

LAPORAN PRAKTIKUM
PEMROGRAMAN BERBASIS WEB
“CRUD (Create Read Update Delete) pada PHP”



Disusun Oleh :

Nama : I Kadek Ari Surya

Nim : 1808561026

Kelas : B

PROGRAM STUDI INFORMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS UDAYANA
BADUNG
2020

KATA PENGANTAR

Puji dan syukur ke hadirat Tuhan Yang Maha Kuasa atas segala rahmat yang diberikan-Nya sehingga tugas Laporan Praktikum Pemrograman Berbasis Web yang berjudul ” CRUD (*Create Read Update Delete*) pada PHP” ini dapat saya selesaikan. Laporan ini saya buat sebagai kewajiban untuk memenuhi tugas. Dalam kesempatan ini, penulis menghaturkan terimakasih yang dalam kepada semua pihak yang telah membantu menyumbangkan ide dan pikiran mereka demi terwujudnya makalah ini. Akhirnya saran dan kritik pembaca yang dimaksud untuk mewujudkan kesempurnaan makalah ini penulis sangat hargai.

Badung, 8 Desember 2020

Penyusun

DAFTAR ISI

COVER	
KATA PENGANTAR.....	ii
DAFTAR ISI.....	iii
BAB I	1
1.1 Latar Belakang.....	1
1.2 Tujuan	1
1.3 Manfaat	1
BAB II.....	3
2.1 CRUD	3
2.2 Simbol Pembuka dan Penutup Skrip PHP	4
2.3 Variabel PHP	4
2.4 Operator Pada PHP	5
2.5 Struktur Kontrol	7
2.6 Fungsi	7
2.7 XAMPP	8
2.8 Visual Studio Code	8
BAB III	9
3.1 Implementasi CRUD pada PHP.....	9
3.1.1 Tampilan Sederhana.....	9
3.1.2 Database dan Penjelasan Kode Program	11
BAB IV.....	23
4.1. Simpulan.....	23
4.2. Saran.....	23
DAFTAR PUSTAKA.....	24

BAB I

PENDAHULUAN

1.1 Latar Belakang

PHP merupakan singkatan dari "Hypertext Preprocessor". PHP banyak digunakan open source script dengan bahasa scripting. PHP dieksekusi pada server. PHP gratis untuk didownload. File PHP dapat berupa teks, HTML, CSS, JavaScript, dan PHP. Kode kode PHP dijalankan di server, dan hasilnya dikembalikan ke browser sebagai file HTML. PHP yang asli memiliki ekstensi ".php". PHP dapat menghasilkan konten halaman dinamis. PHP dapat membuat, membuka, membaca, menulis, menghapus, dan menutup file pada server. PHP dapat mengumpulkan data formulir. PHP dapat mengirim dan menerima cookies. PHP dapat menambah, menghapus, memodifikasi data dalam database. PHP dapat digunakan untuk mengontrol akses pengguna. PHP dapat mengenkripsi data.

Website merupakan halaman situs sistem informasi yang digunakan untuk menampilkan informasi teks, gambar, animasi, suara atau gabungan dari semuanya baik yang bersifat statis maupun dinamis dimana masing-masing dihubungkan dengan jaringan-jaringan halaman (hyperlink).

Sehingga dalam membuat website dengan informasi yang dinamis maka dibutuhkan bahasa pemrograman untuk melakukan itu. Salah satunya bisa dengan menggunakan bahasa PHP. Untuk mulai menggunakan PHP, kita dapat menemukan web host dengan PHP dan dukungan MySQL, lalu instal web server pada PC, dan kemudian instal PHP dan MySQL. Jika server telah mengaktifkan dukungan untuk PHP, kita tidak perlu melakukan apa-apa. Oleh karena itu pada praktikum ini, lebih fokus dijelaskan tentang operasi CRUD (Create, Read, Update, Delete) dengan bahasa pemrograman PHP.

1.2 Tujuan

- 1.2.1 Mengimplementasikan bahasa pemrograman PHP pada pembuatan CRUD.

1.3 Manfaat

1.3.1 Mengetahui dan memahami penggunaan perintah PHP pada pembuatan CRUD.

BAB II

KAJIAN PUSTAKA

2.1 CRUD

CRUD adalah singkatan dari create, read, update, delete nah create, read, update, delete sendiri merupakan sebuah fungsi yang di gunakan oleh banyak web dan banyak aplikasi pengelola data. Seperti aplikasi web perpustakaan, web sekolah, web pendaftaran sekolah dan masih banyak lainnya. Fungsi ini sesuai namanya sebagai berikut.

- Create, sesuai dengan artinya dalam bahasa indonesia create adalah membuat maka bisa di artikan create adalah sebuah fungsi untuk membuat atau menambah data contohnya jika kalian pernah membuka sosial media layaknya instagram, facebook, twitter dan tetangga tetangganya. Maka kalian pasti akan di minta untuk mendaftar atau membuat akun terlebih dahulu.
- Read, jika create adalah membuat maka read dalam crud ini adalah sebuah fungsi yang di gunakan untuk menampilkan data, supaya bisa di lihat dan di baca oleh para pengunjung. contohnya jika kalian pernah berkunjung ke sebuah situs perpustakaan, toko online dan lain sebagainya maka di sana pasti kalian akan melihat sebuah gambar dan tulisan nah di situlah fungsi read di gunakan. Jadi fungsi read ini juga tidak hanya sebuah fungsi untuk menampilkan tulisan ya bisa juga di gunakan untuk menampilkan sebuah gambar, video maupun lagu-lagu.
- Update, untuk update sendiri adalah sebuah fungsi untuk meng edit atau meng ubah sebuah isi data yang telah di buat oleh fungsi create. fungsi ini juga sangat penting untuk di gunakan oleh para developer web karena jika tidak ada fungsi ini maka jika ada kesalahan kita harus meng hapusnya terlebih dahulu dan membuatnya lagi.
- Delete, dan yang terakhir adalah delete jadi fungsi delete dalam crud adalah untuk menghapus sebuah isi data. Tak kalah

pentingnya dengan update delete juga sangatlah penting karena jika tidak ada sebuah fungsi untuk delete maka admin akan kesusahan.

2.2 Simbol Pembuka dan Penutup Skrip PHP

Skrip PHP selalu diawali dengan tanda ‘<’. Skrip PHP dapat diletakkan dimana saja dalam suatu dokumen HTML. Beberapa server yang sudah diatur konfigurasi directive ‘shorthand-support’, dapat mengawali skrip dengan tanda ‘<’. Tetapi demi kompatibilitas maksimum, disarankan menggunakan bentuk standar ‘<?php’.

2.3 Variabel PHP

Suatu variabel digunakan untuk menyimpan suatu nilai, dapat berupa teks, angka, atau array. Variabel dalam PHP menggunakan simbol ‘\$’ di awal namanya. Sintaks perintah membuat variabel: \$nama_var = nilai;

Tipe data variabel tidak perlu dideklarasikan, PHP akan otomatis mengkonversi atau menentukan tipe data variabel berdasarkan nilai yang disimpannya. Contoh:

```
<?php
$nama='Alvina Khansa';
$nilai=90;
?>
```

Variabel nama diatas otomatis akan bertipe string, variabel nilai akan bertipe integer.

Scope atau ruang lingkup variabel adalah bagian dari skrip yang dapat mereferensikan variabel tersebut. Ada 3 scope variabel dalam PHP:

- Scope Local : Suatu variabel yang dibuat pada suatu fungsi akan menjadi variabel lokal (memiliki scope local) dan hanya bisa diakses di dalam fungsi. Nama variabel yang sama dapat dibuat dalam fungsi yang berbeda, sebab variabel lokal hanya dikenali oleh fungsi yang membentuk variabel tersebut. Variabel lokal akan dihapus setelah fungsi usai dieksekusi.
- Scope Global: Scope global dimiliki oleh variabel yang dibuat diluar fungsi. Variabel dengan scope global dapat

diakses dari bagian manapun dari program selama perintah tersebut ditulis diluar suatu fungsi. Variabel global dapat diakses dari dalam suatu fungsi dengan menggunakan kata kunci 'global'.

- Scope Statik : Ketika suatu fungsi selesai digunakan, secara normal semua variabelnya akan dihapus. Jika diinginkan variabel-variabel tersebut tidak dihapus ketika fungsi selesai dipakai, gunakan kata kunci 'static' saat membuat variabel.

2.4 Operator Pada PHP

A. Operator Hitung

Operator	Penjelasan	Contoh	Hasil
+	Pertambahan	$x=2$ $y=x+2$	$y=4$
-	Pengurangan	$x=2$ $y=5-x$	$y=3$
*	Perkalian	$x=4$ $y=x*5$	$y=20$
/	Pembagian	$y=15/5$	$y=3$
%	Sisa hasil bagi	$x=10\%5$ $y=10\%8$ $z=5\%2$	$x=0$ $y=2$ $z=1$
++	Inkremen	$x=5$ $x++$	$x=6$
--	dekremen	$x=5$ $x--$	$x=4$

B. Operator Penugasan

Operator	Contoh	Sama dengan
=	x=y	x=y
+=	x+=y	x=x+y
-=	x-=y	x=x-y
=	x=y	x=x*y
/=	x/=y	x=x/y
.=	x.=y	x=x.y
%=	x%=y	x=x%y

C. Operator Perbandingan

Operator	Penjelasan	Contoh
==	sama dengan	5==8 mengembalikan nilai <i>false</i>
!=	tidak sama dengan	5!=8 mengembalikan nilai <i>true</i>
<>	tidak sama dengan	5<>8 mengembalikan nilai <i>true</i>
>	lebih besar dari	5>8 mengembalikan nilai <i>false</i>
<	lebih kecil dari	5<8 mengembalikan nilai <i>true</i>
>=	lebih besar dari atau sama dengan	5>=8 mengembalikan nilai <i>false</i>
<=	lebih kecil dari atau sama dengan	5<=8 mengembalikan nilai <i>true</i>

D. Operator Logika

Operator	Penjelasan	Contoh
&&	and	x=6; y=3; (x < 10 && y > 1) mengembalikan <i>true</i>
	or	x=6; y=3; (x==5 y==5) mengembalikan <i>false</i>
!	not	x=6; y=3; !(x==y) mengembalikan <i>true</i>

2.5 Struktur Kontrol

A. If Else

Digunakan untuk mengetes apakah suatu kondisi itu benar (*true*) atau salah (*false*).

B. Perulangan for dan foreach

Digunakan untuk melakukan perulangan jika jumlah perulangan sudah diketahui sebelumnya.

C. Perulangan while

Digunakan ketika ingin melakukan perulangan selama kondisi yang ditentukan adalah *true*.

2.6 Fungsi

Fungsi merupakan salah satu teknik pemrograman modular. Sebuah aplikasi besar disusun dari modul-modul yang berupa sebuah fungsi atau prosedur. Fungsi berisi sekelompok kode dengan tugas dan tujuan spesifik. Fungsi tidak akan dieksekusi ketika program dijalankan. Fungsi hanya akan dieksekusi jika dilakukan pemanggilan terhadapnya. Pemanggilan dapat dilakukan dari mana saja dalam program. Keuntungan teknik ini, modul-modul yang dibuat dapat digunakan berkali-kali (*reuse*).

Sebuah fungsi dibuat dengan aturan sintaks:

```
function namaFungsi() {
kode-kode yang akan dieksekusi;
}
```

Beberapa petunjuk dalam membuat sebuah fungsi:

- Namai fungsi yang menggambarkan fungsinya

- Nama fungsi dimulai dengan huruf atau garis bawah (underscore), tidak boleh angka

2.7 XAMPP

XAMPP adalah suatu bundel web server yang populer digunakan untuk di Windows karena kemudahan instalasinya. Bundel program open source tersebut berisi antara lain server web Apache, interpreter PHP, dan basis data MySQL.

2.8 Visual Studio Code

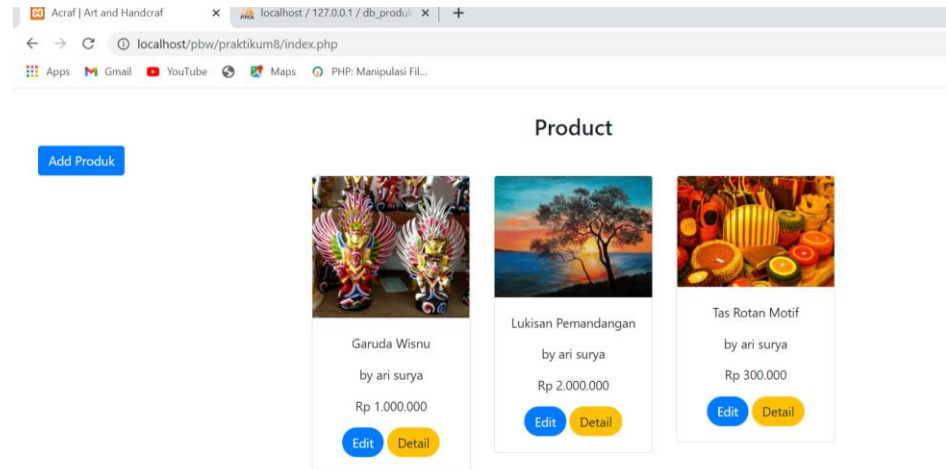
Visual Studio Code adalah editor source code yang dikembangkan oleh Microsoft untuk Windows, Linux dan MacOS. Ini termasuk dukungan untuk debugging, GIT Control yang disematkan, penyorotan sintaks, penyelesaian kode cerdas, cuplikan, dan kode refactoring. Hal ini juga dapat disesuaikan, sehingga pengguna dapat mengubah tema editor, shortcut keyboard, dan preferensi. Visual Studio Code gratis dan open-source, meskipun unduhan resmi berada di bawah lisensi proprietary. Kode Visual Studio didasarkan pada Elektron, kerangka kerja yang digunakan untuk menyebarkan aplikasi Node.js untuk desktop yang berjalan pada Blink layout. Meskipun menggunakan kerangka Elektron, Visual Studio Code tidak menggunakan Atom dan menggunakan komponen editor yang sama (diberi kode nama "Monaco") yang digunakan dalam Visual Studio Team Services yang sebelumnya disebut Visual Studio Online.

BAB III PEMBAHASAN

3.1 Implementasi CRUD pada PHP

3.1.1 Tampilan Sederhana

Halaman Utama



Halaman Detail Produk

Detail Produk



Nama Produk

Garuda Wisnu

Pemilik Produk

ari surya

Harga Produk

1000000

Stok Produk

12

Lokasi

Jl.Soetami

Status

Tersedia

Kategori

statue

Deskripsi Produk

patung dibuat dari kayu cendana yang sudah kering

[Kembali](#)

Halaman Edit Produk

Edit Produk

Nama Produk

Lukisan Pemandangan

Pemilik Produk

ari surya

Harga Produk

2000000

Stok Produk

12

Lokasi

Jl.Soetami

Status

Tersedia

Kategori

paint

Deskripsi Produk

Bahan lukisan dari kanvas berkualitas dan cat minyak

Foto :



[Choose File](#)

No file chosen

[Update](#)

[Delete](#)

Halaman Tambah Barang

Tambah Barang

Nama Produk

Pemilik Produk

Harga Produk

Stok Produk

Lokasi

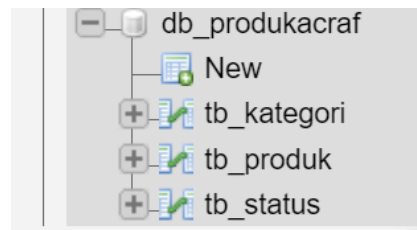
Status

Kategori

Deskripsi Produk

Pilih Foto No file chosen

3.1.2 Database dan Penjelasan Kode Program



➤ functions.php

```

functions.php
1  <?php
2  //KONEKSI DATABASE
3  $conn=mysqli_connect('localhost','root','','db_produkacraf');
4
5  function query($query){
6      global $conn;
7      $result = mysqli_query($conn, $query);
8      $rows=[];
9      while( $row = mysqli_fetch_assoc($result)){
10         $rows[] = $row;
11     }
12     return $rows;
13 }
14

```

- Terlebih dahulu dibuat variabel conn untuk melakukan koneksi ke dalam database yang telah dibuat.
- Setelah itu terdapat fungsi query untuk membuat penulisan query pada php yang lebih singkat, sehingga setiap kali melakukan query tidak perlu untuk menuliskan mysqli_query dan lain-lainnya.

```

15
16 function add($data){
17     global $conn;
18     //ambil data dari setiap form
19     $nama_produk= $data["nama_produk"];
20     $owner_produk= $data["owner_produk"];
21     $harga_produk= $data["harga_produk"];
22     $stok_produk= $data["stok_produk"];
23     $lokasi= $data["lokasi"];
24     $status = $data["status"];
25     $deskripsi_produk= $data["deskripsi_produk"];
26     $kategori= $data["kategori"];
27
28     //upload foto
29     $foto_produk=upload();
30     if( !$foto_produk){
31         return false;
32     }
33
34     //query insert data
35     $query= "INSERT INTO tb_produk VALUES('','$nama_produk', $harga_produk, $stok_produk, '$lokasi', $status,
36         '$deskripsi_produk', $kategori,
37         '$foto_produk', '$owner_produk')";
38     mysqli_query($conn, $query);
39     return mysqli_affected_rows($conn);
40 }
41

```

- Fungsi add digunakan untuk menambahkan data ke dalam tb_produk. Fungsi ini menerima parameter dengan nama \$data. Selanjutnya semua data yang dibawa disalin kedalam variabel lokal, lalu kemudian dimasukkan ke dalam tb_produk dengan menggunakan perintah INSERT.

```

41 function update($data){
42     global $conn;
43     //ambil data dari setiap form
44     $id_produk= $data["id_produk"];
45     $nama_produk= $data["nama_produk"];
46     $owner_produk= $data["owner_produk"];
47     $harga_produk= $data["harga_produk"];
48     $stok_produk= $data["stok_produk"];
49     $lokasi= $data["lokasi"];
50     $status= $data["status"];
51     $deskripsi_produk= $data["deskripsi_produk"];
52     $kategori= $data["kategori"];
53     $fotolama=$data["fotolama"];
54
55     //cek user apakah user milih foto baru atau tidak
56     if( $_FILES["foto_produk"]["error"]=== 4){
57         $foto=$fotolama;
58     } else{
59         $foto=upload();
60     }
61     //query insert data
62     $query= "UPDATE tb_produk SET
63         nama_produk= '$nama_produk',
64         harga_produk= '$harga_produk',
65         stok_produk= '$stok_produk',
66         lokasi= '$lokasi',
67         id_status= '$status',
68         deskripsi_produk= '$deskripsi_produk',
69         id_kategori= '$kategori',
70         foto_produk= '$foto',
71         owner_produk= '$owner_produk'
72         WHERE id_produk= $id_produk
73     ";
74     var_dump($id_produk);
75     mysqli_query($conn, $query);
76     return mysqli_affected_rows($conn);
77 }

```

- Fungsi update digunakan untuk memperbarui data ke dalam tb_produk. Fungsi ini menerima parameter dengan nama \$data. Selanjutnya semua data yang dibawa disalin kedalam variabel lokal, lalu dicek apakah ada perubahan file foto atau tidak, kemudian dilakukan pembaruan data ke dalam tb_produk dengan menggunakan perintah UPADATE.

```

78
79 function delete($id){
80     global $conn;
81     mysqli_query($conn,"DELETE FROM tb_produk WHERE id_produk=$id");
82     return mysqli_affected_rows($conn);
83 }
84
85

```

- Fungsi delete digunakan untuk menghapus data yang terdapat pada tb_produk sesuai dengan id_produk yang ingin dihapus.


```

86
87 function upload(){
88     $namaFile=$_FILES['foto_produk']['name'];
89     $ukuranFile=$_FILES['foto_produk']['size'];
90     $error=$_FILES['foto_produk']['error'];
91     $tmpName=$_FILES['foto_produk']['tmp_name'];
92
93     //cek apakah tidak ada foto diupload
94     //error 4 artinya tidak ada foto yang diupload;
95     if( $error!=0 ){
96         echo "
97         <script>
98             alert('Pilih Gambar Terlebih Dahulu');
99         </script>
100     ";
101     return false;
102     }
103
104     //cek apakah yang diupload itu foto
105     $ekstensiGambarValid=['jpg', 'jpeg', 'png'];
106     $ekstensiGambar=explode('.', $namaFile);
107     $ekstensiGambar= strtolower(end($ekstensiGambar));
108     if( ! in_array($ekstensiGambar, $ekstensiGambarValid ) ) {
109         echo "
110         <script>
111             alert('Yang Anda Upload Bukan Gambar');
112         </script>
113     ";
114     return false;
115     }
116
117     //cek ukuran dalam byte
118     if( $ukuranFile > 1000000 ){
119         echo "
120         <script>
121             alert('Ukuran Gambar Lebih Dari 1 Mb');
122         </script>
123     ";
124     return false;
125     }
126     //lolos mengecek
127     //generate nama foto baru
128
129     $namaFileBaru=uniqid();
130     $namaFileBaru .= '.';
131     $namaFileBaru .= $ekstensiGambar;
132     //memindahkan data
133     move_uploaded_file($tmpName, 'img/'.$namaFileBaru );
134     return $namaFileBaru;
135 }
136
137 ?>

```

- Fungsi upload digunakan untuk mengecek kesesuaian file gambar yang dipilih oleh pengguna mulai dari format hingga ukuran file. batas file maksimal yang set adalah 1 mb.

➤ index.php

```

index.php
1  <?php
2
3  require 'functions.php';
4  $produk=query("SELECT * FROM tb_produk");
5
6  ?>
7
8  <!DOCTYPE html>
9  <html>
10 <head>
11     <title>Acraf | Art and Handcraf</title>
12     <!-- meta tag -->
13     <meta charset="utf-8">
14     <!-- <meta http-equiv="refresh" content="3600"> -->
15     <meta name="keywords" content="Art and Handcraf">
16     <meta name="description" content="Platform penjualan barang seni dan kerajinan tangan">
17     <meta name="author" content="Tim Acraf">
18     <!-- link css -->
19     <link rel="stylesheet" href="css/bootstrap.min.css">
20     <link rel="stylesheet" href="css/style.css">
21     <!-- fontawesome -->
22     <script src="https://kit.fontawesome.com/fe118aecdc.js" crossorigin="anonymous"></script>
23 </head>
24

```

- Pada line 3, halaman ini dihubungkan dengan file functions.php untuk dapat menggunakan fungsi-fungsi yang telah dibuat didalamnya. Halaman index.php berfungsi untuk menampilkan semua produk yang ada dalam database. Sehingga pada line 4, dilakukan deklarasi variabel produk yang nilainya diperoleh dari pemanggilan fungsi query. Seleksi dilakukan terhadap semua kolom yang terdapat pada tabel tb_produk.
- Pada bagian head html, seperti biasa berisikan tag meta sebagai identitas dari website dan link untuk menghubungkan ke dalam file css dan fontawesome.

```

25 <body data-spy="scroll" data-target="#navbarResponsive">
26 <!-- product -->
27 <div id="product">
28   <h3 class="heading text-center">Product</h3>
29   <div class="heading-underline"></div>
30
31   <!-- tambah produk -->
32   <div class="row justify-content-first">
33     <div class="col-md-2">
34       <form action="" method="">
35         <a href="add.php" class="btn btn-primary">Add Produk </a>
36       </form>
37     </div>
38   </div>
39   <!-- akhir tambah produk -->
40   <!-- row produk -->
41   <div class="row justify-content-center">
42     <!-- produk -->
43     <?php foreach ($produk as $data): ?>
44       <div class="col-md-2">
45         <div class="card text-center">
46           " alt="">
47           <div class="card-body">
48             <p class="product-title color-palette1"><?php echo $data["nama_produk"] ?></p>
49             <p class="product-owner">by <?php echo $data["owner_produk"] ?></p>
50             <p class="product-price"><?php echo $harga = "Rp " . number_format($data["harga_produk"],0,',','.'); ?></p>
51             <a href="edit.php?id=<?php echo $data["id_produk"]?>" class="btn btn-primary rounded-pill">Edit</a>
52             <a href="detail.php?id=<?php echo $data["id_produk"]?>" class="btn btn-warning rounded-pill">Detail</a>
53           </div>
54         </div>
55       </div>
56     <?php endforeach; ?>
57     <!-- akhir produk -->
58   </div>
59   <!-- row produk -->
60 </div>
61 <!-- end product -->
62 </body>
63

```

- Pada bagian body terdapat 2 bagian program yang utama. Pertama adalah program untuk menambahkan data produk. Program ini akan berjalan ketika pengguna menekan tombol Add Product. Pengguna akan diarahkan pada halaman add.php. Semua proses penambahan akan dijelaskan pada penjelasan berikutnya di halaman add.php
- Program utama yang kedua adalah untuk membaca semua data yang terdapat pada tb_produk untuk ditampilkan pada halaman utama website. Penampilan data produk dilakukan dengan membuka tag php dalam file html dengan menggunakan perintah foreach. Setiap kali perulangan dilakukan, pembacaan variabel array data disesuaikan dengan nama kolom pada database untuk dapat menampilkan produk sesuai dengan urutan.

```

65 <!-- javascript -->
66 <script src="https://code.jquery.com/jquery-3.2.1.slim.min.js"
  integrity="sha384-KJ3o2DKtIkvYIK3UENzmM7KcRr/rE9/Qp6aAZG3wFDMVNA/GpGFF93hXpG5KkN"
  crossorigin="anonymous"></script>
67 <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.12.9/umd/popper.min.js"
  integrity="sha384-ApNbgh9B+Y1QKtv3Rn7W3mgPxhU9K/ScQsAP7hUibX39j7fakFPskvXusvfa0b4Q"
  crossorigin="anonymous"></script>
68 <script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/js/bootstrap.min.js"
  integrity="sha384-JZR6Spejh4U02d8j0t6vLEHfe/JQGiRRSQQxSfFWpi1MquVdAyjUar5+76PvcMv1"
  crossorigin="anonymous"></script>
69 </html>

```

- Pada bagian akhir html, dideklarasikan juga script yang dibutuhkan untuk menunjang kinerja program pada halaman

utama. Tag javascript ini merupakan file untuk menjalankan framework bootstrap css yang digunakan.

➤ detail.php

```
detail.php
1  <?php
2
3  require 'functions.php';
4  $id_produk= $_GET["id"];
5  $produk= query("SELECT * FROM tb_produk INNER JOIN tb_status USING(id_status) INNER JOIN tb_kategori
USING(id_kategori) WHERE id_produk=$id_produk") [0];
6
7  ?>
8
9  <!DOCTYPE html>
10 <html>
11 <head>
12     <title>Acraf | Detail Product </title>
13     <!-- meta tag -->
14     <meta charset="utf-8">
15     <!-- <meta http-equiv="refresh" content="3600"> -->
16     <meta name="keywords" content="Art and Handcraf">
17     <meta name="description" content="Platform penjualan barang seni dan kerajinan tangan">
18     <meta name="author" content="Tim Acraf">
19     <!-- link css -->
20     <link rel="stylesheet" href="css/bootstrap.min.css">
21     <link rel="stylesheet" href="css/style.css">
22     <!-- fontawesome -->
23     <script src="https://kit.fontawesome.com/fe118aecdc.js" crossorigin="anonymous"></script>
24 </head>
```

- File detail.php berfungsi untuk menampilkan detail informasi produk yang lebih lengkap sesuai dengan semua data yang terdapat pada tabel tb_produk. Karena tb_produk berelasi dengan beberapa tabel, sehingga dibutuhkan join ke beberapa tabel untuk memperoleh informasi yang lebih lengkap.
- Ketika pengguna di halaman utama menekan tombol detail, secara tidak langsung ada id yang dibawa oleh program. Untuk mengetahui id tersebut, digunakan fungsi \$_GET. Kemudian id tersebut ditangkap pada url yang dituju. Pada line 3, dihubungkan kembali dengan file functions.php. Setelah itu id yang terdapat pada url disimpan dalam variabel \$id_produk.
- Setelah id produk ditangkap, selanjutnya akan dilakukan query pada tb_produk. Seleksi dilakukan terhadap semua komponen yang terdapat dalam tb_produk. Kemudian dilanjutkan dengan melakukan INNER JOIN ke tb_status menggunakan foreign key id_status, kemudian INNER JOIN lagi ke tb_kategori menggunakan foreign key id_kategori dengan persyaratan produk yang diselect adalah sesuai dengan id_produk yang telah ditangkap pada url tadi. Diakhir

query diisi [0] untuk membatasi hanya satu indeks data yang kita ambil dari hasil yang didapatkan.

```
detail.php
25 <body>
26   <div id="detail">
27     <div class="row justify-content-center">
28       <h2>Detail Produk</h2>
29     </div>
30     <div class="row justify-content-center">
31       <div class="col-md-5 card p-4">
32         " width="300" style="display: block; margin:
33           auto;">
34         <form action="">
35           <div class="form-group">
36             <label for="nama_produk">Nama Produk</label>
37             <input type="text" name="nama_produk" id="nama_produk" class="form-control
38               rounded-pill" required value="<?=$produk["nama_produk"]>" disabled>
39           </div>
```

.....

```
69   <div class="form-group">
70     <label for="deskripsi_produk">Deskripsi Produk</label>
71     <textarea type="text" name="deskripsi_produk"
72       id="deskripsi_produk" class="form-control rounded-pill" required disabled><?=$
73       produk["deskripsi_produk"]></textarea>
74   </div>
75   <a href="index.php?name=update" class="btn btn-primary rounded-pill"> Kembali </a>
76 </form>
77 </div>
78 </div>
79 </div>
80 </body>
```

- Setelah semua data dengan id_produk yang dituju disimpan pada variabel \$produk, langkah selanjutnya adalah mencetaknya. Untuk mencetak data tersebut, terlebih dahulu digunakan tag form agar terlihat lebih rapi. Pada masing-masing tag input, nilai value yang diberikan disesuaikan dengan nama kolom yang terdapat pada hasil query yang didapatkan. Setelah itu, agar user tidak merubah-rubah lagi data yang terdapat dalam tag input, maka ditambahkan sintak disabled.
- Setelah semua data ditampilkan, pada bagian akhir form terdapat tombol Kembali. Ketika user menekan tombol ini, maka akan diarahkan ke halaman utama index.php

➤ add.php

```

add.php
1  <?php
2
3  require 'functions.php';
4
5  if( isset($_POST["add"])){
6
7      if( add($_POST)> 0) {
8          echo "
9              <script>
10                 alert('Data Berhasil Ditambahkan');
11                 document.location.href='index.php';
12             </script>
13             ";
14      } else{
15          echo "
16              <script>
17                 alert('Data Gagal Ditambahkan');
18                 document.location.href='index.php';
19             </script>
20             ";
21      }
22  }
23

```

- Pada add.php digunakan untuk menambahkan data yang ingin dimasukan ke dalam tb_produk. Pada line 3, halaman ini dihubungkan dengan functions.php. Pada line 5 dilakukan pengecekan apakah tombol add sudah ditekan apa belum. Ketika sudah terpenuhi, selanjutnya dilakukan pengecekan terhadap pemanggilan fungsi add. Jika nilainya lebih dari 0 maka berarti data berhasil ditambahkan, namun jika tidak berarti data gagal ditambahkan.

```

add.php
41 </head>
42 <body>
43     <div id="add">
44         <div class="row justify-content-center">
45             <h2>Tambah Barang</h2>
46         </div>
47         <div class="row justify-content-center">
48             <div class="col-md-6">
49                 <form action="" method="post" enctype="multipart/form-data">
50                     <div class="form-group">
51                         <label for="nama_produk">Nama Produk</label>
52                         <input type="text" name="nama_produk" id="nama_produk" class="form-control rounded-pill" required placeholder="masukan nama produk">
53                     </div>
54
55                     <div class="form-group">
56                         <label for="owner_produk">Pemilik Produk</label>
57                         <input type="text" name="owner_produk" id="owner_produk" class="form-control rounded-pill" required placeholder="masukan nama anda">
58                     </div>
59                 </form>
60             </div>
61         </div>
62     </div>
63

```

.....
.....

```

<div class="form-group">
  <label for="status">Status</label>
  <select class="form-control rounded-pill" name="status" id="status">
    <option value="">- pilih status</option>
    <option value="1">Tersedia</option>
    <option value="2">Tidak Tersedia</option>
  </select>
</div>

<div class="form-group">
  <label for="kategori">Kategori</label>
  <select class="form-control rounded-pill" name="kategori" id="kategori">
    <option value="">- pilih kategori</option>
    <option value="1">Souvenir</option>
    <option value="2">Paint</option>
    <option value="3">Furniture</option>
    <option value="4">Statue</option>
    <option value="5">Handcraf</option>
  </select>
</div>

<div class="form-group">
  <label for="deskripsi_produk">Deskripsi Produk</label>
  <textarea type="text" name="deskripsi_produk"
    id="deskripsi_produk" class="form-control rounded-pill" required
    placeholder="masukan dekrripsi produk"></textarea>
</div>

<div class="form-group">
  <label for="foto_produk">Pilih Foto</label>
  <input type="file" name="foto_produk" id="foto_produk" accept="image/*" >
</div>

<button type="submit" name="add" class="btn btn-primary rounded-pill">Add Produk</button>
</form>

```

- Pada bagian body, terdapat tag form untuk menginputkan data yang akan ditambahkan ke dalam database. Teknisnya hampir sama seperti proses input biasanya. Hanya saja di setiap tag input yang digunakan harus disertakan dengan name. Selain itu terdapat juga beberapa tipe input berupa section yakni data pilihan status dan kategori. Value dari setiap optionnya disesuaikan dengan id pada tabel tb_status dan tb_kategori. Setelah semua data diinputkan maka user diminta untuk menekan tombol Add produk.

➤ edit.php

```

edit.php
1  <?php
2  require 'functions.php';
3  $id_produk= $_GET["id"];
4  $produk= query("SELECT * FROM tb_produk INNER JOIN tb_status USING(id_status) INNER JOIN tb_kategori
    USING(id_kategori) WHERE id_produk=$id_produk") [0];
5
6  if( isset($_POST["update"])){
7
8      if( update($_POST)> 0) {
9          echo "
10             <script>
11                 alert('Data Berhasil Diupdate');
12                 document.location.href='index.php';
13             </script>
14             ";
15         } else{
16             echo "
17                 <script>
18                     alert('Data Gagal Diupdate');
19                     document.location.href='index.php';
20                 </script>
21                 ";
22             }
23         }
24     }

```

- Sebelum melakukan pemanggilan fungsi untuk melakukan update ataupun delete data, terlebih dahulu dilakukan query terhadap id_produk yang ditangkap melalui url. Proses select datanya hampir sama dengan proses pada detail.php. Setelah data yang dicari sudah ditemukan, dilakukan pengecekan sudah atau belumnya tombol update di tekan. Ketika sudah terpenuhi, selanjutnya dilakukan pengecekan terhadap pemanggilan fungsi update. Jika nilainya lebih dari 0 maka berarti data berhasil diupdate, namun jika tidak berarti data gagal diupdate.

```

24
25 if( isset($_POST["delete"])){
26
27     if( delete($id_produk)> 0) {
28         echo "
29         <script>
30             alert('Data Berhasil Dihapus');
31             document.location.href='index.php';
32         </script>
33         ";
34     } else{
35         echo "
36         <script>
37             alert('Data Gagal Dihapus');
38             document.location.href='index.php';
39         </script>
40         ";
41     }
42 }
43
44 ?>

```

- Setelah melakukan update user juga bisa memilih opsi untuk menghapus data. Dilakukan pengecekan sudah atau belumnya tombol delete di tekan. Ketika sudah terpenuhi, selanjutnya dilakukan pengecekan terhadap pemanggilan fungsi delete. Jika nilainya lebih dari 0 maka berarti data berhasil di hapus, namun jika tidak berarti data gagal dihapus. Secara otomatis user akan diarahkan ke halaman index.php

```

edit.php
62 <body>
63 <div id="edit">
64 <div class="row justify-content-center">
65 <h2>Edit Produk</h2>
66 </div>
67 <div class="row justify-content-center">
68 <div class="col-md-6">
69 <form action="" method="post" enctype="multipart/form-data">
70 <input type="hidden" name="id_produk" id="id_produk" required value="<?> $produk["id_produk"];
71 >" >
72 <input type="hidden" name="fotoLama" value="<?> $produk["foto_produk"]; ?>" >
73 <div class="form-group">
74 <label for="nama_produk">Nama Produk</label>
75 <input type="text" name="nama_produk" id="nama_produk" class="form-control rounded-pill"
76 required value="<?> $produk["nama_produk"]; ?>" >
77 </div>

```

.....
.....


```
123
124
125     <div class="form-group">
126       <label for="foto_produk">Foto :</label><br>
127       
128       <input type="file" name="foto_produk" id="foto_produk" accept="image/*" />
129     </div>
130
131     <button type="submit" name="update" class="btn btn-primary rounded-pill">Update</button>
132     <button type="submit" name="delete" class="btn btn-danger rounded-pill">Delete</button>
133
134   </form>
135 </div>
136 </div>
137 </div>
138 </body>
```

BAB IV PENUTUP

4.1. Simpulan

4.1.1 Pengaplikasian CRUD dalam PHP membutuhkan beberapa pemahaman. Pertama kita harus membuat database terlebih dahulu di PHP Myadmin untuk bisa dihubungkan secara local di website yang akan dibuat. Setelah itu barulah kita bisa membuat logika program menggunakan bahasa PHP. Dalam implementasinya, juga dikombinasikan dengan bahasa SQL terutama dalam melakukan query ke dalam database. Setelah itu data yang didapatkan dapat dicetak dengan bantuan bahasa HTML.

4.2. Saran

4.2.1 Dalam membuat CRUD dalam PHP dibutuhkan pemahaman yang lebih mendalam tentang alur kerja sistem yang akan dibangun sehingga dibutuhkan pemahaman yang lebih terutama relasi tabel, sintak SQL dan bahasan pemrograman web lainnya yang mendukung.

DAFTAR PUSTAKA

Sutarman, S.Kom [2003]. Membangun Aplikasi Web dengan PHP dan MySQL,
Graha Ilmu, Yogyakarta.