

## BAB 4

### ANALISA DATA DAN PEMBAHASAN

Pada bab empat ini maka dilakukanlah pembahasan serta analisis data, dimulai dari pembahasan terlebih dahulu. Pembahasan diawali sesuai dengan bab tiga pada tahapan implementasi, hal tersebut dilakukan dikarenakan pada tahapan implementasi merupakan metode penanaman algoritma yang diterapkan pada studi kasus kali ini.

#### 1.1 Analisa data

Analisa data dilakukan guna mendapati nilai yang nantinya dijadikan sebagai bahan pertimbangan apakah algoritma *sequential search* berjalan dengan baik ketika digunakan. Pengambilan data dilakukan pada halaman `index.php` menggunakan element pada google chrome yaitu inspect, kemudian pilih menu network sehingga kita dapat mengetahui berapa kecepatan dan ukuran kapasitas web.

Tabel 4. 1 daftar tabel pengujian

Nama	Uji 1	Uji 2	Uji 3	Uji 4	Uji 5	Rata - rata
Wadah serbet	50 ms	82 ms	64 ms	68 ms	68 ms	66,4 ms
Tatakan Lilin	53 ms	83 ms	50 ms	64 ms	47 ms	59,4 ms
Buku Agenda	81 ms	49 ms	44 ms	54 ms	71 ms	59,8 ms
Kantung Serut	45 ms	68 ms	63 ms	48 ms	48 ms	54,4 ms
Wadah cat	47 ms	60 ms	46 ms	53 ms	57 ms	52,6 ms
Total Rata - Rata Hasil Uji						
58,52 ms = 0,005852 S/detik						

Dengan menggunakan rumus menghitung nilai rata – rata:

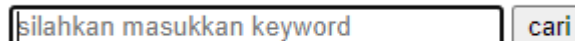
$$\text{Nilai rata – rata} = \frac{\text{jumlah nilai}}{\text{banyaknya data}}$$

Dengan hasil percobaan tersebut didapatkan nilai rata – rata 0,005852 S/detik merupakan angka yang sudah cukup untuk digunakan sebagai pencarian dengan algoritma *sequential search*.

## 1.2 Pembahasan

Pada sub bab pembahasan dimulai dari tahapan implementasi, pada kasus ini algoritma di-implementasikan pada bagian search box guna melakukan pencarian array, dengan penjelasan sebagai berikut :

pada kasus ini algoritma di-implementasikan pada bagian search box guna melakukan pencarian array, dengan penjelasan sebagai berikut :



silahkan masukkan keyword cari

Gambar 4.1 search box pada index.php

Dimulai dari halaman index.php

Buat menggunakan html untuk membuat tipe inputan sebagai search box

```
<form action="" method="post">
  <input type="text" name="keyword" size="30" autofocus
    placeholder="silahkan masukkan keyword"
    autocomplete="off">
  <button type="submit" name="search">cari</button>
</form>
```

Berkut merupakan penjelasan pada tag HTML :

1. Pada baris pertama tag `<form action="" method="post">` berfungsi untuk membuat form inputan tapi tag form input tidak akan berjalan setidaknya di bahasa PHP. Kemudian tag form memiliki dua buah atribut yang mana mempunyai fungsi masing masing, diantaranya action berfungsi untuk pengolahan data, dimana data tersebut dikirimkan, jika dikosongkan maka data akan dikirimkan dan diolah kehalaman yang sama. Kemudian tag method

digunakan untuk menampilkan data yang dikelola, jika menggunakan \$\_POST maka data tidak akan dikirimkan ke URL-nya, begitu juga sebaliknya jika memakai \$\_GET maka data akan ditampilkan di URL.

2. Pada baris selanjutnya terdapat tag

```
<input type="text" name="keyword" size="30"
autofocus placeholder="silahkan masukkan keyword"
autocomplete="off">
```

3. Yang berfungsi sebagai inputan dimana terdapat type dari inputan tersebut berupa text, Bernama keyword, dengan ukuran tigapuluh. Terdapat juga tag autofocus berguna ketika user masuk ke halaman tersebut langsung diarahkan pada search box-nya. Kemudian placeholder berfungsi memberikan panduan berupa text yang ada dalam search box dan tag autocomplete berfungsi sebagai penghilang Riwayat pencarian pada search box jika di-non aktifkan.
4. <button type="submit" name="search">cari</button> tag ini berfungsi memunculkan tombol bertipe submit dengan nama search yang didalamnya terdapat nama tombol cari.

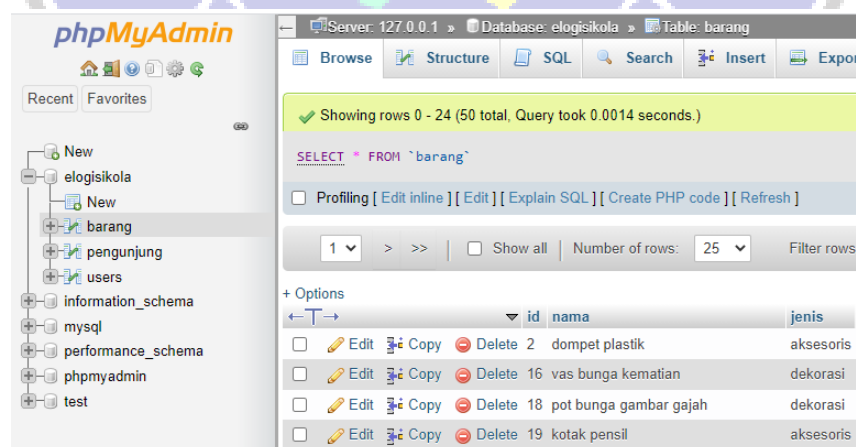
Hubungkan index.php dengan function.php yang mana dalam function.php tersebut berisikan berbagai fungsi yang digunakan pada websiste yang dibuat. Dengan baris kode serta penjelasan sebagai berikut:

```
<?php
require 'functions.php';
$barang = query("SELECT * FROM barang ORDER BY id DESC"
);

//jika tombol cari ditekan

if (isset($_POST["search"])) {
    $barang = search($_POST["keyword"]);
    // var_dump($barang);
}
?>
```

1. Buka tag php, dengan menggunakan bahasa php maka dapat menghubungkan antara file index dengan file function.
2. \$barang = query("SELECT \* FROM barang "); yang mana terdapat variable \$barang guna menampung seluruh data yang ada di database dengan table barang. Datanya diambil dari database elogis ikola. Dengan tampilan database sebagai berikut



The screenshot shows the phpMyAdmin interface. The left sidebar displays the database structure, including 'elogisikola' and its tables: 'barang', 'pengunjung', and 'users'. The main area shows the 'barang' table with the following data:

id	nama	jenis
2	dompet plastik	aksesoris
16	vas bunga kematian	dekorasi
18	pot bunga gambar gajah	dekorasi
19	kotak pensil	aksesoris

Gambar 4.2 tampilan database dengan table barang

3. Pada logika if yang mana jika tombol ditekan ambil data method `$_POST` pilih tombol yang Bernama search, kemudian terdapat variable `$barang` = yang berisikan function search dimana mengambil data dari `$_POST` (apapun yang diketikkan sebagai keyword).
4. Dengan pengujian menggunakan `var_dump` pada variable `$barang` agar mendapatkan nilai yang dicari. Dari tag input yang Bernama keyword yang mana dikelola pada halaman `functions.php` Hasil pengujian dari tag `var_dump` berisikan array yang diambil dari table barang yang ada pada database. Dengan detail database dengan array sebagai berikut:

```
array(1) { [0]=> array(6) { ["id"]=> string(2) "26" ["nama"]=>
string(16) "keranjang sampah" ["jenis"]=> string(9) "aksesoris"
["diskripsi"]=> string(101) "keranjang sampah gunakan sebagai
wadah sampahmu, lumayan sebagai penghemat biaya pengeluaran
bulanan" ["harga"]=> string(6) "100000" ["gambar"]=> string(17)
"60d4be693baaa.png" } }
```

*Gambar 4.3 tampilan nilai yang dicari pada table barang*

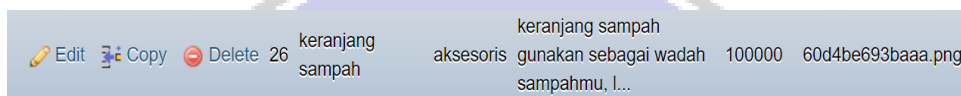
Dengan detail array sebagai berikut :




1. `array(1) {[0]}` merupakan sebuah array berisikan satu data kemudian pada nilai nol `{[0]}` nilai tersebut berupa array numerik yang index-nya berupa angka nol.
2. `array(6)` yang merupakan sebuah array dengan index didalamnya berjumlah enam yang pertama `{ ["id"]=> string(2) "26"` dengan key berupa `["id"]` dan value berupa string dengan urutan ke-26.
3. Kemudian terdapat `["nama"]=> string(16) "keranjang sampah"` yang bernilai string dengan panjang enambelas karakter.
4. `["jenis"]=> string(9) "aksesoris"` bernilai string dengan panjang Sembilan karakter.
5. `["diskripsi"]=> string(101) "keranjang sampah gunakan sebagai wadah sampahmu, lumayan sebagai`

penghemat biaya pengeluaran bulanan " bernilai string dengan panjang seratus satu karakter.

6. ["harga"]=> string(6) "100000" bernilai string dengan enam karakter.
7. Dan yang terakhir adalah ["gambar"]=> string(17) "60d4be693baaa.png".

Sehingga menghasilkan data pada table barang seperti gambar dibawah ini.



 Edit	 Copy	 Delete	26	keranjang sampah	keranjang sampah	aksesoris	gunakan sebagai wadah	100000	60d4be693baaa.png
--	--	--	----	------------------	------------------	-----------	-----------------------	--------	-------------------

*Gambar 4.4 tampilan nilai yang dicari pada table barang*

Kemudian agar nilai string pada gambar berubah menjadi string random, yaitu dengan melakukan generate nama menjadi nama yang baru, hal tersebut bertujuan agar ketika ada string atau nama yang sama diunggah, maka string atau nama tersebut tidak tertumpuk atau berubah seluruh data yang memiliki kesamaan pada string atau nama pada bagian gambar. Dengan code untuk melakukan generate serta penjelasan sebagai berikut :

```
//generate nama gambar baru
$namaFileBaru = uniqid();
$namaFileBaru .= '.';
$namaFileBaru .= $ekstensiGambar;
```

1. Pada baris pertama membuat variabel yang digunakan untuk menampung tag `uniqid()`; tag tersebut berfungsi untuk membangkitkan angka dan huruf secara random nantinya digunakan sebagai nama gambar yang baru.
2. Pada baris kedua lakukan perangkaian variable `$namaFileBaru` menggunakan tag `.=` (titik sama dengan) yang berarti tempelkan atau gabungkan dengan nilai `'.'`; (titik).

3. Kemudian pada baris ketiga gabungkan variable nama file baru dengan variable ekstensi gambar.

Kemudian buat fungsi pencarian pada halaman `function.php` dimana halaman tersebut merupakakan kumpulan fungsi – fungsi untuk menjalankan perintah pada website yang dibuat. Pembahasan hanya akan berfokus pada fungsi pencarian yang mana algoritma diimplementasikan pada fungsi tersebut.

```
//fungsi pencarian
function search ($keyword) {

    $query = "SELECT * FROM barang WHERE nama LIKE '%" .
$keyword%' ";
    return query($query);
}
```

Pada fungsi pencarian dibuat didalam file Bernama *function.php* yang dijelaskan sebagai berikut:

1. Fungsi tersebut mendapatkan data dari `$_POST["keyword"]` yang ada pada `index.php` nantinya data tersebut ditampung ke variable `$keyword`.
2. Fungsi tersebut berisikan variable `$query` yang menampung, data dari table barang dimana nama mengandung huruf – huruf pada keyword.
3. Kemudian kembalikan pada query yang mengandung variable `$query`.

Kemudian buat logika pada `index.php` yang mana nantinya akan menunjukkan hasil dari pencarian tersebut.:

```
<!-- fungsi sequential search jika data tidak ditemukan  
-->
```

```
<?php if(empty($barang)) :?>  
    <tr>  
        <td colspan="7">data yang anda cari tidak  
ditemukan</td>  
    </tr>  
<?php endif ;?>
```

Yang mana code diatas dijelaskan sebagai berikut :

1. Buat logika if yang berisikan jika variable yang dicari (\$barang) kosong atau tidak ditemukan maka tampilkan data yang dicari tidak ditemukan, kemudian akhiri dengan endif.

Jika disederhanakan dalam bentuk psudo code algoritma sequential search berbentuk seperti ini :

1. Cari data data barang dimana data tersebut memiliki kemiripan dengan data keyword yang ingin dicari.
2. Jika data yang dicari ketemu maka tampilkan data yang dicari tersebut.
3. Kalau tidak maka tampilkan data yang anda cari tidak ditemukan.
4. Dan akhiri dengan pencarian.