

***Dokumentasi Program Aplikasi Web
Tes Programmer di Fast Print***

A. PENDAHULUAN	3
1. Objektif	3
2. Latar belakang (background)	3
3. Hak akses sistem	3
B. Design Sistem.....	3
1. Entitas Relational Diagram (ERD).....	3
2. Class Diagram	4
3. Diagram aktivitas.....	5
C. Database.....	6
D. Program aplikasi	7
1. File koneksi.....	7
2. Membuat CRUD database produk.....	7
2.1 Create data produk ke dalam database	7
2.1 Update data produk	9
3. Membuat CRUD database kategori	12
3.1 Create data kategori	12
3.2 Update kategori	13
3.4 Tampilkan seluruh data kategori dari database	16
4. CRUD data status	16
4.1 Create database status	16
4.2 Update data status	17
5. Tampilan layout	19
5.1 Tambah data.....	19
5.2 Update data	20
5.3 Tampilan data.....	21

A. PENDAHULUAN

1. Objektif

Mengembangkan aplikasi berbasis web untuk membantu proses pemasaran dan proses penjualan di jejaring internet agar dapat memperluas jaringan perdagangan serta memudahkan proses arsip data yang dapat dikelola.

2. Latar belakang (background)

Saat ini sudah memasuki era digital, dimana perekonomian dapat berkembang dengan adanya digital. Mulai dari segi pemasaran serta arsip data yang lebih muda dengan memanfaatkan teknologi internet/digital. Aplikasi web ini diharapkan dapat membantu pemasaran dan juga arsip data agar dapat diakses dimana saja dan kapan saja sesuai dengan hak akses yang diberikan oleh developer kepada user

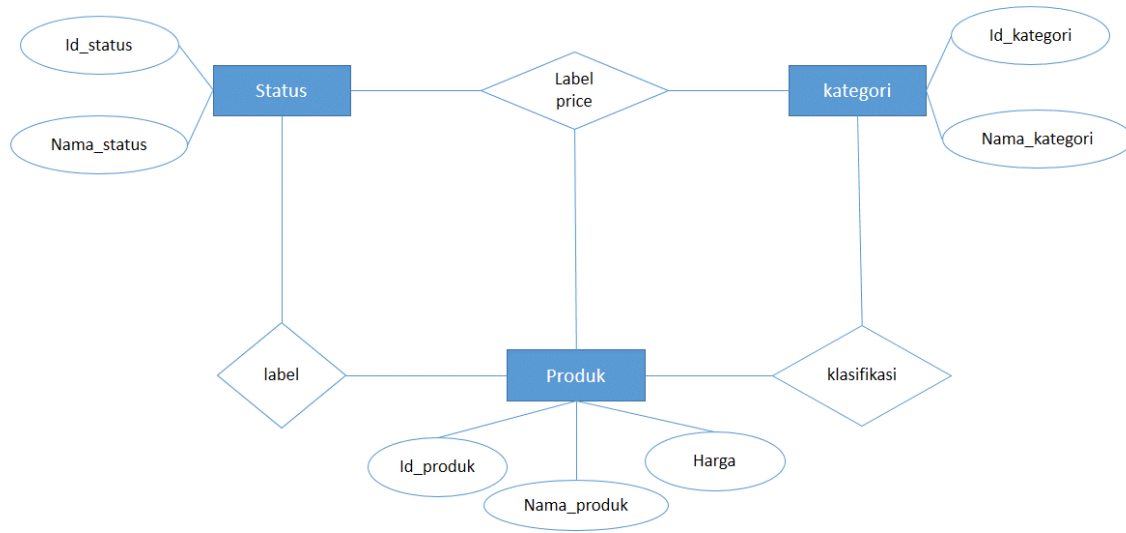
3. Hak akses sistem

User	Deskripsi
Admin	Untuk mengelola system secara keseluruhan mulai dari memasukan data ke database, menghapus data dari database, dan mengedit data dari database.

B. Design Sistem

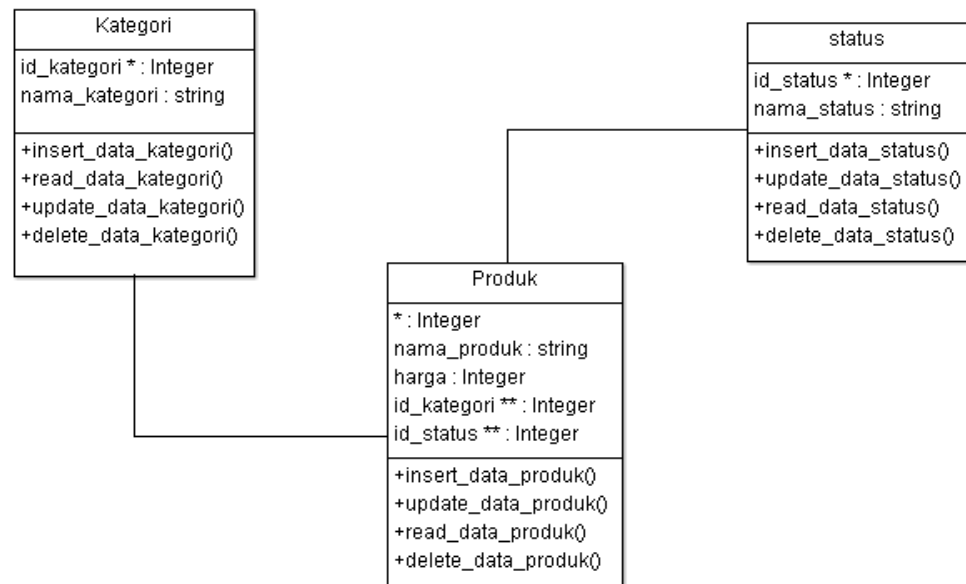
1. Entitas Relational Diagram (ERD)

Diagram yang digunakan untuk merancang suatu basis data, dipergunakan untuk memperlihatkan hubungan atau relasi antar entitas atau objek yang terlihat beserta atributnya.



2. Class Diagram

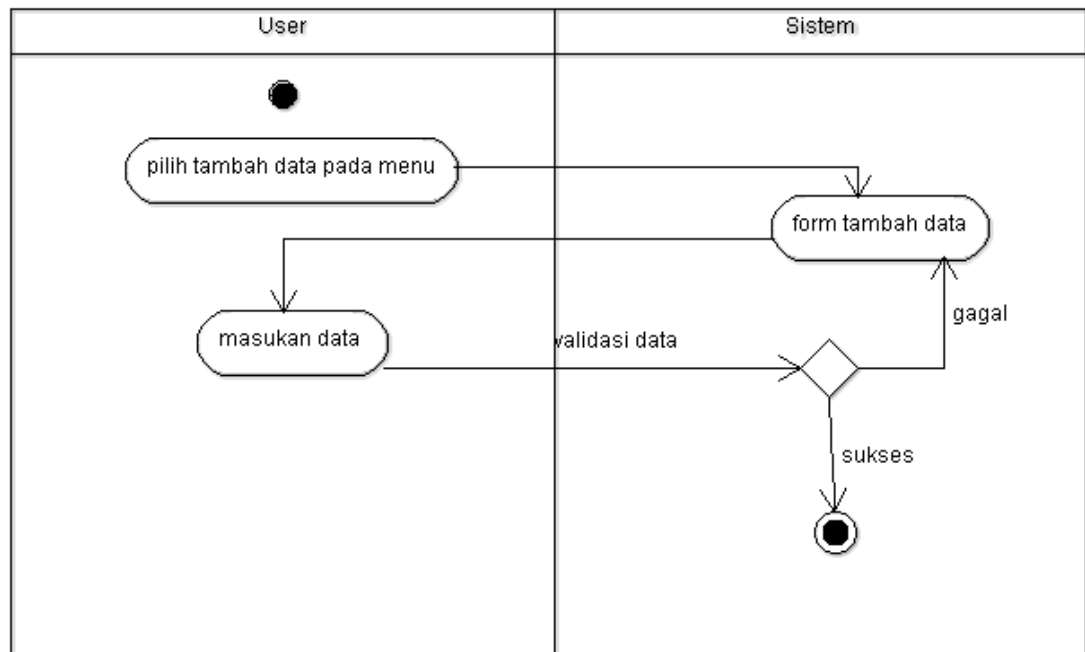
class diagram adalah salah satu jenis diagram struktur pada UML yang menggambarkan detail struktur dengan jelas



3. Diagram aktivitas

diagram yang dapat memodelkan proses-proses yang terjadi pada sebuah sistem.

Runtutan proses dari suatu sistem digambarkan secara vertikal.



C. Database

Database Name	Web Server	IP	Username	Password
fastprint	Web Server	Localhost	XXX	XXXX

Tabel Produk				
Field Name	Data Type & Length	Required	Primary Key	Description
Id_produk	Integer (12)	Yes	Yes	Pembeda setiap produk
nama_produk	varchar (255)	Yes	-	Informasi data dari sebuah nama produk
harga	integer (12)	Yes	-	Informasi data harga yang dipatok untuk setiap produk

Tabel status				
Field Name	Data Type & Length	Required	Primary Key	Description
Id_status	Integer (12)	Yes	Yes	Pembeda setiap status
nama_status	varchar (255)	Yes	-	Informasi data dari sebuah nama status

Tabel kategori				
Field Name	Data Type & Length	Required	Primary Key	Description
Id_kategori	Integer (12)	Yes	Yes	Pembeda setiap status
nama_kategori	varchar (255)	Yes	-	Informasi data dari sebuah nama kategori

D. Program aplikasi

1. File koneksi

Buat file koneksi.php untuk membuat settingan koneksi database agar nantinya dapat direlasikan kedalam file codingan yang lain

```

koneksi.php
1  <?php
2
3  $db_host = "localhost";
4  $db_user = "root";
5  $db_pass = "";
6  $db_name = "fastprint";
7
8  $connection = mysqli_connect($db_host, $db_user, $db_pass, $db_name);
9
10
11
12  ?>
```

2. Membuat CRUD database produk

2.1 Create data produk ke dalam database

- Buat file [tambah_produk.php](#) untuk membuat form tambah data produk untuk selanjutnya dikirim ke file [simpan_produk.php](#) dengan menggunakan metode POST

```
<form action="simpan_produk.php" method="POST">

    <div class="form-group">
        <label>Nama Produk</label>
        <input type="text" name="nama_produk" class="form-control" required>
    </div>

    <div class="form-group">
        <label>Harga</label>
        <input type="number" name="harga" class="form-control" required>
    </div>

    <div class="form-group">
        <label>id status</label>
        <input type="number" name="id_status" class="form-control" required>
    </div>

    <div class="form-group">
        <label>id kategori</label>
        <input type="number" name="id_kategori" class="form-control" required>
    </div>
```

- Buat file [simpan_produk.php](#) untuk menampung form dari file [tambah_data_produk.php](#) dan disimpan kedalam database ditabel produk dengan nama database 'fastprint'


```

simpan_produk.php
1  <?php
2
3  //include koneksi database
4  include('koneksi.php');
5
6  //get data dari form
7
8  $nama_produk = $_POST['nama_produk'];
9  $harga       = $_POST['harga'];
10 $id_status   = $_POST['id_status'];
11 $id_kategori  = $_POST['id_kategori'];
12
13 //query insert data ke dalam database
14 $query = "INSERT INTO produk (nama_produk, harga, id_status, id_kategori)
15 VALUES ('$nama_produk', '$harga', '$id_status', '$id_kategori')";
16
17 //kondisi pengecekan apakah data berhasil dimasukkan atau tidak
18 if($connection->query($query)) {
19     //redirect ke halaman index.php
20     header("location: index.php");
21 } else {
22     //pesan error gagal insert data
23     echo "Data Gagal Disimpan!";
24 }
25
26 ?>

```

2.1 Update data produk

- Buat form data produk yang sudah mengambil data dari database berdasarkan id

```

update_produk.php
1  <?php
2
3  include('koneksi.php');
4
5  $id = $_GET['id'];
6
7  $query = "SELECT * FROM produk WHERE id_produk = $id LIMIT 1";
8
9  $result = mysqli_query($connection, $query);
10
11 $row = mysqli_fetch_array($result);
12
13 ?>

```

Kemudian buat form dengan menggunakan metode POST dan dikirim ke file yang bernama `function_update_produk.php` lalu tampilkan data berdasarkan id dari masing field dalam table produk

```
<form action="function_update_produk.php" method="POST">

<div class="form-group">
  <label>ID Produk</label>
  <input type="text" name="id_produk" value="<?php echo $row['id_produk'] ?>"
    placeholder="Masukkan Nama Siswa" class="form-control">
</div>

<div class="form-group">
  <label>Nama Produk</label>
  <input type="text" name="nama_produk" value="<?php echo $row['nama_produk'] ?>"
    placeholder="Masukkan Nama Siswa" class="form-control">
</div>

<div class="form-group">
  <label>Harga</label>
  <input type="number" name="harga" value="<?php echo $row['harga'] ?>"
    placeholder="Masukkan Nama Siswa" class="form-control">
</div>

<div class="form-group">
  <label>Id Kategori</label>
  <input type="text" name="id_kategori" value="<?php echo $row['id_kategori'] ?>"
    placeholder="Masukkan Nama Siswa" class="form-control">
</div>

<div class="form-group">
  <label>Id Status</label>
  <input type="text" name="id_status" value="<?php echo $row['id_status'] ?>"
    placeholder="Masukkan Nama Siswa" class="form-control">
</div>
```

Buat file `function_update_produk.php` untuk menyimpan perubahan pada formulir

```

function_update_produk.php
1  <?php
2
3  //include koneksi database
4  include('koneksi.php');
5
6  //get data dari form
7  $id_produk = $_POST['id_produk'];
8  $nama_produk = $_POST['nama_produk'];
9  $harga = $_POST['harga'];
10 $id_kategori = $_POST['id_kategori'];
11 $id_status = $_POST['id_status'];
12
13 //query update data ke dalam database berdasarkan ID
14 $query = "UPDATE produk SET nama_produk =
15 '$nama_produk', harga = '$harga', id_kategori = '$id_kategori' WHERE id_produk = '$id_produk'";
16
17 //kondisi pengecekan apakah data berhasil diupdate atau tidak
18 if($connection->query($query)) {
19     //redirect ke halaman index.php
20     header("location: index.php");
21 } else {
22     //pesan error gagal update data
23     echo "Data Gagal Diupate!";
24 }
25
26 ?>

```

- Untuk menghapus data dari database buat file hapus_produk.php

```

<?php

include('koneksi.php');

//get id
$id = $_GET['id'];

$query = "DELETE FROM produk WHERE id_produk = '$id'";

if($connection->query($query)) {
    header("location: index.php");
} else {
    echo "DATA GAGAL DIHAPUS!";
}

?>

```

- Tampilkan data produk dengan menggunakan sql query

```
<?php
include('koneksi.php');
$no = 1;
$query = mysqli_query($connection,"SELECT * FROM produk
INNER JOIN kategori ON produk.id_kategori = kategori.id_kategori
INNER JOIN status ON produk.id_status = status.id_status");
while($row = mysqli_fetch_array($query)){
?>
```

- Lalu tampilkan kedalam html dan css agar lebih rapih

```
<td><?php echo $row['id_produk'] ?></td>
<td><?php echo $row['nama_produk'] ?></td>
<td><?php echo $row['harga'] ?></td>
<td><?php echo $row['nama_kategori'] ?></td>
<td><?php echo $row['nama_status'] ?></td>
```

3. Membuat CRUD database kategori

3.1 Create data kategori

- Buat file [tambah_kategori.php](#) untuk membuat form tambah data dan selanjutnya dikirim ke file [simpan_kategori.php](#) dengan menggunakan metode POST

```
<form action="simpan_kategori.php" method="POST">

<div class="form-group">
  <label>ID Kategori</label>
  <input type="number" name="id_kategori" class="form-control" required>
</div>

  <div class="form-group">
    <label>Nama Kategori</label>
    <input type="text" name="nama_kategori" class="form-control" required>
  </div>
```

- Buat file [simpan_kategori.php](#) untuk menyimpan kedalam database

```

<?php

//include koneksi database
include('koneksi.php');

//get data dari form
$id_kategori      = $_POST['id_kategori'];
$nama_kategori    = $_POST['nama_kategori'];

//query insert data ke dalam database
$query = "INSERT INTO kategori (id_kategori, nama_kategori)
VALUES ('$id_kategori', '$nama_kategori')";

//kondisi pengecekan apakah data berhasil dimasukkan atau tidak
if($connection->query($query)) {

    //redirect ke halaman index.php
    header("location: index.php");

} else {

    //pesan error gagal insert data
    echo "Data Gagal Disimpan!";

}

?>

```

3.2 Update kategori

- Buat file update_kategori.php, buat function sql yang mengambil data dari database berdasarkan id

```
<?php

include('koneksi.php');

$id = $_GET['id'];

$query = "SELECT * FROM kategori WHERE id_kategori = $id LIMIT 1";

$result = mysqli_query($connection, $query);

$row = mysqli_fetch_array($result);

?>
```

- Kemudian buat form yang sudah terisi file database berdasarkan id yang telah di set sebelumnya dan kirimkan form tersebut ke file functions_update_kategori.php dengan menggunakan metode POST

```
<form action="function_update_kategori.php" method="POST">

<div class="form-group">
  <label>ID kategori</label>
  <input type="text" name="id_kategori" value="<?php echo $row['id_kategori'] ?>"
    placeholder="Masukkan Nama Siswa" class="form-control">
</div>

  <div class="form-group">
    <label>Nama kategori</label>
    <input type="text" name="nama_kategori" value="<?php echo $row['nama_kategori'] ?>"
      placeholder="Masukkan Nama Siswa" class="form-control">
  </div>

</form>
```

- Buat file function_update_kategori.php untuk membuat function yang dapat menyimpan perubahan kedalam database

```

function_update_kategori.php
1  <?php
2
3  //include koneksi database
4  include('koneksi.php');
5
6  //get data dari form
7  $id_kategori = $_POST['id_kategori'];
8  $nama_kategori = $_POST['nama_kategori'];
9
10
11  //query update data ke dalam database berdasarkan ID
12  $query = "UPDATE kategori SET nama_kategori = '$nama_kategori'
13  WHERE id_kategori = '$id_kategori'";
14
15  //kondisi pengecekan apakah data berhasil diupdate atau tidak
16  if($connection->query($query)) {
17      //redirect ke halaman index.php
18      header("location: index.php");
19  } else {
20      //pesan error gagal update data
21      echo "Data Gagal Diupate!";
22  }
23
24  ?>

```

3.3 hapus data dari database berdasarkan id yang di request

```

hapus_kategori.php
1  <?php
2
3  include('koneksi.php');
4
5  //get id
6  $id = $_GET['id'];
7
8  $query = "DELETE FROM kategori WHERE id_kategori = '$id'";
9
10 if($connection->query($query)) {
11     header("location: index.php");
12 } else {
13     echo "DATA GAGAL DIHAPUS!";
14 }
15
16 ?>

```

3.4 Tampilkan seluruh data kategori dari database

- Didalam file index.php buat query database ke table kategori dan ambil semua data kategori dari database dengan menggunakan syntax

```

<?php
include('koneksi.php');
$no = 1;
$query = mysqli_query($connection,"SELECT * FROM Kategori
");

```

- Kemudian di file yang sama tampilkan data kategori

```

while($row = mysqli_fetch_array($query)){
?>

<tr>

<td><?php echo $row['id_kategori'] ?></td>

<td><?php echo $row['nama_kategori'] ?></td>

```

4. CRUD data status

4.1 Create database status

- Buat file tambah_status.php untuk membuat form, lalu kirimkan form dengan menggunakan metode POST ke file simpan_status.php

- \

```
<form action="simpan_status.php" method="POST">

<div class="form-group">
    <label>ID Status</label>
    <input type="text" name="id_status" class="form-control" required>
</div>

<div class="form-group">
    <label>Nama Status</label>
    <input type="text" name="nama_status" class="form-control" required>
</div>
```

- Buat file `simpan_status.php` untuk menyimpan ke dalam database

```
simpan_status.php
1  <?php
2
3  //include koneksi database
4  include('koneksi.php');
5
6  //get data dari form
7
8
9  $id_status = $_POST
10 $nama_status = $_POST['nama_status'];
11
12 //query insert data ke dalam database
13 $query = "INSERT INTO status (id_status, nama_status) VALUES ('$id_status', '$nama_status')";
14
15 //kondisi pengecekan apakah data berhasil dimasukkan atau tidak
16 if($connection->query($query)) {
17
18     //redirect ke halaman index.php
19     header("location: index.php");
20
21 } else {
22
23     //pesan error gagal insert data
24     echo "Data Gagal Disimpan!";
25
26 }
27
28 ?>
```

An associative array of variables passed to the current script via the HTTP POST method.

4.2 Update data status

- Buat file `update_status.php` untuk membuat form update data status, pertama buat query database

```
<?php

include('koneksi.php');

$id = $_GET['id'];

$query = "SELECT * FROM status WHERE id_status = $id LIMIT 1";

$result = mysqli_query($connection, $query);

$row = mysqli_fetch_array($result);

?>
```

- buat form di file update_status.php lalu tampilkan data berdasarkan id di form tersebut dengan menggunakan metode POST lalu kirimkan data tersebut ke file function_update_status.php

```
<div class="form-group">
<form action="function_update_status.php" method="POST">

<div class="form-group">
  <label>ID Status</label>
  <input type="text" name="id_status" value="<?php echo $row['id_status'] ?>"
    placeholder="Masukkan Nama Siswa" class="form-control">
</div>

<div class="form-group">
  <label>Nama Status</label>
  <input type="text" name="nama_status" value="<?php echo $row['nama_status'] ?>"
    placeholder="Masukkan Nama Siswa" class="form-control">
</div>
```

- buat file function_update _status.php untuk menyimpan perubahan kedalam database

```

function_update_status.php
1  <?php
2
3  //include koneksi database
4  include('koneksi.php');
5
6  //get data dari form
7  $id_status = $_POST['id_status'];
8  $nama_status = $_POST['nama_status'];
9
10
11 //query update data ke dalam database berdasarkan ID
12 $query = "UPDATE status SET nama_status = '$nama_status' WHERE id_status = '$id_status'";
13
14 //kondisi pengecekan apakah data berhasil diupdate atau tidak
15 if($connection->query($query)) {
16     //redirect ke halaman index.php
17     header("location: index.php");
18 } else {
19     //pesan error gagal update data
20     echo "Data Gagal Diupate!";
21 }
22
23 ?>

```

5. Tampilan layout

5.1 Tambah data

TAMBAH Produk

Nama Produk

Harga

id status

id kategori

SIMPAN

RESET

5.2 Update data

EDIT Produk

ID Produk

2

Nama Produk

sepatu

Harga

600

Id Kategori

1

Id Status

1

UPDATE

RESET

5.3 Tampilan data

DATA Produk					
<div>TAMBAH PRODUK</div>					
ID Produk	NAMA produk	harga	kategori	status	
2	sepatu	600	sport (olahraga)	dijual	<div>EDIT</div> <div>HAPUS</div>
3	sendal	129900	sport (olahraga)	dijual	<div>EDIT</div> <div>HAPUS</div>
4	sepatu futsal	1200000	kecantikan	dijual	<div>EDIT</div> <div>HAPUS</div>
5	kaos kaki baggus	15000	sport (olahraga)	dijual	<div>EDIT</div> <div>HAPUS</div>