

DOKUMENTASI TECHNICAL TEST
SELEKSI STAFF IT DEVELOPER KOKOLA GROUP



Tanggal Pembuatan:

7 Mei 2025

Developer:

Muhammad Ali Yusuf

KOKOLA GROUP

GRESIK

DAFTAR ISI

DAFTAR ISI	i
PENDAHULUAN	1
STRUKTUR PROYEK	2
TEKNOLOGI YANG DIGUNAKAN	3
FITUR APLIKASI.....	4
DETAIL IMPLEMENTASI	5
Soal A: Manajemen Plant Product.....	5
Soal B: Fungsi Pencarian String	6
PANDUAN INSTALASI	7
PANDUAN PENGGUNAAN	8
Soal A: Manajemen Plant Product.....	8
Soal B: Fungsi Pencarian String	8
STRUKTUR DATABASE.....	9
KEAMANAN APLIKASI.....	10
PENUTUP	11

PENDAHULUAN

Proyek ini dikembangkan untuk proses *recruitment* Kokola Group untuk posisi Web Developer yakni membuat sistem manajemen dan relasi antara data Plant (Pabrik) dan Product (Produk). Aplikasi terdiri dari dua bagian utama sesuai dengan permintaan:

1. **Soal A:** Manajemen Plant Product - Aplikasi web untuk mengelola hubungan antara Plant dan Produk
2. **Soal B:** Implementasi algoritma pencarian string sederhana

Kedua bagian aplikasi ini diimplementasikan menggunakan arsitektur MVC (Model-View-Controller) untuk memudahkan pengembangan, pemeliharaan di masa mendatang.

Berikut link repository github beserta video demo aplikasi yang sudah dibuat:

- Repository Github: https://github.com/muhammadaliyusuf/kokola_test
- Vide demo aplikasi:
<https://drive.google.com/drive/folders/152cSfoFeweq3v6DPo7gFr8hzsFQerC01>

STRUKTUR PROYEK

Berikut adalah struktur direktori lengkap dari proyek:

```
kokola_test
├── soal-A/
│   ├── config/
│   │   └── database.php          # File konfigurasi koneksi
├── database
│   ├── controllers/
│   │   └── PlantProductController.php # Controller utama untuk produk
│   ├── models/
│   │   └── PlantProduct.php        # Model untuk tabel 'plant_products'
│   ├── views/
│   │   ├── layouts/
│   │   │   ├── header.php        # Header untuk semua halaman
│   │   │   ├── footer.php        # Footer untuk semua halaman
│   │   └── main.php              # Halaman utama daftar produk
│   ├── assets/
│   │   ├── css/
│   │   │   └── style.css          # Styling tambahan (custom CSS)
│   │   └── js/
│   │       └── script.js          # Script AJAX untuk form tambah data
│   └── index.php                  # Entry point utama aplikasi
└── soal-B/
    └── index.php                  # Implementasi fungsi pencarian string
```

TEKNOLOGI YANG DIGUNAKAN

Proyek ini dikembangkan dengan menggunakan teknologi berikut:

- **Bahasa Pemrograman:**
 - PHP 7.4+
 - JavaScript
 - HTML5
 - CSS3
- **Framework dan Library:**
 - Bootstrap 5.3.0-alpha1 (Frontend)
 - jQuery 3.6.0 (Manipulasi DOM dan AJAX)
- **Database:**
 - MySQL/MariaDB
- **Arsitektur:**
 - MVC (Model-View-Controller)
- **Komunikasi Data:**
 - AJAX untuk operasi asinkron

FITUR APLIKASI

Soal A: Manajemen Plant Product

- Menampilkan daftar Plant dan Produk yang terkait
- Menampilkan produk yang belum memiliki relasi dengan Plant manapun
- Formulir untuk menambahkan produk baru dan mengaitkannya dengan Plant
- Validasi input untuk mencegah duplikasi data
- Operasi CRUD menggunakan AJAX (tanpa refresh halaman)

Soal B: Fungsi Pencarian String

- Implementasi algoritma pencarian substring sederhana
- Pencarian posisi kemunculan substring dalam string target
- Visualisasi hasil pencarian dengan representasi grafis
- Antarmuka pengguna yang responsif dan user-friendly

DETAIL IMPLEMENTASI

Soal A: Manajemen Plant Product

Arsitektur MVC

Aplikasi ini dikembangkan dengan menggunakan arsitektur MVC (Model-View-Controller) yang membagi fungsionalitas aplikasi menjadi tiga komponen utama:

1. Model (PlantProduct.php):

- Menangani interaksi dengan database
- Mendefinisikan struktur data dan operasi-operasi CRUD
- Mengimplementasikan logika bisnis terkait Plant dan Product

2. View (main.php, header.php, footer.php):

- Menampilkan antarmuka pengguna
- Menampilkan data Plant dan Product
- Menampilkan formulir untuk menambahkan data baru

3. Controller (PlantProductController.php):

- Menerima input dari pengguna
- Menghubungkan Model dan View
- Memproses data dan menghasilkan respons yang sesuai

Alur Kerja Aplikasi

1. Tampilan Daftar Plant dan Product:

- Controller mengambil data dari Model
- Data ditampilkan dalam tabel pada View
- Produk dikelompokkan berdasarkan Plant dengan format yang mudah dibaca

2. Penambahan Data Baru:

- Pengguna mengisi formulir dengan nama produk dan Plant terkait
- Form dikirim melalui AJAX ke Controller
- Controller memvalidasi input
- Controller menyimpan data melalui Model jika valid
- Controller mengirimkan respons JSON ke View
- View memperbarui tampilan tabel tanpa refresh halaman

3. Validasi dan Keamanan:

- Input dibersihkan menggunakan fungsi htmlspecialchars dan strip_tags
- Query database menggunakan prepared statements untuk mencegah SQL Injection
- Validasi client-side dan server-side untuk memastikan integritas data

Detail Implementasi Model

Model PlantProduct mengimplementasikan beberapa metode utama:

- getAllGroupedByPlant(): Mengambil data produk yang dikelompokkan berdasarkan Plant
- getProductsWithoutPlant(): Mengambil produk yang belum memiliki relasi dengan Plant
- create(): Membuat relasi baru antara Plant dan Product
- exists(): Memeriksa apakah relasi tertentu sudah ada
- createProduct(): Membuat produk baru jika belum ada di database
- getAllPlants(): Mengambil semua data Plant untuk ditampilkan dalam dropdown

Detail Implementasi Controller

Controller PlantProductController memiliki dua metode utama:

- index(): Menampilkan halaman utama dengan tabel data
- addPlantProduct(): Memproses penambahan data baru melalui AJAX

Soal B: Fungsi Pencarian String

Algoritma Pencarian

Implementasi algoritma pencarian string sederhana yang mencari posisi kemunculan substring dalam string target:

1. Konversi string input dan target ke lowercase untuk pencarian case-insensitive
2. Inisialisasi array kosong untuk menyimpan posisi kemunculan
3. Loop melalui string target dengan panjang string input
4. Periksa apakah substring pada posisi saat ini cocok dengan string input
5. Jika cocok, tambahkan posisi ke array hasil
6. Kembalikan array posisi kemunculan

Fitur Visualisasi

Hasil pencarian divisualisasikan dengan representasi grafis:

- Menampilkan string target

- Menampilkan tanda '^' di bawah string target untuk menunjukkan posisi kemunculan

PANDUAN INSTALASI

Berikut adalah langkah-langkah untuk menginstal dan menjalankan aplikasi:

1. Persiapan Server:

- Pastikan server web (Apache/Nginx) dan MySQL/MariaDB sudah terinstal
- PHP versi 7.4 atau lebih tinggi

2. Persiapan Database:

- Buat database baru dengan nama kokola_test
- Import struktur database dari file SQL yang disediakan

3. Konfigurasi Aplikasi:

- Salin semua file proyek ke direktori web server
- Sesuaikan pengaturan koneksi database di config/database.php jika diperlukan:

```
private $host = 'localhost';      // Sesuaikan dengan host database
private $dbname = 'kokola_test'; // Sesuaikan dengan nama database
private $username = 'root';      // Sesuaikan dengan username database
private $password = '';          // Sesuaikan dengan password database
```

4. Pengujian Instalasi:

- Akses aplikasi melalui browser dengan URL:
http://localhost/kokola_test/soal_a/
- Pastikan tidak ada pesan error yang muncul
- Periksa apakah data Plant dan Product ditampilkan dengan benar

PANDUAN PENGGUNAAN

Soal A: Manajemen Plant Product

1. Melihat Daftar Plant dan Product:

- Buka halaman utama aplikasi
- Tabel akan menampilkan daftar Plant beserta Produk terkait
- Produk tanpa Plant akan ditampilkan dengan tanda "-" pada kolom Plant

2. Menambahkan Produk Baru:

- Klik tombol "Tambah Product" di bagian atas tabel
- Pilih Plant dari dropdown (opsional, bisa dibiarkan kosong)
- Masukkan nama produk pada field "Product Name"
- Klik tombol "Save" untuk menyimpan data
- Tabel akan diperbarui secara otomatis tanpa refresh halaman

Soal B: Fungsi Pencarian String

1. Mencari Substring:

- Buka halaman aplikasi Soal B
- Masukkan teks yang ingin dicari pada field input
- Klik tombol "Search"
- Hasil pencarian akan ditampilkan di bawah formulir

2. Membaca Hasil Pencarian:

- Posisi kemunculan substring akan ditampilkan dalam format angka
- Representasi visual akan menunjukkan posisi dengan tanda '^'
- Jika tidak ada hasil yang ditemukan, akan muncul pesan "No matches found"

STRUKTUR DATABASE

Aplikasi ini menggunakan tiga tabel utama:

1. **m_plant:**

- id (int, primary key): ID unik untuk setiap Plant
- kode (varchar): Kode Plant
- name (varchar): Nama Plant

2. **m_product:**

- id (int, primary key): ID unik untuk setiap Produk
- name (varchar): Nama Produk

3. **m_plant_product:**

- id (int, primary key): ID unik untuk setiap relasi
- id_plant (int, foreign key): Referensi ke tabel m_plant
- id_product (int, foreign key): Referensi ke tabel m_product

KEAMANAN APLIKASI

Aplikasi telah diimplementasikan dengan memperhatikan aspek keamanan:

1. Pencegahan SQL Injection:

- Penggunaan prepared statements untuk semua query database
- Parameter binding untuk semua input pengguna

2. Pencegahan XSS (Cross-Site Scripting):

- Pembersihan input dengan htmlspecialchars dan strip_tags
- Validasi input pada sisi server

3. Validasi Input:

- Validasi form pada sisi client menggunakan JavaScript
- Validasi pada sisi server sebelum penyimpanan data

4. Sanitasi Output:

- Penggunaan htmlspecialchars untuk semua output yang ditampilkan ke pengguna

PENUTUP

Demikian dokumentasi lengkap untuk proyek Technical Test pada proses *recruitment* Kokola Group. Jika ada pertanyaan atau kebutuhan informasi tambahan, silakan hubungi tim pengembang.

Dibuat oleh: Muhammad Ali Yusuf

Tanggal: 7 Mei 2025