

**LAPORAN
PEMOGRAMAN BERORIENTASI OBJEK**



OLEH:

NAMA : ZUL FADLY RAHMAN

NIM : F1G120055

KELOMPOK : II (DUA)

ASISTEN PENGAMPU:

WAHID SAFRI JAYANTO

PROGRAM STUDI ILMU KOMPUTER

FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM

UNIVERSITAS HALU OLEO

2021

HALAMAN PENGESAHAN LAPORAN PRAKTIKUM



OLEH
ZUL FADLY RAHMAN (F1G120055)

Laporan praktikum Pemograman Berorientasi Objek ini disusun sebagai tugas akhir menyelesaikan praktikum Pemograman Berorientasi Objek sebagai salah satu syarat lulus matakuliah Pemograman Berorientasi Objek. Menerangkan bahwa yang tertulis dalam laporan lengkap ini adalah benar dan dinyatakan telah memenuhi syarat.

Kendari, Desember 2021

Menyetujui,

Asisten Pratikum,

Wahid Safri Jayanto

F1G117059

Pratikan,



Zul Fadly Rahman

F1G120050

DAFTAR ISI

LAPORAN.....	i
HALAMAN PENGESAHAN LAPORAN PRAKTIKUM.....	ii
DAFTAR ISI.....	iii
DAFTAR GAMBAR	vi
DAFTAR TABEL.....	vii
KATA PENGANTAR	viii
PRAKTIKUM 1	1
1.1 Pendahuluan	1
1.2 Waktu Dan Tempat	1
1.3 Alat Dan Bahan	2
PRAKTIKUM 2.....	3
2.1 Pengerian Class, Objek, Property Dan Method	3
2.1.1 Class.....	3
2.1.2 Property	3
2.1.3 Method.....	4
2.1.4 Object.....	4
2.2 Constructor dan destructor	5

2.2.1 Constructor	5
2.2.2 Destructor	5
2.3 Enkapsulasi (<i>encapsulation</i>)	6
2.3.1 Public	7
2.3.2 Protected	7
2.3.3 Private	7
2.4 Interfaces	8
PRAKTIKUM 3.....	9
3.1 Model data berbasis objek.....	9
3.1.1 Entity relationship model.....	9
3.1.2 Semantic model	10
3.2 Model data berbasis record	10
3.2.1 Relational model.....	10
3.2.2 Hierarchical model	10
3.2.3 Network model	11
3.3 Entitas (Entity)	12
3.4 Atribut	12
3.4.1 Tipe-tipe atribut	12
3.5 Relationship.....	13
3.6 Pengenalan crud	15

3.6.1 Create.....	15
3.6.2 Read.....	15
3.6.3 Update.....	15
3.6.4 Delete.....	16
DAFTAR PUSTAKA	17

DAFTAR GAMBAR

Gambar 3.1 Contoh hierarchical model	11
Gambar 3.2 Network Model	12
Gambar 3.3 Unary Degree	14
Gambar 3.4 Binary Degree	14
Gambar 3.5 Ternary Degree	14

DAFTAR TABEL

table 1. 1 Table alat dan bahan.....	2
--------------------------------------	---

KATA PENGANTAR

Assalamu Alaikum Wr. Wb.

Puji syukur kami panjatkan kehadiran Allah SWT yang senantiasa melimpahkan rahmat dan ridho-Nya, sehingga kami dapat menyelesaikan penyusunan laporan praktikum Pemrograman Berorientasi Objek

Adapun laporan ini kami telah usahakan semaksimal mungkin dan tentunya dengan bantuan berbagai pihak, sehingga dapat memperlancar pembuatan laporan ini. Untuk itu kami tak lupa pula menyampaikan banyak terimakasih kepada semua pihak yang telah membantu dalam pembuatan laporan ini.

Namun, kami menyadari jika dalam menyusun laporan praktikum ini masih jauh dari kata sempurna. Oleh karena itu, dengan hati yang terbuka kritik serta saran yang konstruktif guna kesempurnaan laporan ini. Demikian laporan ini kami susun, apabila ada kata-kata yang kurang berkenan dan banyak terdapat kekurangan, kami mohon maaf yang sebesar- besarnya.

Semoga bermanfaat....

Kendari, Desember 2021

Penyusun

PRAKTIKUM 1

1.1 Pendahuluan

Object Oriented Programming (OOP) merupakan model pemrograman yang berbasis pada konsep obyek, diantaranya berisi data, sering dikenal sebagai atribut dan kode, dalam bentuk prosedur, sering dikenal sebagai metode. Sebuah fitur dari obyek adalah bahwa prosedur obyek dapat mengakses dan sering memodifikasi data dari obyek yang saling berhubungan. Dalam OOP, program dirancang dengan membuat obyek yang dapat berinteraksi satu sama lain. Pemrograman berorientasi obyek didefinisikan menggunakan objek, tetapi tidak semua teknik terkait dan struktur dalam bahasa pemrograman mendukung OOP.

PHP adalah bahasa pemrograman untuk dijalankan melalui halaman web, umumnya digunakan untuk mengolah informasi di internet. Sedangkan dalam pengertian lain PHP adalah singkatan dari Hypertext Preprocessor yaitu bahasa pemrograman webserver-side yang bersifat open source atau gratis. PHP merupakan script yang menyatu dengan HTML dan berada pada server (Kurniawan, 2010:2).

1.2 Waktu Dan Tempat

Kegiatan praktikum Pemrograman Berorientasi Objek ini dimulai dari tanggal 30 September sampai 2 Desember dilaksanakan setiap hari Kamis pukul 10.00-12.00 WITA di Laboratorium Aljabar lantai 3 gedung A, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Halu Oleo, Kendari.

1.3 Alat Dan Bahan

Adapun alat dan bahan yang di gunakan pada praktikum kali ini adalah sebagai berikut:

No.	Alat dan Bahan	Kegunaan
1.	Laptop	Sebagai tempat untuk menyimpan data, untuk mengerjakan projek dan sebagai tempat untuk mengoding.
2.	<i>Xampp</i>	Sebagai penghubung antara <i>chrome</i> dan <i>sublime</i> .
3.	<i>Sublime</i>	Sebagai tempat mengoding sebuah program.
4.	<i>Crome</i>	Sebagai tempat untuk melihat hasil <i>running</i> dari program yang telah di buat.

table 1. 1 Table alat dan bahan

PRAKTIKUM 2

2.1 Pengerian Class, Objek, Property Dan Method

2.1.1 Class

Class adalah sebuah spesifikasi yang jika di instansikan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan design berorientasi objek. Diagram ini menjelaskan bagaimana hubungan antar class pada aplikasi tersebut terjadi yang terdiri dari nama class, attribute, dan operation. Class diagram menggambarkan hubungan antar class dalam sistem yang sedang dibangun dan bagaimana mereka saling berkolaborasi untuk mencapai suatu tujuan.

```
<?php
class laptop {
    // isi dari class laptop...
}
?>
```

2.1.2 Property

Property (atau disebut juga dengan atribut) adalah data yang terdapat dalam sebuah class. Melanjutkan analogi tentang laptop, property dari laptop bisa berupa merk, warna, jenis processor, ukuran layar, dan lain-lain.

Jika anda sudah terbiasa dengan program PHP, property ini sebenarnya hanyalah variabel yang terletak di dalam class. Seluruh aturan dan tipe data yang biasa diinput ke dalam variabel, bisa juga diinput kedalam property. Aturan tata cara penamaan property sama dengan aturan penamaan variabel.

```

<?php
class laptop {
    var $pemilik;
    var $merk;
    var $ukuran_layar;
    // lanjutan isi dari class laptop...
}
?>

```

2.1.3 Method

Method adalah tindakan yang bisa dilakukan di dalam class. Jika menggunakan analogi class laptop kita, maka contoh method adalah: menghidupkan laptop, mematikan laptop, mengganti cover laptop, dan berbagai tindakan lain.

Method pada dasarnya adalah function yang berada di dalam class. Seluruh fungsi dan sifat function bisa diterapkan ke dalam method, seperti argumen/parameter, mengembalikan nilai (dengan keyword return), dan lain-lain.

```

<?php
class laptop {
    var $pemilik;
    var $merk;
    var $ukuran_layar;
    // lanjutan isi dari class laptop...
}
?>

```

2.1.4 Object

Object atau Objek adalah hasil cetak dari class, atau hasil ‘konkrit’ dari class. Jika menggunakan analogi class laptop, maka objek dari class laptop bisa berupa: laptop_andi, laptop_anto, laptop_duniaikom, dan lain-lain. Objek dari class laptop akan memiliki seluruh ciri-ciri laptop, yaitu property dan method-nya.

```
<?php
class laptop {
    function hidupkan_laptop() {
        //... isi dari method hidupkan_laptop
    }

    function matikan_laptop() {
        //... isi dari method matikan_laptop
    }

    ... //isi dari class laptop
}
?>
```

2.2 Constructor dan destructor

2.2.1 Constructor

Constructor (bahasa indonesia: konstruktor) adalah method khusus yang akan dijalankan secara otomatis pada saat sebuah objek dibuat (instansiasi), yakni ketika perintah “new” dijalankan.

Constructor biasa digunakan untuk membuat proses awal dalam mempersiapkan objek, seperti memberi nilai awal kepada property, memanggil method internal dan beberapa proses lain yang digunakan untuk ‘mempersiapkan’ objek.

Dalam PHP, constructor dibuat menggunakan method : __construct().

2.2.2 Destructor

Destructor (bahasa indonesia: destruktur) adalah method khusus yang dijalankan secara otomatis pada saat sebuah objek dihapus. Di dalam PHP, seluruh objek secara otomatis dihapus ketika halaman PHP dimana objek itu berada selesai diproses. Tetapi kita juga dapat menghapus objek secara manual.

Destructor biasanya digunakan untuk ‘membersihkan’ beberapa variabel, atau menjalankan proses tertentu sebelum objek dihapus.

Dalam PHP, destructor dibuat menggunakan method : `__destruct()`.

PHP memiliki fitur ‘penghapusan objek massal’ yang dikenal dengan Garbage Collection. Fitur garbage collection akan menghapus seluruh objek secara otomatis ketika halaman PHP selesai di eksekusi, sehingga akan memanggil method `__destruct` dari seluruh objek. Namun PHP menyediakan cara jika kita ingin menghapus objek sebelum mekanisme ini berjalan, yaitu menggunakan fungsi `unset()` atau mengisi variabel objek dengan nilai null.

2.3 Enkapsulasi (*encapsulation*)

Enkapsulasi atau encapsulation adalah sebuah metode untuk mengatur struktur class dengan cara menyembunyikan alur kerja dari class tersebut. Hal ini bertujuan untuk menyembunyikan informasi dari method dan property dengan alasan keamanan.

Struktur class yang dimaksud adalah property dan method, sehingga hanya property dan method tertentu saja yang bisa diakses dari luar class. Enkapsulasi juga dikenal dengan istilah “informating hiding”.

Dengan enkapsulasi, kita bisa memilih property dan method apa saja kah yang boleh diakses, dan mana saja yang tidak boleh diakses. Dengan menghalangi kode program lain untuk mengubah property tertentu, class menjadi lebih terintegrasi dan juga menghindari kesalahan ketika seseorang mencoba untuk mengubahnya. Seorang programer yang merancang class bisa menyediakan property dan method khusus yang memang ditujukan untuk akses diluar.

Untuk membatasi juga memberikan hak akses kepada property dan method pada sebuah class, di dalam OOP terdapat 3 kata kunci, yaitu:

2.3.1 Public

Jika sebuah method ataupun property dinyatakan sebagai public, maka method dan property tersebut dapat diakses dari luar ataupun dari dalam class tersebut, termasuk class turunan. Namun perlu kita ketahui, pada method dan juga property yang tidak ditetapkan hak aksesnya menggunakan public, private dan protected, maka hak akses pada method dan property tersebut adalah secara default menjadi public. Sama halnya seperti penulisan OOP yang sudah kita pelajari sebelumnya.

2.3.2 Protected

Jika sebuah property ataupun method dinyatakan sebagai protected, berarti property ataupun method tersebut tidak bisa diakses dari luar class. Namun bisa diakses oleh class itu sendiri atau turunan dari class tersebut. Apabila kita mencoba mengakses protected property atau protected method dari luar class, maka akan menghasilkan error.

2.3.3 Private

Private adalah sebuah hak akses yang melarang method atau property yang menggunakannya dilarang akses dari luar class. Jadi hak akses private ini hanya bisa diakses dari dalam class itu sendiri.

Jadi, jika sebuah property atau method dijadikan sebagai private, maka satu-satunya yang bisa mengakses adalah class itu sendiri. Class lain tidak bisa mengaksesnya, termasuk class turunannya. Akses private ini sering digunakan untuk menyembunyikan

property dan method agar tidak bisa diakses dari luar class.

2.4 Interfaces

Interface adalah jenis khusus dari blok yang hanya berisi method signature (atau constant). Interface mendefinisikan sebuah(signature) dari sebuah kumpulan method tanpa tubuh. Interface mendefinisikan sebuah cara standar dan umum dalam menetapkan sifat-sifat dari class-class. Mereka menyediakan class-class, tanpa memperhatikan lokasinya dalam hirarki class, untuk mengimplementasikan sifat-sifat yang umum. Dengan catatan bahwa interface-interface juga menunjukkan polimorfisme, dikarenakan program dapat memanggil method interface dan versi yang tepat dari method yang akan dieksekusi tergantung dari tipe object yang melewati pemanggil method interface

PRAKTIKUM 3

3.1 Model data berbasis objek

Dalam pemodelan ini struktur atau hirarki basis data diilustrasikan berdasarkan object. Model ini meliputi: Model keterhubungan entitas (Entity Relationship Model) atau ERD, Model berorientasi object (Object-Oriented Model), Model Data Semantik (Semantic Data Model), Model data Fungsional (Function Data Model).

3.1.1 Entity relationship model

Diagram Hubungan Entitas atau entity relationship diagram merupakan model data berupa notasi grafis dalam pemodelan data konseptual yang menggambarkan hubungan antara penyimpan. Model data sendiri merupakan sekumpulan cara, peralatan untuk mendeskripsikan data-data yang hubungannya satu sama lain, semantiknya, serta batasan konsistensi.

Model data terdiri dari model hubungan entitas dan model relasional. Diagram hubungan entitas ditemukan oleh Peter Chen dalam buku Entity Relational Model-Toward a Unified of Data. Chen mencoba merumuskan dasar-dasar model dan setelah itu dikembangkan dan dimodifikasi oleh Chen dan banyak pakar lainnya. Pada saat itu diagram hubungan entitas dibuat sebagai bagian dari perangkat lunak yang juga merupakan modifikasi khusus, karena tidak ada bentuk tunggal dan standar dari diagram hubungan entitas.

3.1.2 Semantic model

Pada dasarnya semantic Model memiliki arti yang hampir sama dengan entity relationship model. Hanya perbedaan yang tampak pada relasi objek dasar yang tidak dinyatakan dengan simbol melainkan menggunakan kata-kata (semantic).

3.2 Model data berbasis record

Dalam model ini struktur basis data diilustrasikan berdasarkan record. Model ini meliputi: Model relational (Relational Model), Model Hierarkis (Hierarchical Model) dan Model Jaringan (Network Model).

3.2.1 Relational model

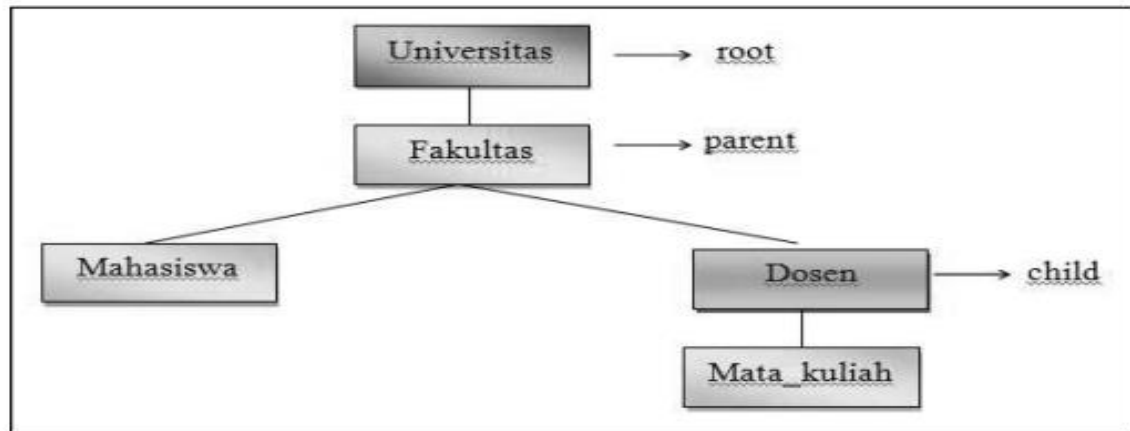
Representasi dalam bentuk tabel yang terdapat sejumlah Baris yang menunjuk record dan kolom yang menunjuk atribut. Model ini banyak digunakan dalam pemodelan dan perancangan Basis Data. Konsep dan terminologi yang digunakan mirip dengan kondisi real yang dihadapi oleh pemakai sehingga mudah dipahami.

Representasi dalam bentuk tabel yang terdapat sejumlah Baris yang menunjuk record dan kolom yang menunjuk atribut. Model ini banyak digunakan dalam pemodelan dan perancangan Basis Data. Konsep dan terminologi yang digunakan mirip dengan kondisi real yang dihadapi oleh pemakai sehingga mudah dipahami.

3.2.2 Hierarchical model

Dalam model ini data disusun menurut struktur pohon. Puncak dari hierarki disebut dengan root sedangkan entitas atau interface di bawahnya dikenal sebagai induk (parent). Entitas induk mempunyai beberapa sub entitas yang disebut anak

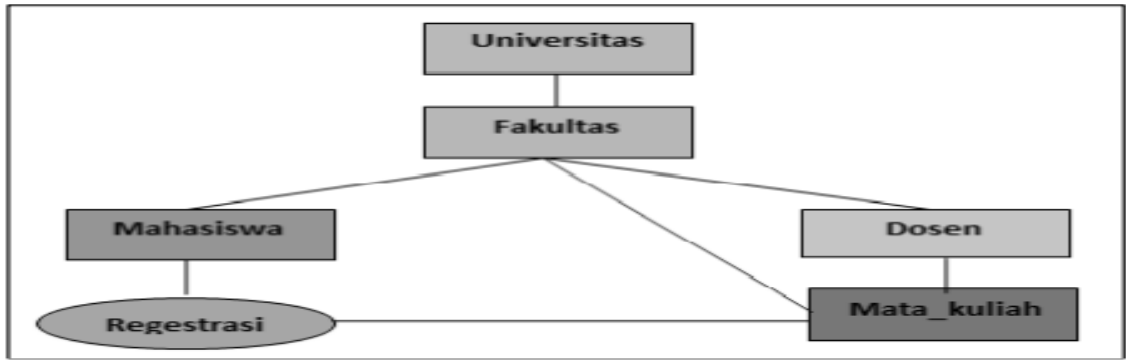
(child). Entitas dalam model hirarki dilambangkan dengan empat persegi panjang. Sedangkan relasi atau hubungan dengan entitas lain dinotasikan dengan garis. Gambar dibawah ini menjelaskan salah satu contoh model hirarki basis data level konseptual sistem perkuliahan.



Gambar 3.1 Contoh hierarchical model

3.2.3 Network model

Dalam model jaringan entitas induk maupun anak bisa lebih dari dua. Model ini merupakan pengembangan model hirarki. Relasi antara entitas dalam network model adalah satu ke satu (one to one) atau satu ke banyak (one to many). Dalam network data model tidak diperbolehkan terdapat relasi banyak ke banyak (many to many). Untuk membuat relasi many to many dalam network 67 Basis Data model dibutuhkan entitas perantara yang disebut sebagai rekaman silang (intersection record).



Gambar 3.2 Network Model

3.3 Entitas (Entity)

Entitas digambarkan dalam bentuk persegi panjang. Entitas adalah individu yang mewakili sesuatu yang nyata dan dapat dibedakan dari sesuatu yang lain. Entitas dapat berupa orang, tempat, objek, atau kejadian. Contoh pekerja, supplier, bahan baku, dan lain-lain.

3.4 Atribut

Atribut merupakan gambaran karakteristik dari sebuah entitas atau himpunan entitas.

Contoh :

atribut untuk himpunan entitas mahasiswa adalah nim, nama, alamat, ipk, program studi, hobi, dsb.

3.4.1 Tipe-tipe atribut

1. Atribut Simple yaitu suatu atribut yang tidak bisa dibagi menjadi bagian yg lebih kecil lagi. Contoh atribut simple adalah Jenis Kelamin.
2. Atribut Composite yaitu suatu atribut yang dapat dibagi menjadi beberapa bagian.

Contoh

atribut composite Nama dapat di bagi menjadi nama depan dan nama belakang.

3. Atribut Single value yaitu suatu atribut yang bisa di isi paling banyak 1 nilai untuk setiap baris

data. Contoh atribut single value adalah Jenis Kelamin.

4. Atribut Multi valued yaitu suatu atribut yang bisa lebih dari 1 nilai yang sejenis untuk setiap baris

data. Contoh atribut multivalued value adalah Alamat, No telp dan hobi. Ketiga atribut tersebut

bisa berisi lebih dari 1. Contoh untuk 1 entitas orang bisa mempunyai lebih dari 1 nilai untuk

atribut hobi yang isinya musik, olahraga begitu juga untuk telp dan alamat (* karena bisa

mempunyai > 1 no telp dan > 1 alamat)

5. Derived Attribute yaitu suatu atribut yang nilainya didapatkan dari hasil pengolahan atribut lain.

Contoh atribut derived adalah umur yaitu didapatkan dari perhitungan tanggal lahir dan tanggal

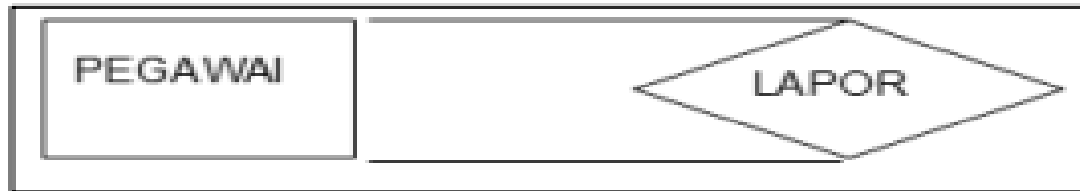
sekarang. IPK yang didapatkan dari penjumlahan nilai di bagi dengan jumlah sks yang diambil.

3.5 Relationship

Relasi atau relationship merupakan hubungan yang terjadi antara satu atau lebih entitas. Derajat dari relationship menjelaskan jumlah entitas yang berpartisipasi dalam suatu relationship. Klasifikasi relasi berdasarkan derajatnya adalah:

1. Unary Degree (Derajat Satu)

Relationship yang hanya melibatkan 1 entitas.



Gambar 3.3 Unary Degree

2. Binary Degree (Derajat Dua)

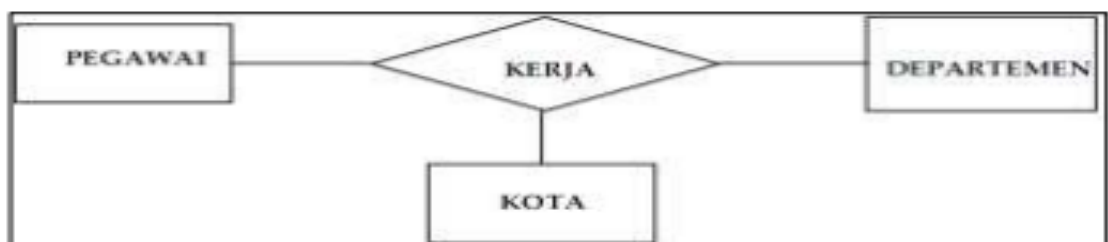
Relationship yang melibatkan 2 entitas.



Gambar 3.4 Binary Degree

3. Ternary Degree (Derajat tiga)

Relationship yang melibatkan 3 entitas.



Gambar 3.5 Ternary Degree

3.6 Pengenalan crud

CRUD adalah singkatan dari create, read, update, and delete. Keempat poin ini menurut Techopedia merupakan fungsi-fungsi utama yang diimplementasikan dalam aplikasi database. Akronim CRUD mengidentifikasi semua fungsi utama yang melekat pada database relasional dan aplikasi yang digunakan untuk mengelolanya, termasuk Oracle Database, Microsoft SQL Server, MySQL, dan lainnya.

3.6.1 Create

Fungsi CRUD yang pertama adalah create. Fungsi ini memungkinkanmu membuat record baru dalam database. Dalam aplikasi SQL, fungsi create sering disebut juga sebagai insert. Kamu dapat membuat baris baru dan mengisinya dengan data yang sesuai dengan setiap atribut. Tetapi, hanya administrator yang dapat menambahkan atribut baru ke tabel itu sendiri.

3.6.2 Read

Fungsi read hampir mirip dengan fungsi search. Fungsi ini memungkinkan kamu untuk mencari dan mengambil data tertentu dalam tabel dan membaca nilainya. Kamu dapat menemukan record yang diinginkan menggunakan kata kunci, atau dengan memfilter data berdasarkan kriteria yang diinginkan.

3.6.3 Update

Untuk memodifikasi record yang telah tersimpan di database, fungsi CRUD yang bisa kamu gunakan adalah fungsi update. Namun, kamu perlu mengubah berbagai informasi terkait agar bisa memodifikasi record yang diinginkan secara utuh.

Record yang ada dalam database harus diubah dan semua nilai atribut diubah untuk mencerminkan karakteristik baru yang diinginkan.

3.6.4 Delete

Ketika ada record atau data yang tidak lagi dibutuhkan dalam database, fungsi CRUD yang digunakan adalah fungsi delete. Fungsi ini dapat digunakan untuk menghapus data tersebut.

Beberapa aplikasi database relasional mungkin mengizinkan kamu untuk melakukan hard delete atau soft delete. Hard delete akan secara menghapus catatan dari database permanen. Sementara, soft delete hanya akan memperbarui status baris untuk menunjukkan bahwa data telah dihapus meskipun data tersebut tetap ada dan utuh.

DAFTAR PUSTAKA

Apa Itu Entity Relationship Diagram (ERD) dan Komponennya. idcloudhost.com. ([Apa Itu Entity Relationship Diagram \(ERD\) dan Komponennya - IDCloudHost](#)). Diakses pada tanggal 9 Desember 2021.

Adhiwibowo, W., April Firman, D. 2017. Model Pengembangan Aplikasi Pembayaran Angsuran Pinjaman Online Menggunakan *Php-Mysql* Dengan Metode *Object Oriented Programming*. *Jurnal Informatika Upgris*. Vol 3(2). Hal 92-98.

Berkenalan dengan CRUD, Elemen Penting dalam Back End Development. glints.com. ([CRUD: Definisi, Fungsi, dan Cara Mengaplikasikannya - Glints Blog](#)). Diakses 9 Desember 2021.

Christianti J. Meliana, dkk. 2015. Analisis dan Perancangan Aplikasi Penyusunan Jadwal Mengajar Sesuai Data Ketersediaan Mengajar Dosen. *Jurnal Teknik Informatika dan Sistem Informasi*. Vol 1(2). Hal 132-141.

Maiyana, E ., Mira. S. , Fitri, R. 2019. Penyewaan *Online Wedding Organizer* Berbasis Web. *Jurnal Sains dan Informatika*. Vol 5(1). Hal 24-30.

Maryam, S. Modul Basis Data Tingkat XI – Semester 3. SMK Negeri 1 Cimahi.

Munif, A. 2013. Basis Data Edisi Pertama. Jakarta: Kementerian Pendidikan & Kebudayaan.

Nimas. 7 juli 2021. Penjelasan Jenis Model Data Pada Sistem Basis Data. Pro.Co.Id. [Penjelasan Jenis Model Data Pada Sistem Basis Data \(pro.co.id\)](#)

Nursyifa, T. 20 Aug 2019. Pengertian enkapsulasi (public, private dan protected) OOP PHP. plimbi.com. [Pengertian enkapsulasi \(public, private dan protected\) OOP PHP - Article - Plimbi Social Journalism | Plimbi.com.](#)

Pahlevi O., Astriana M. & Miftahul K. 2018. Sistem Informasi Inventori Barang Menggunakan Metode *Object Oriented* Di Pt. Livaza Teknologi Indonesia Jakarta. *Jurnal PROSISKO* Vol. 5(1). Hal 27-35

Sandyayuda. 28 april 2013. Perbedaan Abstract Class dan Interface. [Perbedaan Abstract Class dan Interface | Sandya Yuda Pratama \(wordpress.com\).](#)

Tutorial Belajar OOP PHP Part 2: Pengertian Class, Object, Property dan Method. Duniaikom ([Pengertian Class, Object, Property dan Method | Belajar OOP PHP | Duniaikom](#)) diakses pada tanggal 9 Desember 2021.