

Nama : Muhammad Arif

NIM : 20200410700036

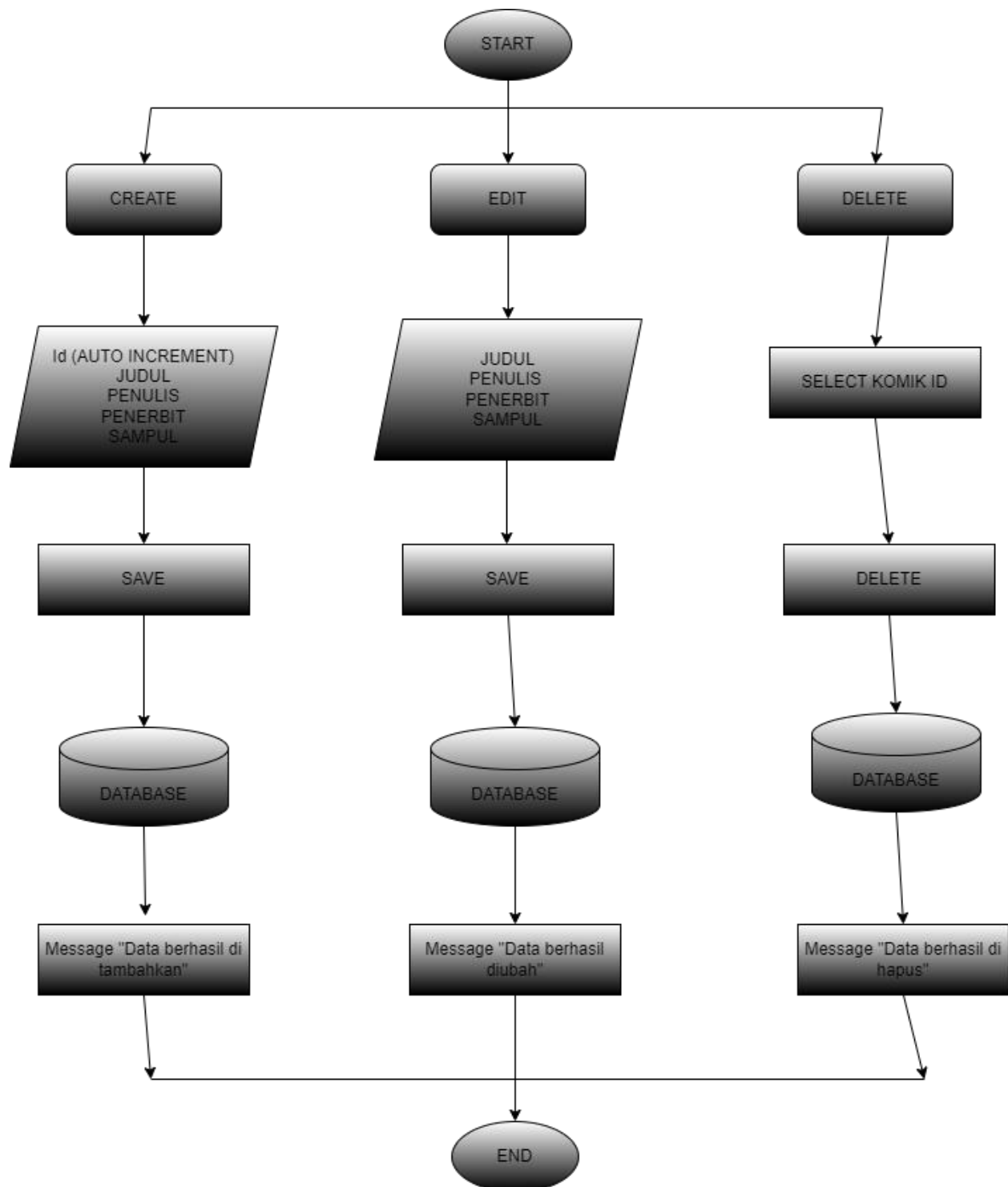
Ujikom Programming

1. Tema : Program CRUD Komik sederhana menggunakan framework CodeIgniter 4 dan style menggunakan Bootstrap.

Dalam program ini saya menggunakan codeigniter 4 sebagai framework PHP untuk mengelola data komik, memiliki model komik untuk mempresentasikan entitas komik, controller komik untuk mengelola CRUD, dan dalam style saya menggunakan Bootstrap v5.

2. Secara umum proses pengerjaan dimulai dari perancangan atau pemodelan, kemudian pembuatan source code, kompilasi atau build, debugging, dokumentasi, dan melakukan unit test.

Flowchart Komik

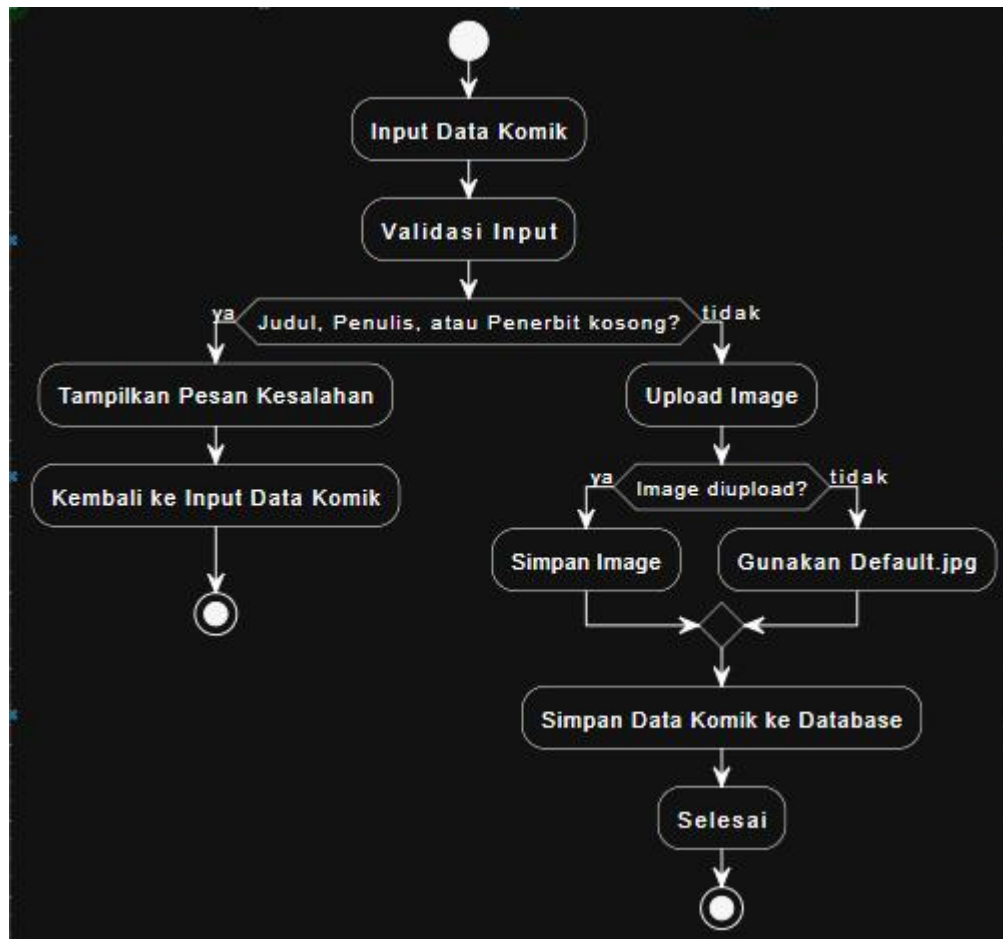


Tabel `komik`

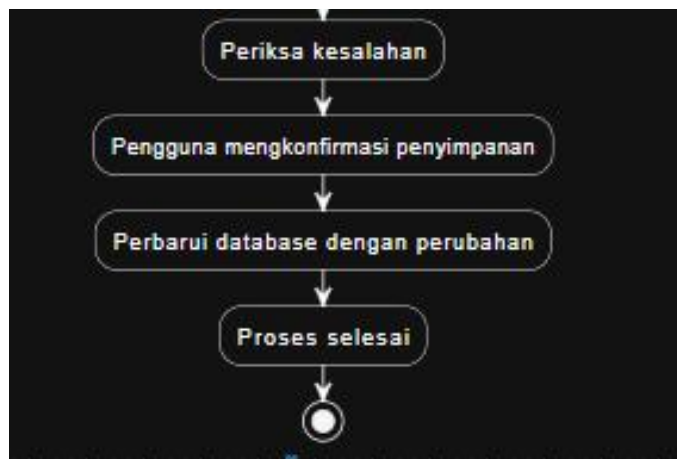
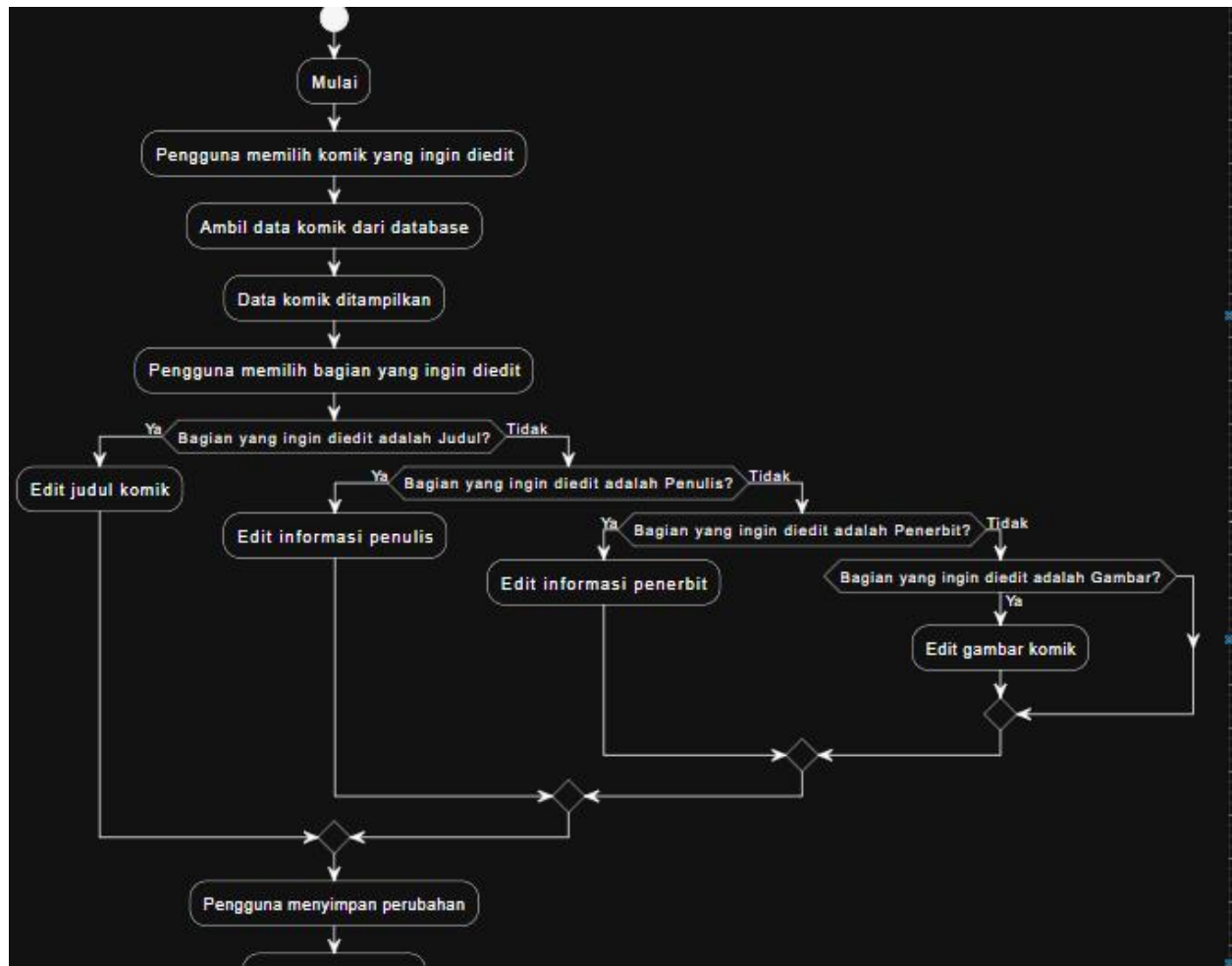
Kolom	Tipe Data	Deskripsi
id	INT	Primary key, auto increment
judul	VARCHAR(255)	Judul komik
slug	VARCHAR(255)	Slug untuk URL SEO friendly
penulis	VARCHAR(255)	Nama penulis komik
penerbit	VARCHAR(255)	Nama penerbit komik
sampul	VARCHAR(255)	Nama file gambar sampul komik (path/filename)
created_at	TIMESTAMP	Timestamp saat record dibuat
updated_at	TIMESTAMP	Timestamp saat record terakhir diupdate

1. **id:** Kunci utama (Primary Key) dari tabel komik, yang secara otomatis bertambah nilainya (auto increment).
2. **judul:** Kolom untuk menyimpan judul komik, dengan tipe data VARCHAR(255) untuk menampung teks dengan panjang maksimal 255 karakter.
3. **slug:** Digunakan untuk menyimpan versi URL ramah SEO dari judul komik, dengan tipe data VARCHAR(255). Slug biasanya digunakan untuk mengonversi judul ke dalam format URL-friendly (misalnya, mengganti spasi dengan dash dan karakter khusus dihilangkan).
4. **penulis:** Kolom untuk menyimpan nama penulis komik, dengan tipe data VARCHAR(100).
5. **penerbit:** Kolom untuk menyimpan nama penerbit komik, dengan tipe data VARCHAR(100).
6. **sampul:** Kolom yang menyimpan nama file gambar sampul komik. Dalam aplikasi nyata, biasanya ini akan berupa path atau nama file di server tempat gambar sampul disimpan.
7. **created_at:** Kolom TIMESTAMP untuk mencatat waktu saat record komik ditambahkan ke database.
8. **updated_at:** Kolom TIMESTAMP untuk mencatat waktu saat record komik terakhir diupdate.

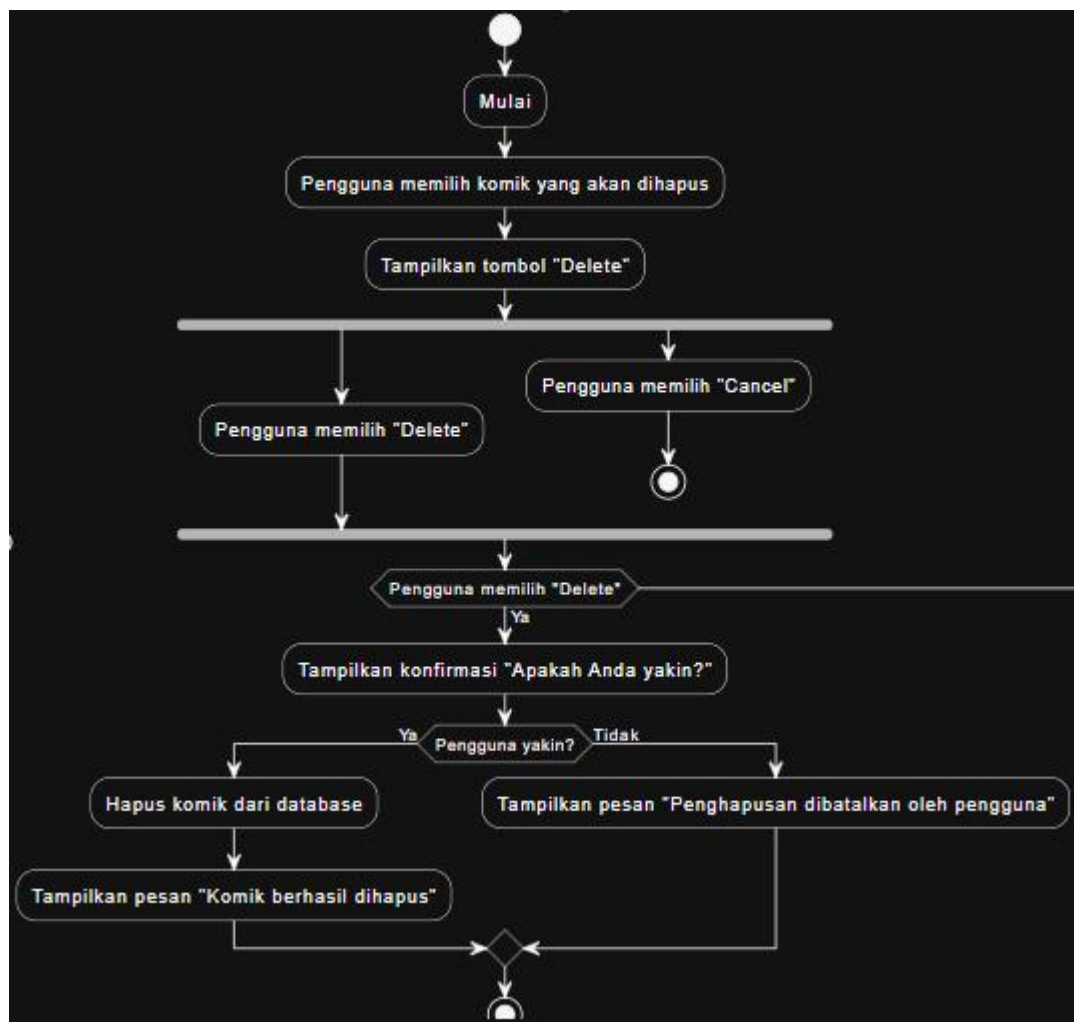
Activity Diagram Input Komik



Activity Diagram Edit



Activity Diagram delete



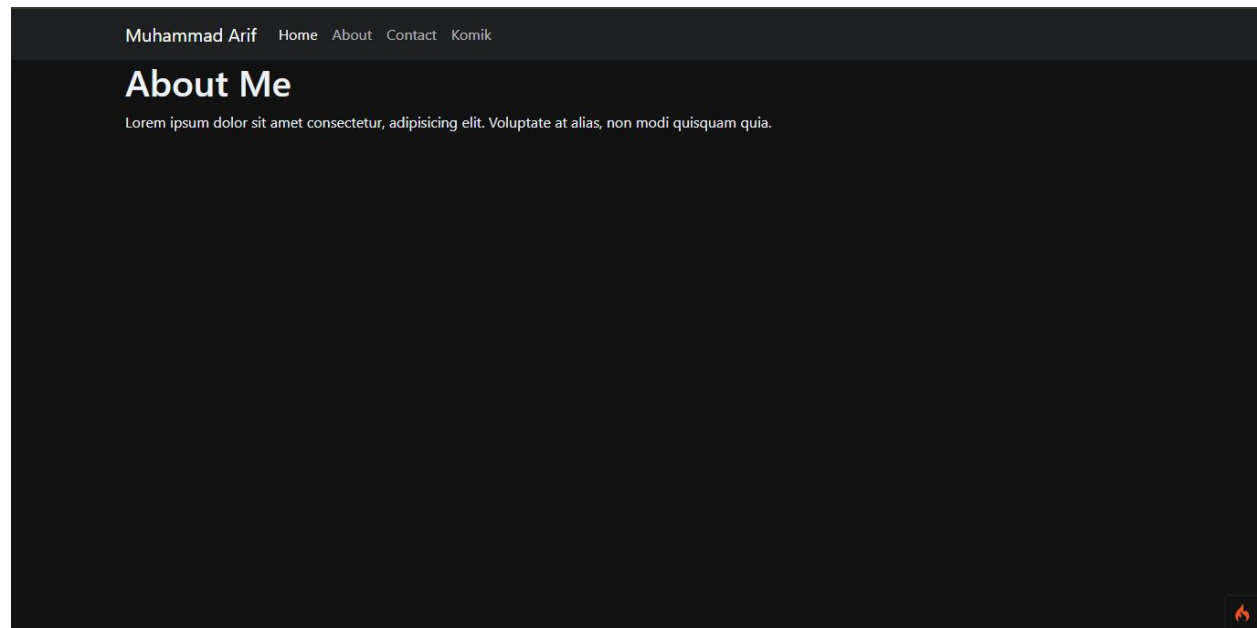
a. UI

Navbar Home

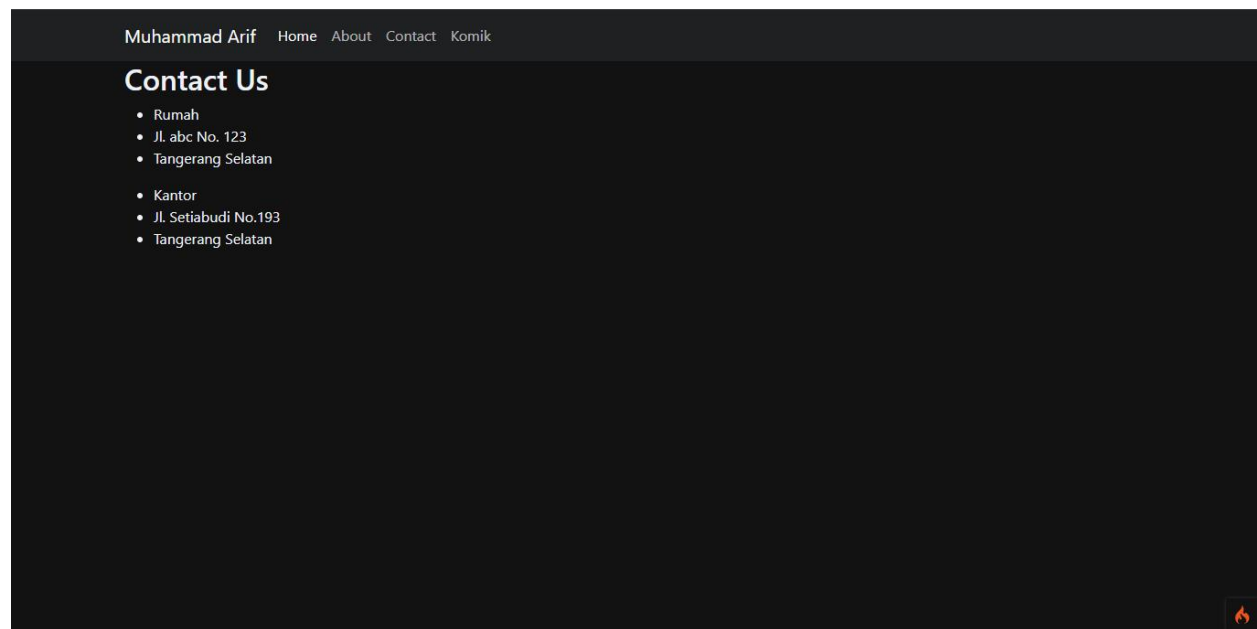
"Temukan cerita-cerita epik dan karakter-karakter tak terlupakan dalam koleksi komik kami."



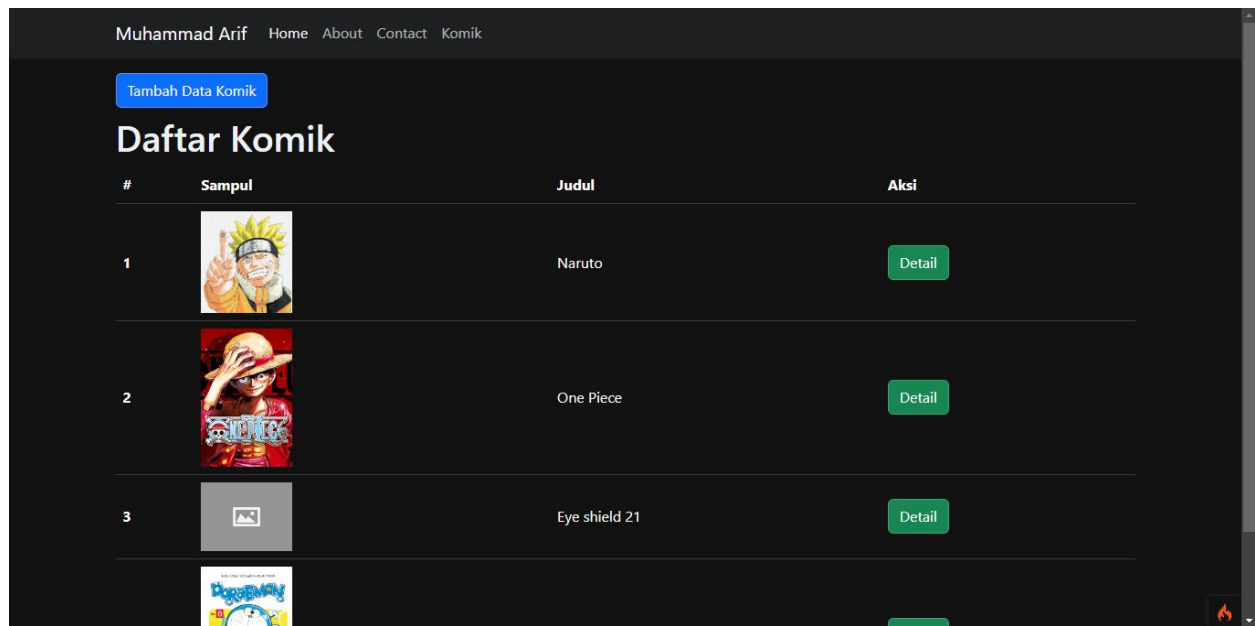
Navbar About me



Navbar Contact



Navbar Komik



b. Menerapkan coding guidlines sesuai dengan bahasa pemrograman yang digunakan

ya, saya menerapkan guideline penulisan sesuai dengan bahasa pemrograman yang di gunakan. Bahasa pemrograman yang saya gunakan ada PHP dengan framework CodeIgniter 4 dan style BootStrap v5

c. Interface input

Tampilan form input data komik

[Muhammad Arif](#) [Home](#) [About](#) [Contact](#) [Komik](#)

Form Tambah Data Komik


Form Judul, Penulis, Penerbit wajib terisi dan image harus format jpg, jpeg, png.

Judul

Penulis

Penerbit

Sampul




Pilih File

Tidak ada file yang dipilih

Upload

Tambah Data



Tampilan untuk form input edit komik.

[Muhammad Arif](#) [Home](#) [About](#) [Contact](#) [Komik](#)

Form Ubah Data Komik

Judul

Naruto


Penulis

Sakahayang

Penerbit

Ajinomoto

Sampul




Pilih File

Tid...ilih

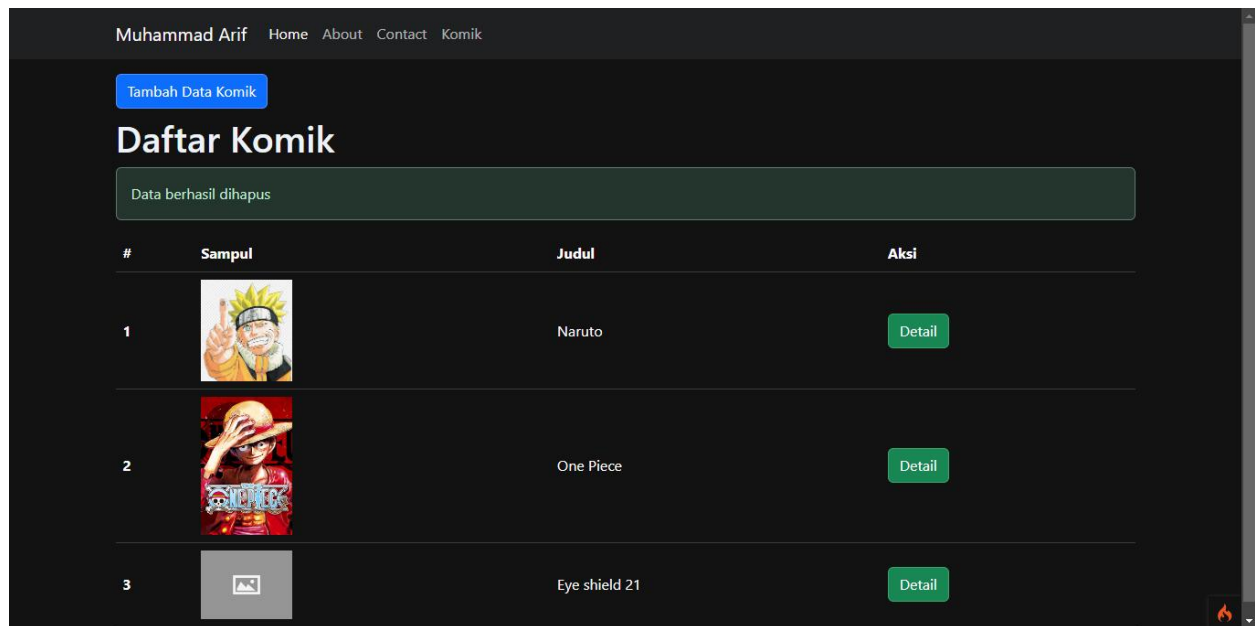
1718235530_a888d6815bb3430bfe64.jpg

Ubah Data

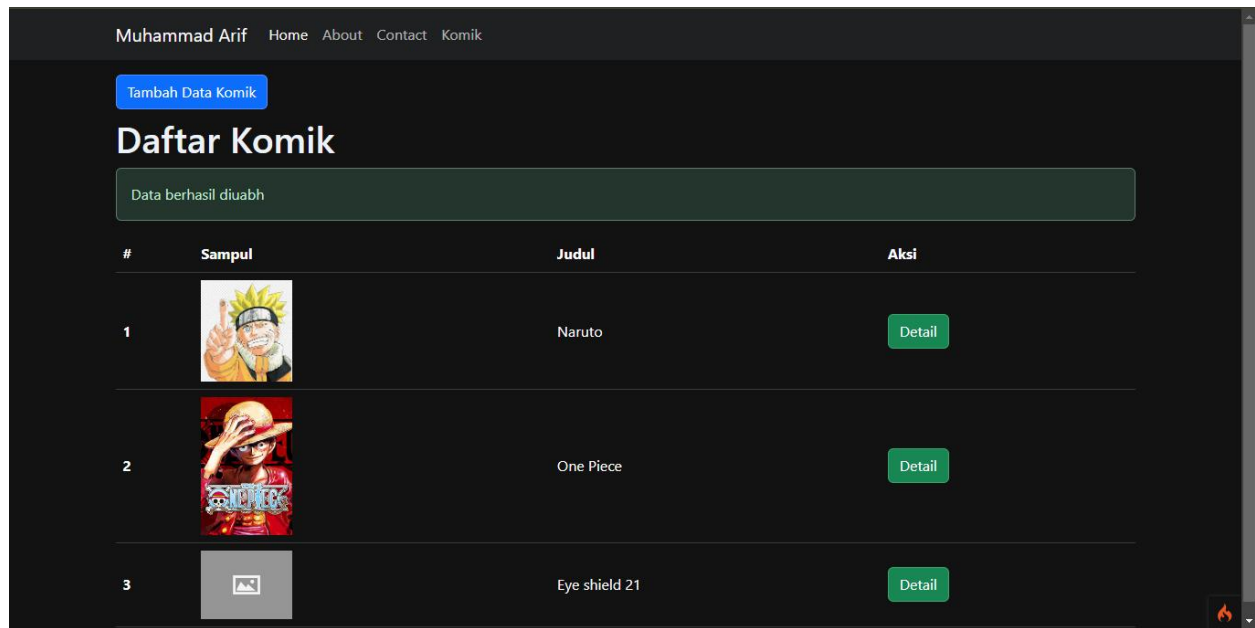


Interface Output

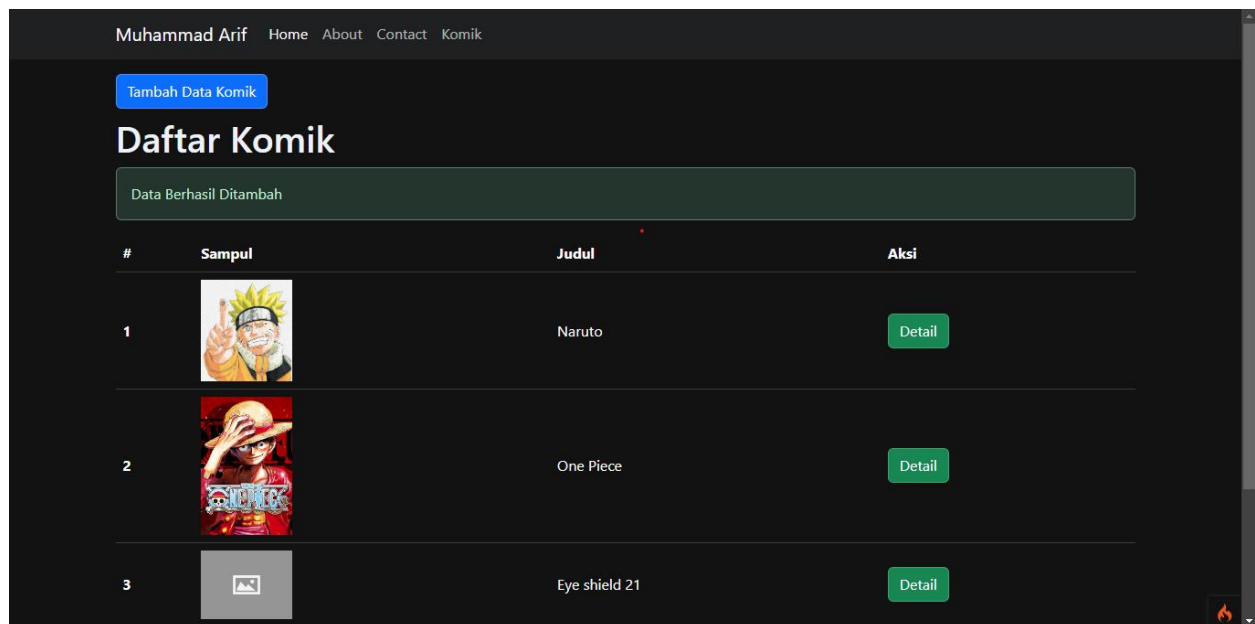
Tampilan pesan data terhapus



Tampilan pesan data diubah



Tampilan pesan data di tambah



d. Program yang dibuat harus menerapkan tipe data yang sesuai, mengikuti syntax Bahasa pemrograman yang digunakan, dan mempunyai struktur control percabangan

(if..then..else) dan pengulangan (do while....for...dll)

Percabangan if then else

Melakukan percabangan kondisi untuk validasi mengunggah gambar dan melakukan generate img foto random yaitu default.jpg Ketika user tidak upload foto.

```
// Validasi form
if (!$this->validate($rules)) {
    $validation = \Config\Services::validation();

    // Cek apakah file yang diupload adalah gambar valid
    if ($fileSampul->isValid() && !in_array($sampul, $validExtension)) {
        session()->setFlashdata('pesan', 'Yang anda upload bukan gambar');
        return redirect()->to(base_url() . 'komik/create')->withInput()->with('validation', $validation);
    }

    return redirect()->to(base_url() . 'komik/create')->withInput()->with('validation', $validation);
} else {
    // Cek apakah file yang diupload adalah gambar valid
    if ($fileSampul->isValid() && !in_array($sampul, $validExtension)) {
        session()->setFlashdata('pesan', 'Yang anda upload bukan gambar');
        return redirect()->to(base_url() . 'komik/create')->withInput();
    }
}

// Apakah tidak ada gambar yang diupload
if ($fileSampul->getError() == 4) {
    $namaSampul = 'default.jpg';
} else {
    // Generate nama sampul random
    $namaSampul = $fileSampul->getRandomName();
    // Pindahkan file ke folder img
    $fileSampul->move('img', $namaSampul);
}
```

Ini adalah proses validasi data yang dikirimkan dari form menggunakan aturan yang didefinisikan sebelumnya dalam variabel \$rules.

Jika validasi gagal (!\$this->validate(\$rules)), maka akan mengarahkan kembali ke halaman form komik/create dengan input yang sudah dimasukkan sebelumnya (withInput()) dan menampilkan pesan validasi yang terkait.

Jika validasi berhasil, dilakukan pengecekan tambahan apakah file yang diupload adalah gambar valid sesuai dengan ekstensi yang diizinkan (\$validExtension). Jika tidak valid, maka akan menampilkan pesan kesalahan yang sesuai dan mengarahkan kembali ke halaman form.

```

// Definisi aturan validasi
$rules = [
    ..... 'judul' => [
    ..... 'rules' => 'required|is_unique[komik.judul]',
    ..... 'errors' => [
    ..... 'required' => '{field} belum diisi',
    ..... 'is_unique' => '{field} sudah terdaftar'
    ..... ]
    ..... ],
    ..... 'penulis' => [
    ..... 'rules' => 'required',
    ..... 'errors' => [
    ..... 'required' => '{field} belum diisi'
    ..... ]
    ..... ],
    ..... 'penerbit' => [
    ..... 'rules' => 'required',
    ..... 'errors' => [
    ..... 'required' => '{field} belum diisi'
    ..... ]
    ..... ],
    ..... 'sampul' => [
    ..... 'rules' => 'max_size[sampul,1024]|is_image[sampul]|mime_in[sampul,image/jpg,image/jpeg,image/png]',
    ..... 'errors' => [
    ..... 'max_size' => 'Ukuran gambar terlalu besar',
    ..... 'is_image' => 'Yang anda pilih bukan gambar',
    ..... 'mime_in' => 'Yang anda pilih bukan gambar'
    ..... ]
    ..... ]
];

```

e. Program yang dibuat harus menerapkan penggunaan prosedur, fungsi, atau method

```

1  <?php
2
3  namespace App\Controllers;
4
5  use App\Models\KomikModel;
6
7  class Komik extends BaseController
8  {
9      protected $komikModel;
10     public function __construct()
11     {
12
13         $this->komikModel = new KomikModel();
14     }
15
16     public function index()
17     {
18         $data = [
19             'title' => 'Daftar Komik',
20             'komik' => $this->komikModel->getKomik()
21         ];
22
23         return view('komik/index', $data);
24     }
25

```

Namespace (namespace App\Controllers;): Digunakan untuk menentukan lokasi dari file class saat ini. Dalam hal ini, controller Komik berada di namespace App\Controllers.

Penggunaan Class (use App\Models\KomikModel;): Mendeklarasikan penggunaan class KomikModel yang berada di namespace App\Models. Ini memungkinkan kita untuk menggunakan KomikModel di dalam controller Komik tanpa harus menuliskan namespace lengkap setiap kali menggunakannya.

Constructor (public function __construct() { ... }): Metode khusus dalam PHP yang dieksekusi secara otomatis ketika sebuah objek dari class dibuat. Pada constructor ini, kita melakukan inisialisasi objek dari class KomikModel dan menyimpannya ke dalam properti \$komikModel yang merupakan bagian dari controller Komik.

Method Index (public function index() { ... }): Method ini menangani permintaan untuk halaman indeks (daftar komik). Ketika dipanggil, method ini melakukan hal berikut:

Membuat sebuah array \$data yang berisi:

'title' => 'Daftar Komik': Sebuah string yang akan digunakan sebagai judul halaman.

'komik' => \$this->komikModel->getKomik(): Memanggil method getKomik() dari objek \$komikModel untuk mengambil daftar komik dari database atau sumber data lainnya.

Memanggil fungsi view() untuk memuat tampilan halaman dengan nama 'komik/index' (artinya file index.php di folder komik dalam folder Views), dan menyertakan data \$data ke dalamnya.

```
public function detail($slug)
{
    $data = [
        'title' => 'Detail Komik',
        'komik' => $this->komikModel->getKomik($slug)
    ];

    // jika komik tidak ada di tabel
    if (empty($data['komik'])) {
        throw new \CodeIgniter\Exceptions\PageNotFoundException('Judul komik ' . $slug . ' tidak ditemukan.');
```

```
    }

    return view('komik/detail', $data);
}

public function create()
{
    $data = [
        'title' => 'Form Tambah Data Komik',
        'validation' => \Config\Services::validation()
    ];

    return view('komik/create', $data);
}
```

Method detail(\$slug): Digunakan untuk menampilkan detail komik berdasarkan slug yang diberikan. Jika komik tidak ditemukan, akan melemparkan pengecualian NotFoundException.

Method create(): Menampilkan form untuk menambahkan data komik baru. Ini termasuk menyediakan objek validasi untuk memvalidasi input form.

Kedua method ini merupakan bagian dari controller Komik yang bertanggung jawab untuk menangani tampilan dan logika terkait dengan komik dalam aplikasi web yang menggunakan framework CodeIgniter 4.

```
public function save()
{
    $validExtension = ['jpg', 'jpeg', 'png'];

    // Mendapatkan ekstensi file yang diupload
    $fileSampul = $this->request->getFile('sampul');
    $sampul = $fileSampul->getClientExtension();

    // Definisi aturan validasi
    $rules = [
        'judul' => [
            'rules' => 'required|is_unique[komik.judul]',
            'errors' => [
                'required' => '{field} belum diisi',
                'is_unique' => '{field} sudah terdaftar'
            ]
        ],
        'penulis' => [
            'rules' => 'required',
            'errors' => [
                'required' => '{field} belum diisi'
            ]
        ],
        'penerbit' => [
            'rules' => 'required',
            'errors' => [
                'required' => '{field} belum diisi'
            ]
        ],
        'sampul' => [
```



```

// Validasi form
if (!$this->validate($rules)) {
    $validation = \Config\Services::validation();

    // Cek apakah file yang diupload adalah gambar valid
    if ($fileSampul->isValid() && !in_array($sampul, $validExtension)) {
        session()->setFlashdata('pesan', 'Yang anda upload bukan gambar');
        return redirect()->to(base_url() . 'komik/create')->withInput()->with('validation', $validation);
    }

    return redirect()->to(base_url() . 'komik/create')->withInput()->with('validation', $validation);
} else {
    // Cek apakah file yang diupload adalah gambar valid
    if ($fileSampul->isValid() && !in_array($sampul, $validExtension)) {
        session()->setFlashdata('pesan', 'Yang anda upload bukan gambar');
        return redirect()->to(base_url() . 'komik/create')->withInput();
    }
}

// Apakah tidak ada gambar yang diupload
if ($fileSampul->getError() == 4) {
    $namaSampul = 'default.jpg';
} else {
    // Generate nama sampul random
    $namaSampul = $fileSampul->getRandomName();
    // Pindahkan file ke folder img
    $fileSampul->move('img', $namaSampul);
}

```

Method `save()` digunakan untuk mengelola proses penyimpanan data komik ke dalam database. Ini melibatkan validasi input form, penanganan file gambar yang diupload, pembuatan slug, penyimpanan data ke database melalui model, dan pengaturan pesan flash untuk memberi tahu pengguna bahwa data telah berhasil disimpan.

```

public function delete($id)
{
    // cari gambar berdasarkan id
    $komik = $this->komikModel->find($id);

    // cek jika file gambarnya default.jpg
    if ($komik['sampul'] != 'default.jpg') {

        // hapus gambar
        unlink('img/' . $komik['sampul']);
    }

    $this->komikModel->delete($id);
    session()->setFlashdata('pesan', 'Data berhasil dihapus');
    return redirect()->to('/komik');
}

public function edit($slug)
{
    $data = [
        'title' => 'Form ubah Data Komik',
        'validation' => \Config\Services::validation(),
        'komik' => $this->komikModel->getKomik($slug)
    ];

    return view('komik/edit', $data);
}

```

Metode delete(\$id): Digunakan untuk menghapus data komik berdasarkan \$id, termasuk penghapusan gambar sampul jika tidak berupa default.jpg, dan memberikan feedback kepada pengguna setelah penghapusan berhasil.

Metode edit(\$slug): Digunakan untuk menampilkan form pengeditan data komik berdasarkan \$slug, dimana data komik yang akan diedit diambil dari database dan ditampilkan dalam form untuk diedit.

```

public function update($id)
{
    // cek judul
    $komikLama = $this->komikModel->getKomik($this->request->getVar('slug'));
    if ($komikLama['judul'] == $this->request->getVar('judul')) {
        $rule_judul = 'required';
    } else {
        $rule_judul = 'required|is_unique[komik.judul]';
    }

    if (!$this->validate([
        'judul' => [
            'rules' => $rule_judul,
            'errors' => [
                'required' => '{field} komik harus diisi.',
                'is_unique' => '{field} komik sudah terdaftar.'
            ]
        ],
        'sampul' => [
            'rules' => 'max_size[sampul,1024]|is_image[sampul]|mime_in[sampul,image/jpg,image/jpeg,image/png]',
            'errors' => [
                'max_size' => 'Ukuran gambar terlalu besar',
                'is_image' => 'Yang anda pilih bukan gambar',
                'mime_in' => 'Yang anda pilih bukan gambar'
            ]
        ]
    ])) {
        // $validation = \Config\Services::validation();
        return redirect()->to('/komik/edit/' . $this->request->getVar('slug'))->withInput();
    }

    $fileSampul = $this->request->getFile('sampul');

    // cek gambar, apakah tetap gambar lama
    if ($fileSampul->getError() == 4) {
        $namaSampul = $this->request->getVar('sampulLama');
    } else {
        // generate nama file random
        $namaSampul = $fileSampul->getRandomName();
        // pindahkan gambar
        $fileSampul->move('img', $namaSampul);
        // hapus file yang lama
        unlink('img/' . $this->request->getVar('sampulLama'));
    }

    $slug = url_title($this->request->getVar('judul'), '-', true);

    $this->komikModel->save([
        'id' => $id,
        'judul' => $this->request->getVar('judul'),
        'slug' => $slug,
        'penulis' => $this->request->getVar('penulis'),
        'penerbit' => $this->request->getVar('penerbit'),
        'sampul' => $namaSampul
    ]);

    session()->setFlashdata('pesan', 'Data berhasil diuab');

    return redirect()->to('/komik');
}

```

Metode update(\$id) ini memungkinkan pengguna untuk menyunting data komik yang sudah ada dalam aplikasi. Prosesnya mencakup validasi input, pengecekan dan manajemen gambar, serta penyimpanan data ke dalam database. Setelah berhasil disimpan, pengguna diberikan informasi melalui pesan flash dan diarahkan kembali ke halaman utama komik.\

f. Program yang dibuat harus menggunakan Array

Berikut adalah array pada navbar

```
public function contact()
{
    $data = [
        'title' => 'Contact Us',
        'alamat' => [
            [
                'tipe' => 'Rumah',
                'alamat' => 'Jl. abc No. 123',
                'kota' => 'Tangerang Selatan'
            ],
            [
                'tipe' => 'Kantor',
                'alamat' => 'Jl. Setiabudi No.193',
                'kota' => 'Tangerang Selatan'
            ]
        ]
    ];
}
```

```

$rules = [
    'judul' => [
        'rules' => 'required|is_unique[komik.judul]',
        'errors' => [
            'required' => '{field} belum diisi',
            'is_unique' => '{field} sudah terdaftar'
        ]
    ],
    'penulis' => [
        'rules' => 'required',
        'errors' => [
            'required' => '{field} belum diisi'
        ]
    ],
    'penerbit' => [
        'rules' => 'required',
        'errors' => [
            'required' => '{field} belum diisi'
        ]
    ],
    'sampul' => [
        'rules' => 'max_size[sampul,1024]|is_image[sampul]|mime_in[sampul,image/jpg,image/jpeg,image/png]',
        'errors' => [
            'max_size' => 'Ukuran gambar terlalu besar',
            'is_image' => 'Yang anda pilih bukan gambar',
            'mime_in' => 'Yang anda pilih bukan gambar'
        ]
    ]
];

// Validasi form

```

Array adalah struktur data yang digunakan untuk menyimpan kumpulan nilai atau elemen, yang disusun dalam urutan tertentu dan diakses melalui indeks. Setiap nilai dalam array disimpan di bawah satu variabel tunggal, dan setiap nilai memiliki posisi numerik yang disebut indeks. Indeks digunakan untuk mengakses dan memanipulasi nilai dalam array.

g. Program yang dibuat harus mempunyai fasilitas untuk menyimpan dan membaca data di media penyimpanan

penyimpan Input menyimpan data dan membaca :

```

public function create()
{
    $data = [
        'title' => 'Form Tambah Data Komik',
        'validation' => \Config\Services::validation()
    ];

    return view('komik/create', $data);
}

```

```

public function save()
{
    $validExtension = ['jpg', 'jpeg', 'png'];

    // Mendapatkan ekstensi file yang diupload
    $fileSampul = $this->request->getFile('sampul');
    $sampul = $fileSampul->getClientExtension();

    // Definisi aturan validasi
    $rules = [
        'judul' => [
            'rules' => 'required|is_unique[komik.judul]',
            'errors' => [
                'required' => '{field} belum diisi',
                'is_unique' => '{field} sudah terdaftar'
            ]
        ],
        'penulis' => [
            'rules' => 'required',

```

```

public function save()
    ],
    'penulis' => [
        'rules' => 'required',
        'errors' => [
            'required' => '{field} belum diisi'
        ]
    ],
    'penerbit' => [
        'rules' => 'required',
        'errors' => [
            'required' => '{field} belum diisi'
        ]
    ],
    'sampul' => [
        'rules' => 'max_size[sampul,1024]|is_image[sampul]|mime_in[sampul,image/jpg,image/jpeg,image/png]',
        'errors' => [
            'max_size' => 'Ukuran gambar terlalu besar',
            'is_image' => 'Yang anda pilih bukan gambar',
            'mime_in' => 'Yang anda pilih bukan gambar'
        ]
    ]
];

// Validasi form
if (!$this->validate($rules)) {
    $validation = \Config\Services::validation();

    // Cek apakah file yang diupload adalah gambar valid
    if ($fileSampul->isValid() && !in_array($sampul, $validExtension)) {
        session()->setFlashdata('pesan', 'Yang anda upload bukan gambar');
        return redirect()->to(base_url() . 'komik/create')->withInput()->with('validation', $validation);
    }
}

```



```

        return redirect()->to(base_url() . 'komik/create')->withInput()->with('validation', $validation);
    } else {
        // Cek apakah file yang diupload adalah gambar valid
        if ($fileSampul->isValid() && !in_array($sampul, $validExtension)) {
            session()->setFlashdata('pesan', 'Yang anda upload bukan gambar');
            return redirect()->to(base_url() . 'komik/create')->withInput();
        }
    }

    // Apakah tidak ada gambar yang diupload
    if ($fileSampul->getError() == 4) {
        $namaSampul = 'default.jpg';
    } else {
        // Generate nama sampul random
        $namaSampul = $fileSampul->getRandomName();
        // Pindahkan file ke folder img
        $fileSampul->move('img', $namaSampul);
    }

    $slug = url_title($this->request->getVar('judul'), '-', true);
    $this->komikModel->save([
        'judul' => $this->request->getPost('judul'),
        'slug' => $slug,
        'penulis' => $this->request->getPost('penulis'),
        'penerbit' => $this->request->getPost('penerbit'),
        'sampul' => $namaSampul
    ]);

    session()->setFlashdata('pesan', 'Data Berhasil Ditambah');
    return redirect()->to(base_url() . 'komik');
}





```

Output menyimpan dan membaca data

Muhammad Arif
Home
About
Contact
Komik

Tambah Data Komik

Daftar Komik

#	Sampul	Judul	Aksi
1		Naruto	Detail
2		One Piece	Detail
3		Eye shield 21	Detail
			Detail

h. Program harus menerapkan hak akses tipe data dengan benar, mempunyai properties, menerapkan inheritance, polymorphy, overloading, dan interface.

```
class Komik extends BaseController
{
    protected $komikModel;
    public function __construct()
    {
        $this->komikModel = new KomikModel();
    }
}
```

Hak Akses Tipe Data: Properti \$komikModel di-declare sebagai protected, yang berarti hanya dapat diakses dari dalam kelas Komik dan kelas turunannya.

Kelas Komik meng-extend BaseController. Ini menunjukkan bahwa Komik mewarisi semua properti dan metode dari BaseController dan dapat menambahkan atau memodifikasi perilaku sesuai kebutuhan.

```
<?php

namespace App\Models;

use CodeIgniter\Model;

class KomikModel extends Model
{
    protected $table = 'komik';
    protected $useTimestamps = true;
    protected $allowedFields = ['judul', 'slug', 'penulis', 'penerbit', 'sampul'];

    public function getKomik($slug = false)
    {
        if ($slug == false) {
            return $this->findAll();
        }

        return $this->where(['slug' => $slug])->first();
    }
}
```


\$table: Properti ini menentukan tabel database yang akan digunakan oleh model ini, dalam hal ini adalah tabel komik.

\$useTimestamps: Properti ini diatur ke true yang berarti CodeIgniter akan secara otomatis mengelola kolom `created_at` dan `updated_at` di tabel komik, jika kolom-kolom ini ada.

\$allowedFields: Properti ini adalah array yang menentukan kolom-kolom mana saja yang diizinkan untuk diisi atau diperbarui melalui metode `insert`, `update`, atau `save` dari model ini. Kolom yang diizinkan adalah judul, slug, penulis, penerbit, dan sampul.

Parameter \$slug: Parameter ini bersifat opsional dan default-nya adalah false. Jika tidak ada nilai yang diberikan untuk parameter ini, maka metode ini akan mengembalikan semua data dari tabel komik.

\$this->findAll(): Jika \$slug adalah false, metode ini akan mengembalikan semua baris dari tabel komik.

\$this->where(['slug' => \$slug])->first(): Jika \$slug diberikan, metode ini akan mencari baris di tabel komik di mana kolom slug cocok dengan nilai \$slug dan mengembalikan baris pertama yang ditemukan.

i. Program yang dibuat harus terdiri dari 2 atau lebih namespace atau package

```
<?php

namespace App\Models;

use CodeIgniter\Model;

class KomikModel extends Model
{
    protected $table = 'komik';
    protected $useTimestamps = true;
    protected $allowedFields = ['judul', 'slug', 'penulis', 'penerbit', 'sampul'];

    public function getKomik($slug = false)
    {
        if ($slug == false) {
            return $this->findAll();
        }

        return $this->where(['slug' => $slug])->first();
    }
}
```

Menampil namespace `App\Models`; untuk class `KomikModel`.

```

1  <?php
2
3  namespace App\Controllers;
4
5  use App\Models\KomikModel;
6
7  class Komik extends BaseController
8  {
9      protected $komikModel;
10     public function __construct()
11     {
12
13         $this->komikModel = new KomikModel();
14     }
15
16     public function index()
17     {
18         $data = [
19             'title' => 'Daftar Komik',
20             'komik' => $this->komikModel->getKomik()
21         ];
22
23         return view('komik/index', $data);
24     }
25

```

Namespace App\Controllers; pada class komik.

j. Program yang dibuat harus menggunakan atau memanfaatkan eksternal library yang sudah ada dan tersedia

```

PS C:\xampp\htdocs\ci4app> composer require filp/whoops
./composer.json has been updated
Running composer update filp/whoops
Loading composer repositories with package information
Updating dependencies
Nothing to modify in lock file
Installing dependencies from lock file (including require-dev)
Nothing to install, update or remove
Generating optimized autoload files
27 packages you are using are looking for funding.
Use the `composer fund` command to find out more!
No security vulnerability advisories found
Using version ^2.15 for filp/whoops
PS C:\xampp\htdocs\ci4app>

```

saya memanfaatkan library debugger eksternal yaitu whoops, Whoops adalah library PHP yang menangani kesalahan dengan cara yang lebih ramah pengguna.

k. Program harus menggunakan basis data

Database yang saya gunakan ada mysql berikut adalah databse saya dan konektor database pada file .env di CI 4.

```

database.default.hostname = localhost
database.default.database = ci4
database.default.username = root
database.default.password =
database.default.DBDriver = MySQLi
#database.default.DBPrefix =
#database.default.port = 3306

```

phpMyAdmin

Terbaru

Favorit

Baru

althea

arif_tgrp

ci4

Baru

komik

indogreat

information_schema

jack

lsrp

mysql

performance_schema

phpmyadmin

roleplay3

s607057_my-db1643623086

test

Server: 127.0.0.1 > Database: ci4 > Tabel: komik

Jelajahi

Struktur

SQL

Cari

Tambahkan

Ekspor

Impor

Hak Akses

Operasi

Pelacakan

Trigger

Menampilkan baris 0 - 3 (total 4, Pencarian dilakukan dalam 0,0004 detik.)

SELECT * FROM `komik`

Profil

[Edit kotak]

[Ubah]

[Jelaskan SQL]

[Buat kode PHP]

[Segarkan]

Tampilkan semua

Jumlah baris: 25

Saring baris: Cari di tabel ini

Sort by key: Tidak ada

Extra options

			Id	Judul	slug	penulis	penarbit	sampul	created_at	updated_at	
<input type="checkbox"/>	Ubah	Salin	Hapus	1	Naruto	naruto	Sakahayang	Ajinomoto	1718235530_a888d6815bb3430bfe64.jpg	2024-06-12 23:38:20	2024-06-21 15:19:16
<input type="checkbox"/>	Ubah	Salin	Hapus	6	One Piece	one-piece	Sakahayang	Gamedia	1719013858_351e4b8affb362d5d427.jpg	2024-06-21 23:50:58	2024-06-21 23:50:58
<input type="checkbox"/>	Ubah	Salin	Hapus	7	Eye shield 21	eye-shield-21	Jumbotron	Gamedia	default.jpg	2024-06-21 23:51:28	2024-06-22 04:52:41
<input type="checkbox"/>	Ubah	Salin	Hapus	9	Doraemon	doraemon	Sakahayang	Gamedia	1719031998_8e8aae92688e5ad87b08.jpg	2024-06-22 04:53:18	2024-06-22 04:53:18

Pilih Semua

Dengan pilihan: Ubah

Salin

Hapus

Ekspor

Tampilkan semua

Jumlah baris: 25

Saring baris: Cari di tabel ini

Sort by key: Tidak ada

Operasi hasil kueri

Cetak

Salin ke clipboard

Ekspor

Tampilkan bagan

Buat tampilan

Markah kueri SQL ini