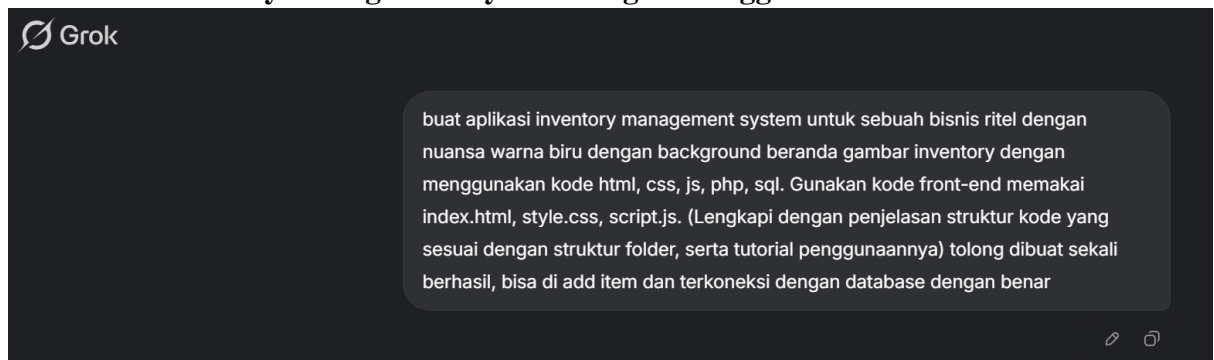
NAMA : NAJWA AULIA ADRIATI

NIM : 12030123140318

KELAS: PENGKODEAN DAN PEMROGRAMAN/F

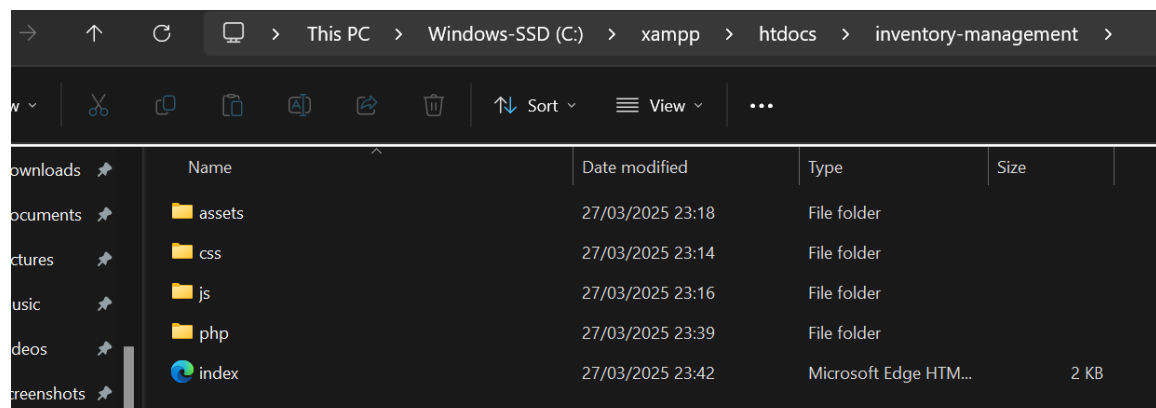
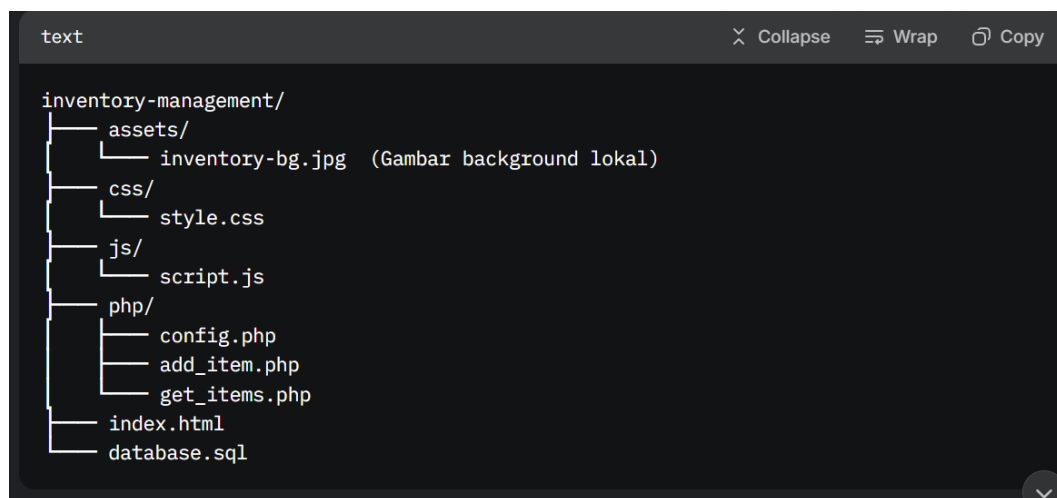
PERTEMUAN 5

- **Membuat inventory management system dengan menggunakan GROK AI**



Pada gambar prompt di atas tertulis bahwa system yang saya inginkan berwarna biru, namun di akhir proses saya meminta GROK AI untuk mengganti warna system dengan nuansa warna coklat.

Struktur folder:



Kode SQL database di phpMyAdmin:

```
1. Database Setup ( database.sql )

Pertama, buat database dan tabel untuk menyimpan data inventory.

sql

CREATE DATABASE inventory_db;
USE inventory_db;

CREATE TABLE items (
  id INT AUTO_INCREMENT PRIMARY KEY,
  name VARCHAR(100) NOT NULL,
  quantity INT NOT NULL,
  price DECIMAL(10, 2) NOT NULL,
  created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);
```

Kode diinput di notepad++

```
3. Frontend - HTML ( index.html )

File utama untuk tampilan antarmuka.

html

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Inventory Management System</title>
  <link rel="stylesheet" href="css/style.css">
</head>
<body>
  <div class="container">
    <h1>Inventory Management System</h1>

    <!-- Form Tambah Item -->
    <div class="form-container">
      <h2>Add New Item</h2>
```

```
4. CSS - Styling ( css/style.css )

Desain dengan nuansa biru dan background inventory.

css

body {
  margin: 0;
  padding: 0;
  font-family: Arial, sans-serif;
  background: url('https://images.unsplash.com/photo-1585314062340-279fb259a4f6?ixlib=
  background-size: cover;
}

.container {
  max-width: 1000px;
  margin: 50px auto;
  padding: 20px;
  background: rgba(255, 255, 255, 0.9);
  border-radius: 10px;
```

5. JavaScript - Interaktivitas (`js/script.js`)

Logika untuk menambah item dan memuat data dari database.

javascript

✕ Collapse ⇅ Wrap 📄 Copy

```
document.addEventListener('DOMContentLoaded', () => {
  loadItems();

  document.getElementById('addItemForm').addEventListener('submit', (e) => {
    e.preventDefault();

    const name = document.getElementById('itemName').value;
    const quantity = document.getElementById('itemQuantity').value;
    const price = document.getElementById('itemPrice').value;

    fetch('php/add_item.php', {
      method: 'POST',
      headers: { 'Content-Type': 'application/x-www-form-urlencoded' },
      body: `name=${name}&quantity=${quantity}&price=${price}`
    })
  })
});
```

2. Konfigurasi Database (`php/config.php`)

File untuk koneksi ke database.

php

✕ Collapse ⇅ Wrap 📄 Copy

```
<?php
$host = "localhost";
$user = "root";
$password = "";
$dbname = "inventory_db";

$conn = mysqli_connect($host, $user, $password, $dbname);

if (!$conn) {
  die("Connection failed: " . mysqli_connect_error());
}
?>
```

• File: `php/get_items.php` :

php

✕ Collapse ⇅ Wrap 📄 Copy

```
<?php
include 'config.php';

// Tambahkan header CORS
header("Access-Control-Allow-Origin: *");
header("Access-Control-Allow-Methods: GET, POST, OPTIONS");
header("Access-Control-Allow-Headers: Content-Type");

$sql = "SELECT * FROM items ORDER BY created_at DESC";
$result = mysqli_query($conn, $sql);

$items = [];
while ($row = mysqli_fetch_assoc($result)) {
  $items[] = $row;
}
```

```
• File: php/add_item.php :

php Collapse Wrap Copy

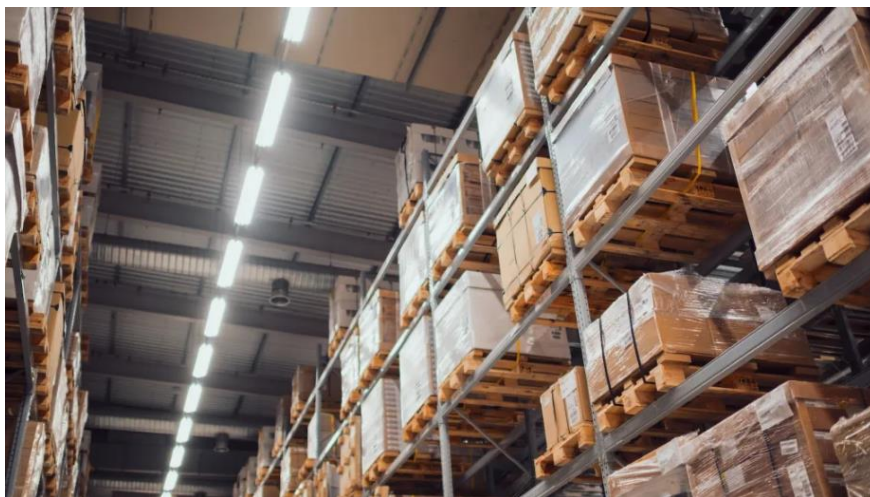
<?php
include 'config.php';

// Tambahkan header CORS
header("Access-Control-Allow-Origin: *");
header("Access-Control-Allow-Methods: GET, POST, OPTIONS");
header("Access-Control-Allow-Headers: Content-Type");

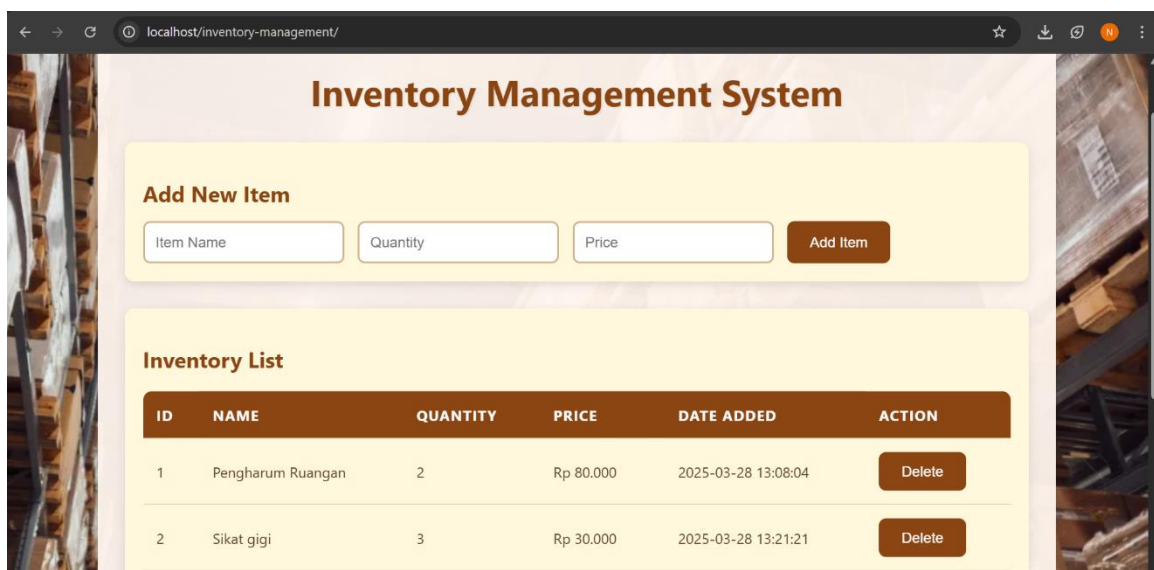
if ($_SERVER['REQUEST_METHOD'] == 'POST') {
    $name = $_POST['name'];
    $quantity = $_POST['quantity'];
    $price = $_POST['price'];

    $sql = "INSERT INTO items (name, quantity, price) VALUES ('$name', '$quantity', '$price')";
    if (mysqli_query($conn, $sql)) {
        echo "Item added successfully!";
    } else {
        echo "Error: " . mysqli_error($conn);
    }
}
```

Gambar untuk background:



Hasil:



Tampilan database di phpMyAdmin:



The screenshot displays the phpMyAdmin interface. On the left sidebar, a tree view shows the database structure: 'New' (expanded), 'information_schema', 'inventory', 'inventory_db' (expanded), 'New' (under inventory_db), 'items' (under New), 'mysql', 'performance_schema', 'perpustakaan', 'phpmyadmin', and 'test'. The main panel shows the 'items' table selected. At the top, there are controls for 'Show all', 'Number of rows' (set to 25), 'Filter rows' (with a search box), and 'Sort by key' (set to None). Below these is an 'Extra options' button. The table data is presented in a grid with columns: 'id', 'name', 'quantity', 'price', and 'created_at'. Each row includes a checkbox and action links for 'Edit', 'Copy', and 'Delete'.

	id	name	quantity	price	created_at
<input type="checkbox"/>	1	Pengharum Ruangan	2	80000.00	2025-03-28 13:08:04
<input type="checkbox"/>	2	Sikat gigi	3	30000.00	2025-03-28 13:21:21
<input type="checkbox"/>	3	Sabun Mandi	2	6000.00	2025-03-28 13:21:56
<input type="checkbox"/>	4	Pasta Gigi	5	50000.00	2025-03-28 13:32:09
<input type="checkbox"/>	5	Sapu	5	150000.00	2025-04-12 09:47:07