**DOKUMENTASI**

**TUGAS EAS PEMROGRAMAN INTEGRATIF**

**PENGAPLIKASIAN ARSITEKTUR MVC**

****

Penyusun :

**I Gede Pradhana Indra Widnyana**

**05311840000031**

**DEPARTEMEN TEKNOLOGI INFORMASI**

**INSTITUT TEKNOLOGI SEPULUH NOPEMBER SURABAYA**

Awalnya saya ingin menjelaskan konsep yang ingin saya terapkan sesuai dengan soal yang diberikan oleh bapak. Penjelasannya terkait penggunaan nama file, database (table dalam database) yang saya gunakan adalah sebagai berikut :

1. Karena pada soal terdapat bapak (a) dan ibu (b) , maka terdapat tabel user, yang terdapat ID(incase terdapat nama yang sama). Karena soal menyertakan “bapak” dan ”ibu”, maka perlu dibuat adanya gender sebagai keterangan tambahan pembeda.

**Tabel. User**

|  |  |  |
| --- | --- | --- |
| **id (INT(50),A\_I,Pkey,unique)** | **name (VARCHAR(255))** | **gender (tinyint(1))** |
| Ket: |  |  |

* **id** selalu angka (INTEGER) sepanjang 50, digunakan sebanyak itu sebagai antisipasi jika user banyak yang masuk ke database banyak dan AutoIncrement karena **id** adalah Primary key. Harus unique.
* **name** untuk nama user, yang mana berupa char (character), maka saya menggunakan varchardengan maksimal 255 character sebagai antisipasi.
* **Gender** dibuat sebagai boolean, dalam mysql, tinyint(1) sama seperti boolean dimana dalam kasus ini **0 adalah laki laki**, dan **1 adalah perempuan**.
* Karena terdapat multiple sumbangan dalam 1 user (lebih dari satu sumbangan) , maka kita harus membagi tabel sesuai relational database.

**Tabel.sumbangan**

**id (INT(50),A\_I,Pkey,unique) Userid(INT(50),Fkey) Jenis(int (10),Fkey) Jumlah(int(50))**

Ket:

* **Id** untuk mengidentifikasi item agar ada yang menjadi pembeda antara sumbangan yang berulang.
* **Userid** adalah foreign key dari user.id agar tahu kepemilikan sumbangan.
* **Jenis** berisi foreign key id dari jenis\_sumbangan.
* **Jumlah** adalah jumlah dari sumbangan
* Terakhir adalah tabel jenis sumbangan

**Tabel.jenis\_sumbangan**

**id (INT(50),A\_I,Pkey,unique) Name(varchar(255))**

Ket:

* **Id** untuk identifikasi item sumbangan
* **Name** untuk nama jenis sumbangan(beras,uang,obat,dll)

1. View yang dibutuhkan dalam kasus ini agar sesuai soal dan dan dapat diimplementasikan adalah **Form view** untuk mendaftarkan sumbangan dan **view rekap**
2. Controller yang dibutuhkan adalah controller **jenis.php** (agar mudah dipahami) untuk setiap page view rekap dan **home.php** untuk menampilkan view dari form sumbangan
3. Model yang dibutuhkan adalah model **sumbangan.php** (agar mudah dipahami) yang berisi **getFilter(data)** data berupa jenis sumbangan, lalu ada **getSumbangan()** untuk mengambil semua data sumbangan, dan ada **setSumbangan(data)** untuk memasukkan sumbangan ke database. Jumlah data tersebut bisa tidak hanyasatu (lebih dari satu), Namun pasti ada tidak mungkin tidak ada data.

**Pengaplikasian :**

* Untuk pengaplikasian saya menggunakan folder yang sama dengan tugas-tugas sebelumnya yakni foler hellomvc , namun saya tambahkan beberapa file sesuai dengan ketentuan yang saya buat diatas, seperti **sumbangan.php, rekap.html, jenis.php,** mengedit (modifikasi) dengan menambahkan isi dari **Base.html dan index.html,** serta dengan memodifikasi dengan menambahkan route pada **index.php**
* **Berikut ini beberapa code dan isi dari file-file tersebut :**

1. Dibawah ini adalah code dari file **Home.php** yang berisi tentang form awal pada website COVID-19 saya. **(template warna dan form menggunakan bootstrap)**

<?php

namespace App\Controllers;

use Core\View;

use app\models\sumbangan;

class Home extends \Core\Controller

{

    public function index()

    {

        $hasil = sumbangan::getName();

        View::renderTemplate("home/index.html", [

            'judul' => 'Donasi COVID-19',

            'jenis' => $hasil

        ]);

    }

    public function inputdata()

    {

        $pesan = "";

        $jsumbang = $\_POST['jenisSumbangan'];

        $jumlah = $\_POST['jumlah'];

        if(!isset($\_POST['submit'])) return;

        foreach($jumlah as $jm){

            if($jm == "" || $jm == 0 ) $pesan = "Tolong isi Jumlah Sumbangan anda";

        }

        foreach($jsumbang as $js){

            if($js == "") $pesan = "Tolong Jenis Sumbangan anda";

        }

        if($\_POST['name'] == "") $pesan = "Tolong isi nama anda";

        if($pesan != ""){

            $hasil = sumbangan::getName();

            View::renderTemplate("home/index.html", [

            'judul' => 'Sumbangan COVID-19',

            'jenis' => $hasil,

            'alert' => $pesan

        ]);

        return;

        }

        if(!isset($\_POST['submit'])) return;

        $userid = sumbangan::setUser( $\_POST['name'], $\_POST['gender'] );

        $iter = 0;

        foreach($jsumbang as $js){

            $jsid = sumbangan::isThere($js);

            if( $jsid >= 1){

                $done = sumbangan::setSumbangan($userid,$jsid,$jumlah[$iter]);

            }

            else{

                $jsid = sumbangan::setJS($js);

                $done = sumbangan::setSumbangan($userid, $jsid[0], $jumlah[$iter]);

            }

            $iter++;

        }

        $hasil = sumbangan::getName();

        View::renderTemplate("home/index.html", [

            'judul' => 'Donasi COVID-19',

            'jenis' => $hasil,

            'pesan' => 'Terima kasih, Sumbangan anda telah berhasil dimasukkan',

        ]);

    }

}

1. Dibawah ini adalah isi dari file **index.html**yang sebagian besar isinya untuk tampilan dari form dan website itu sendiri baik dari setiap row yang terdapat pada form pengisian sumbangan, dll.

{% extends "base.html" %}

{% block body %}

<div class="container">

    {% if pesan is defined %}

    <div class="alert alert-warning alert-dismissible fade show" role="alert">

        {{ pesan }}

        <button type="button" class="close" data-dismiss="alert" aria-label="Close">

          <span aria-hidden="true">&times;</span>

        </button>

    </div>

    {% endif %}

    {% if alert is defined %}

    <div class="alert alert-danger alert-dismissible fade show" role="alert">

        {{ alert }}

        <button type="button" class="close" data-dismiss="alert" aria-label="Close">

          <span aria-hidden="true">&times;</span>

        </button>

    </div>

    {% endif %}

    <form action="inputdata" method="POST">

        <div class="form-group">

          <label for="name">Nama</label>

          <input type="text" class="form-control" id="name" name="name" placeholder="Contoh : Tejo">

        </div>

        <div class="form-group">

          <label for="gender">Jenis Kelamin</label>

          <select class="form-control" id="gender" name="gender">

            <option value="0">Laki-laki</option>

            <option value="1">Perempuan</option>

          </select>

        </div>

        <div class="form-group field\_wrapper">

            <label for="jenisSumbangan[]">Jenis Sumbangan</label>

            <div class="row">

                <div class="col-11">

                    <input class="form-control" type="text" list="jenisSumbangan[]" name="jenisSumbangan[]" />

                    <datalist id="jenisSumbangan[]">

                        {% for a in jenis %}

                        <option>{{ a.name }}</option>

                        {% endfor %}

                    </datalist>

                    <label for="jumlah[]">Jumlah</label>

                    <input type="number" class="form-control" id="jumlah[]" name="jumlah[]" placeholder="Contoh: 5">

                    <small id="jumlah" class="form-text text-muted">dalam kg/pcs/liter.</small>

                </div>

                <div class="col">

                    <a href="javascript:void(0);" class="add\_button btn btn-primary" title="Add field">+</a>

                </div>

            </div>

          </div>

          <button type="submit" class="btn btn-primary" name="submit" id="submit">Sumbangkan</button>

      </form>

</div>

<script type="text/javascript">

    $(document).ready(function(){

        var maxField = 10; //Input fields increment limitation

        var addButton = $('.add\_button'); //Add button selector

        var wrapper = $('.field\_wrapper'); //Input field wrapper

        var fieldHTML = '<div><label for="jenisSumbangan[]">Jenis Sumbangan</label><div class="row"> '+

                        '    <div class="col-11"> '+

                        '        <input class="form-control" type="text" list="jenisSumbangan" name="jenisSumbangan[]" /> '+

                        '        <datalist id="jenisSumbangan"> '+

                        '            <option>Beras</option> '+

                        '            <option>Minyak Goreng</option> '+

                        '            <option>Mie Instant</option> '+

                        '            <option>Gula</option> '+

                        '        </datalist> '+

                        '        <label for="jumlah">Jumlah</label> '+

                        '        <input type="number" class="form-control" id="jumlah" name="jumlah[]" placeholder="Contoh: 5"> '+

                        '        <small id="jumlah" class="form-text text-muted">dalam kg/pcs/liter.</small> '+

                        '    </div> '+

                        '    <div class="col"> '+

                        '        <a href="javascript:void(0);" class="remove\_button btn btn-danger">-</a> '+

                        '    </div> '+

                        '</div></div>'; //New input field html

        var x = 1; //Initial field counter is 1

        //Once add button is clicked

        $(addButton).click(function(){

            //Check maximum number of input fields

            if(x < maxField){

                x++; //Increment field counter

                $(wrapper).append(fieldHTML); //Add field html

            }

        });

        //Once remove button is clicked

        $(wrapper).on('click', '.remove\_button', function(e){

            e.preventDefault();

            $(this).parent('div').parent('div').parent('div').remove(); //Remove field html

            x--; //Decrement field counter

        });

    });

</script>

{% endblock %}

1. Dibawah ini merupakan code dari file **index.php** yang isinya terkait route dari semua yang ada pada hellomvc saya, termasuk tugas sebelum-sebelumnya. Untuk tugas EAS, saya cantumkan pada line 40-44 bahwa 3 route tersbut adalah route tambahan khusus untuk tugas EAS ini.

<?php

//echo 'Requested URL = "' . $\_SERVER['QUERY\_STRING'] . '"';

// require '../app/controllers/Posts.php';

// require '../core/Router.php';

// $router = new Router();

// //echo get\_class($router);

// Require the controller class

//require '../App/Controllers/Posts.php';

require\_once dirname(\_\_DIR\_\_) .'/vendor/autoload.php';

/\*\*

 \* Autoloader

 \*/

spl\_autoload\_register(function ($class) {

    $root = dirname(\_\_DIR\_\_);   // get the parent directory

    $file = $root . '/' . str\_replace('\\', '/', $class) . '.php';

    if (is\_readable($file)) {

        require $root . '/' . str\_replace('\\', '/', $class) . '.php';

    }

});

$router = new Core\Router();

// Add the routes

$router->add('', ['controller' => 'Home', 'action' => 'index']);

$router->add('posts', ['controller' => 'Posts', 'action' => 'index']);

$router->add('{controller}/{action}');

$router->add('{controller}/{id:\d+}/{action}');

$router->add('lingkaran', ['controller' => 'lingkaran', 'action' => 'index']);

$router->add('hitung', ['controller' => 'lingkaran', 'action' => 'hitung']);

$router->add('tests1/index', ['controller' => 'tests1', 'action' => 'index']);

$router->add('things/index', ['controller' => 'things', 'action' => 'index']);

$router->add('datas/index', ['controller' => 'datas', 'action' => 'index']);

$router->add('foods/index', ['controller' => 'foods', 'action' => 'index']);

//Router tambahan pada tugas EAS HELLOMVC

$router->add('inputdata', ['controller' => 'Home', 'action' => 'inputdata']);

$router->add('Jenis', ['controller' => 'Jenis', 'action' => 'Index']);

$router->add('filter', ['controller' => 'Jenis', 'action' => 'filter']);

$router->dispatch($\_SERVER['QUERY\_STRING']);

// echo '<pre>';

// var\_dump($router->getRoutes());

// echo '</pre>';

// // Match the requested route

// $url = $\_SERVER['QUERY\_STRING'];

// // Display the routing table

// echo '<pre>';

// echo htmlspecialchars(print\_r($router->getRoutes(), true));

// echo '</pre>';

// if ($router->match($url)) {

//     echo '<pre>';

//     var\_dump($router->getParams());

//     echo '</pre>';

// } else {

//     echo "No route found for URL '$url'";

// }

1. Dibawah ini adalah file **jenis.php** yang berfungsi untuk menampilkan semua sumbangan yang ada dengan memanggil fungsi pada file **rekap.html** baik itu untuk menampilkan semuanya ataupun menampilkan fungsi **filtering** saja

<?php

namespace App\Controllers;

use Core\View;

use App\Models\sumbangan;

class Jenis extends \Core\Controller

{

    public function index()

    {

        $hasil = sumbangan::getName();

        $data = sumbangan::getSumbangan();

        View::renderTemplate("home/rekap.html", [

            'judul' => 'List Sumbangan COVID-19',

            'jenis' => $hasil,

            'data' => $data,

            'nav' => FALSE,

            'name'=> 'all'

        ]);

    }

    public function filter()

    {

        if(!isset($\_GET['submit'])) return;

        $hasil = sumbangan::getName();

        $data = sumbangan::getFilterSumbangan($\_GET['submit']);

        View::renderTemplate("home/rekap.html", [

            'judul' => 'List Sumbangan ' . $\_GET['submit'] . ' COVID-19',

            'jenis' => $hasil,

            'data' => $data,

            'nav' => FALSE,

            'name' => $\_GET['submit']

        ]);

    }

}

1. Dibawah ini adalah code dari file **rekap.html** yang mengatur “tampilan” dari fitur list sumbangan yang sudah dibahas sebelumnya, terdapat juga fungsi **filtering** pada file **rekap.html** ini.

{% extends "base.html" %}

{% block body %}

<div class="container-fluid mt-5 mb-5">

    <div class="row">

        <div class="col-4">

            <h2>Jenis sumbangan</h2>

            <div class="list-group">

                <form action="Jenis">

                    <button type="submit" class="list-group-item list-group-item-action {% if name == 'all' %} active bg-dark{% endif %}">

                    Tampilkan semua

                    </button>

                </form>

                <form action="filter" method="GET">

                {% for a in jenis %}

                        <button type="submit" value="{{ a.name }}" name="submit" class="{% if name == a.name %} active bg-secondary {% endif %} list-group-item list-group-item-action">{{ a.name }}</button>

                {% endfor %}

                </form>

              </div>

        </div>

        <div class="col">

            <table id="tablepagination" class="table table-bordered table-hover" style="width:100%">

                <thead>

                  <tr>

                    <th scope="col">No</th>

                    <th scope="col">Nama</th>

                    <th scope="col">Jenis Sumbangan</th>

                    <th scope="col">Jumlah (kg atau pcs)</th>

                  </tr>

                </thead>

                <tbody>

                    {% for c in data %}

                  <tr>

                    <th scope="row">{{ c.nomer }}</th>

                    <td>{% if c.gender == TRUE %}Ibu {% else %}Bapak {% endif %}{{ c.name }}</td>

                    <td>{{ c.jenis }}</td>

                    <td>{{ c.jumlah }}</td>

                  </tr>

                  {% endfor %}

                </tbody>

              </table>

        </div>

    </div>

</div>

<script type="text/javascript">

    $(document).ready(function() {

    $('#tablepagination').DataTable();

} );

</script>

{% endblock %}

1. Berikut ini adalah isi dari file **sumbangan.php** yang isinya tentang public static function (public function) **sejumlah 7 diantaranya getName, setUser, isThere, setJS, setSumbangan, getSumbangan, getFilterSumbangan.** Semua fungsi tersebut digunakan untuk fetch data yang sudah ada dari database ke website **(seperti pada fitur list sumbangan)** dan menginput data dari website ke database.

<?php

namespace App\Models;

use Core\Model;

use PDO;

use PDOException;

class sumbangan extends Model

{

    public static function getName()

    {

        try {

            $db = static::getDB();

            $stmt = $db->query('SELECT `name` FROM `jenis\_sumbangan` ORDER BY `name`');

            $results = $stmt->fetchAll(PDO::FETCH\_ASSOC);

            return $results;

        } catch (PDOException $e) {

            echo $e->getMessage();

        }

    }

    public static function setUser($name, $gender)

    {

        try {

            $db = static::getDB();

            $stmt = $db->query("INSERT INTO `user`(`name`, `gender`) VALUES ('". $name ."', ". $gender .")");

            $results = $stmt->fetchAll(PDO::FETCH\_ASSOC);

            return $db->lastinsertId();

        } catch (PDOException $e) {

            echo $e->getMessage();

        }

    }

    public static function isThere($jsumbang)

    {

        try {

            $db = static::getDB();

            $stmt = $db->query("SELECT `id` FROM `jenis\_sumbangan` WHERE `name`='".$jsumbang."' LIMIT 1");

            $results = $stmt->fetch(PDO::FETCH\_ASSOC);

            if($results < 1) return FALSE;

            return $results["id"];

        } catch (PDOException $e) {

            echo $e->getMessage();

        }

    }

    public static function setJS($jsumbang)

    {

        try {

            $db = static::getDB();

            $stmt = $db->query("INSERT INTO `jenis\_sumbangan`(`name`) VALUES ('". $jsumbang ."')");

            $results = $stmt->fetchAll(PDO::FETCH\_ASSOC);

            return $db->lastinsertId();

        } catch (PDOException $e) {

            echo $e->getMessage();

        }

    }

    public static function setSumbangan($userid,$jenisid,$jumlah)

    {

        try {

            $db = static::getDB();

            $stmt = $db->query("INSERT INTO `sumbangan`(`userid`, `jenis`, `jumlah`) VALUES ('".$userid."' ,'".$jenisid."','".$jumlah."')");

            $results = $stmt->fetchAll(PDO::FETCH\_ASSOC);

            return;

        } catch (PDOException $e) {

            echo $e->getMessage();

        }

    }

    public static function getSumbangan()

    {

        try {

            $db = static::getDB();

            $stmt = $db->query("SELECT (@cnt := @cnt + 1) AS nomer, b.name, c.name jenis, a.jumlah, b.gender

            FROM `sumbangan` a

            CROSS JOIN (SELECT @cnt := 0) AS dummy

            INNER JOIN user b ON b.id = a.userid

            INNER JOIN jenis\_sumbangan c ON c.id=a.jenis

            WHERE 1");

            $results = $stmt->fetchAll(PDO::FETCH\_ASSOC);

            return $results;

        } catch (PDOException $e) {

            echo $e->getMessage();

        }

    }

    public static function getFilterSumbangan($name)

    {

        try {

            $db = static::getDB();

            $stmt = $db->query("SELECT (@cnt := @cnt + 1) AS nomer, b.name, c.name jenis, a.jumlah, b.gender

            FROM `sumbangan` a

            CROSS JOIN (SELECT @cnt := 0) AS dummy

            INNER JOIN user b ON b.id = a.userid

            INNER JOIN jenis\_sumbangan c ON c.id=a.jenis

            WHERE c.name= '".$name."'");

            $results = $stmt->fetchAll(PDO::FETCH\_ASSOC);

            return $results;

        } catch (PDOException $e) {

            echo $e->getMessage();

        }

    }

}

1. Dibawah ini merupakan code dari file **base.html** yang isinys hampir sama dengan tugas-tugas sebelumnya karena memang saya menggunakan folder yang sama, hanya saja yang membedakan adalah saya membuat navbar sesuai dengan website yaitu **Sumbang dan List Sumbangan, lalu setting judul website, dll** . **Pada file ini juga saya menginstall bootstrap untuk tampilan.**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>{{Judul}}</title>

    <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css" integrity="sha384-ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQUOhcWr7x9JvoRxT2MZw1T" crossorigin="anonymous">

    <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js" integrity="sha384-JjSmVgyd0p3pXB1rRibZUAYoIIy6OrQ6VrjIEaFf/nJGzIxFDsf4x0xIM+B07jRM" crossorigin="anonymous"></script>

    <script src="https://code.jquery.com/jquery-3.3.1.slim.min.js" integrity="sha384-q8i/X+965DzO0rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo" crossorigin="anonymous"></script>

    <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js" integrity="sha384-UO2eT0CpHqdSJQ6hJty5KVphtPhzWj9WO1clHTMGa3JDZwrnQq4sF86dIHNDz0W1" crossorigin="anonymous"></script>

    <link rel="stylesheet" type="text/css" href="https://cdn.datatables.net/v/bs4/dt-1.10.21/datatables.min.css"/>

    <script type="text/javascript" src="https://cdn.datatables.net/v/bs4/dt-1.10.21/datatables.min.js"></script>

</head>

<body>

    <nav class="navbar navbar-expand-lg navbar-dark bg-dark">

        <a class="navbar-brand" href="#">DONASI COVID-19</a>

        <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarNav" aria-controls="navbarNav" aria-expanded="false" aria-label="Toggle navigation">

          <span class="navbar-toggler-icon"></span>

        </button>

        <div class="collapse navbar-collapse" id="navbarNav">

          <ul class="navbar-nav">

            <li class="nav-item">

              <a class="nav-link" href="./">Sumbang {% if nav == TRUE %}<span class="sr-only">(current)</span>{% endif %}</a>

            </li>

            <li class="nav-item">

              <a class="nav-link" href="Jenis">List Sumbangan {% if nav == FALSE %}<span class="sr-only">(current)</span>{% endif %}</a>

            </li>

          </ul>

        </div>

      </nav>

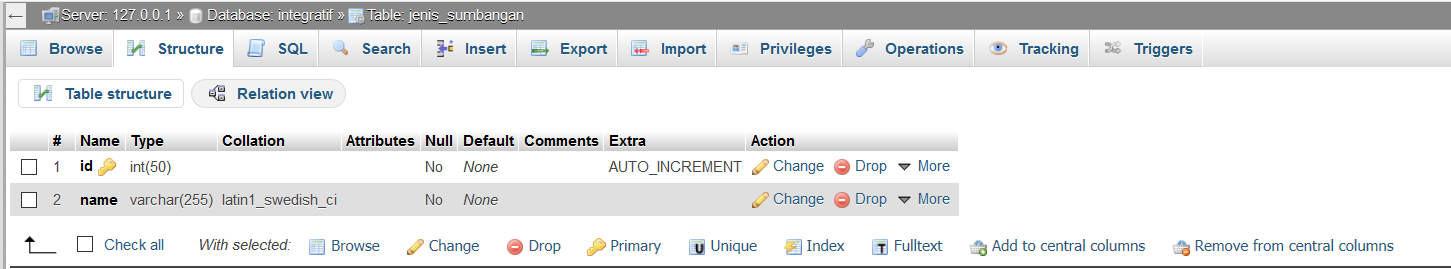
    {% block body %}

    {% endblock %}

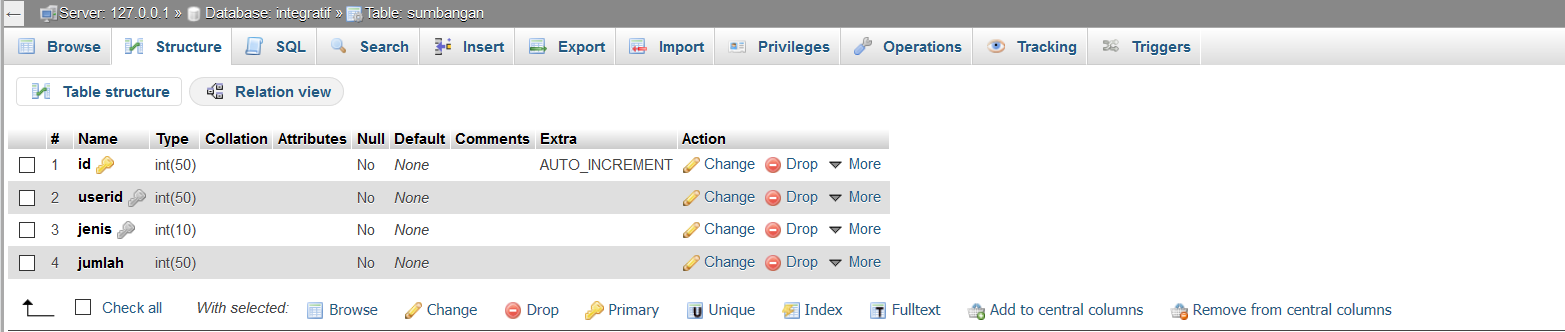
</body>

</html>

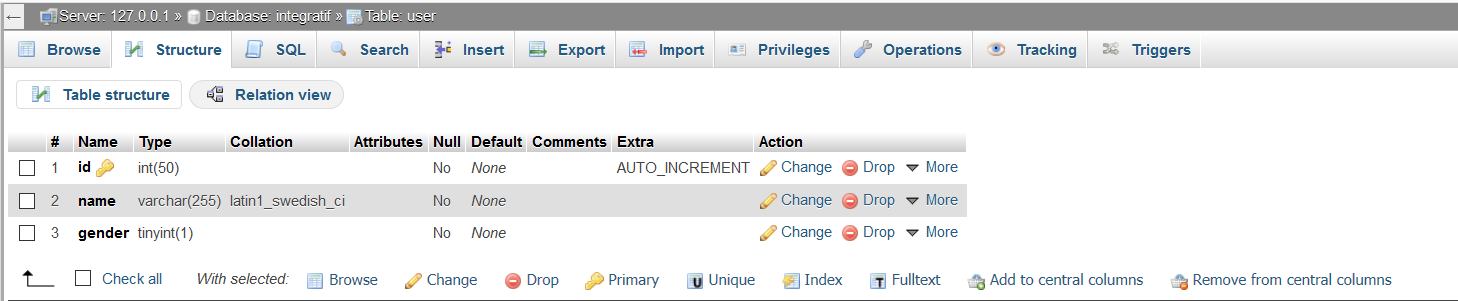
* Berikut ini adalah screenshot dari **structure 3 table database** yang saya buat yaitu tabel **jenis\_sumbangan, sumbangan, dan user:**



**Gambar structure pada table Jenis\_sumbangan**

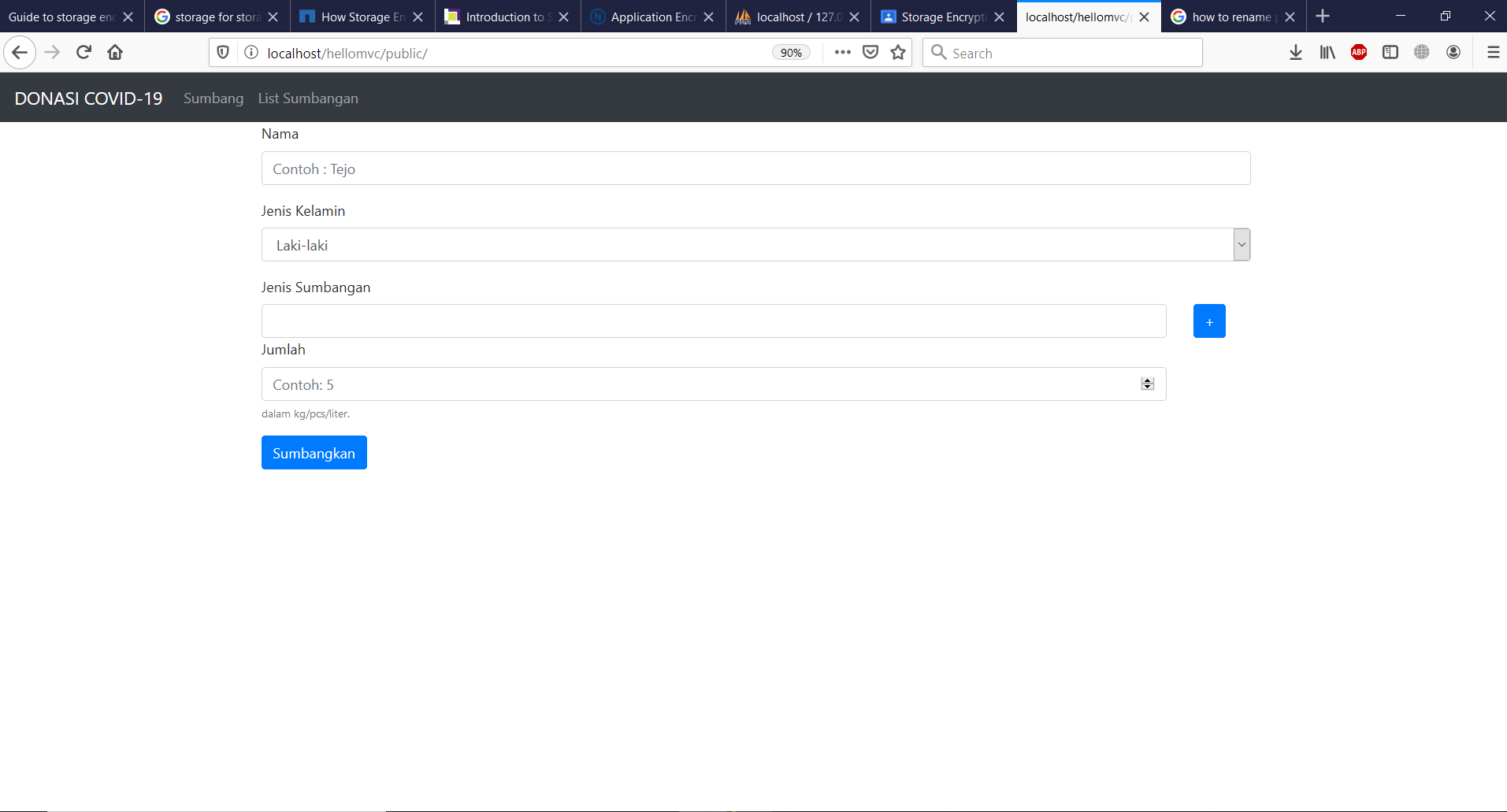


**Gambar structure pada table sumbangan**

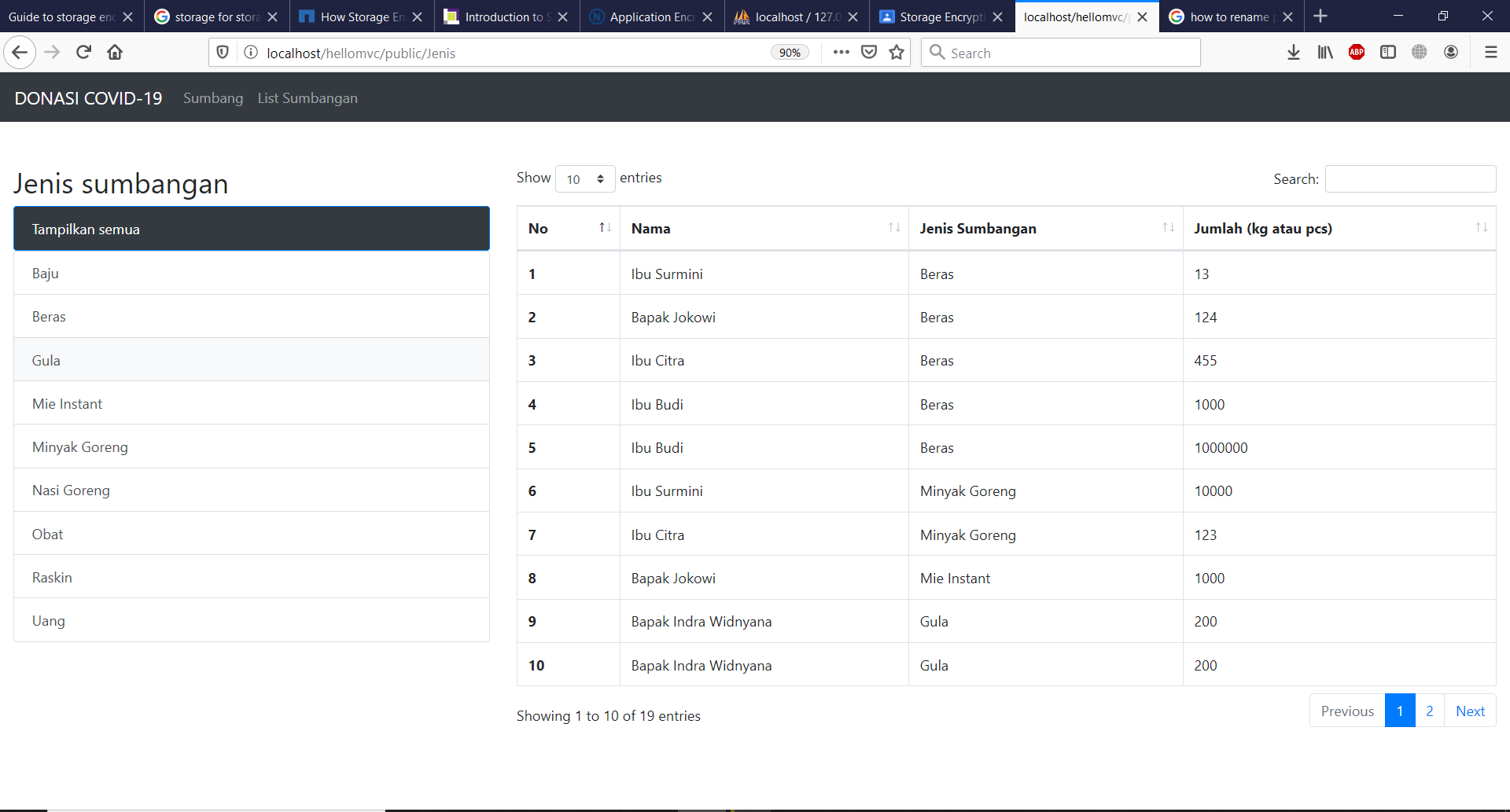


**Gambar structure pada table user**

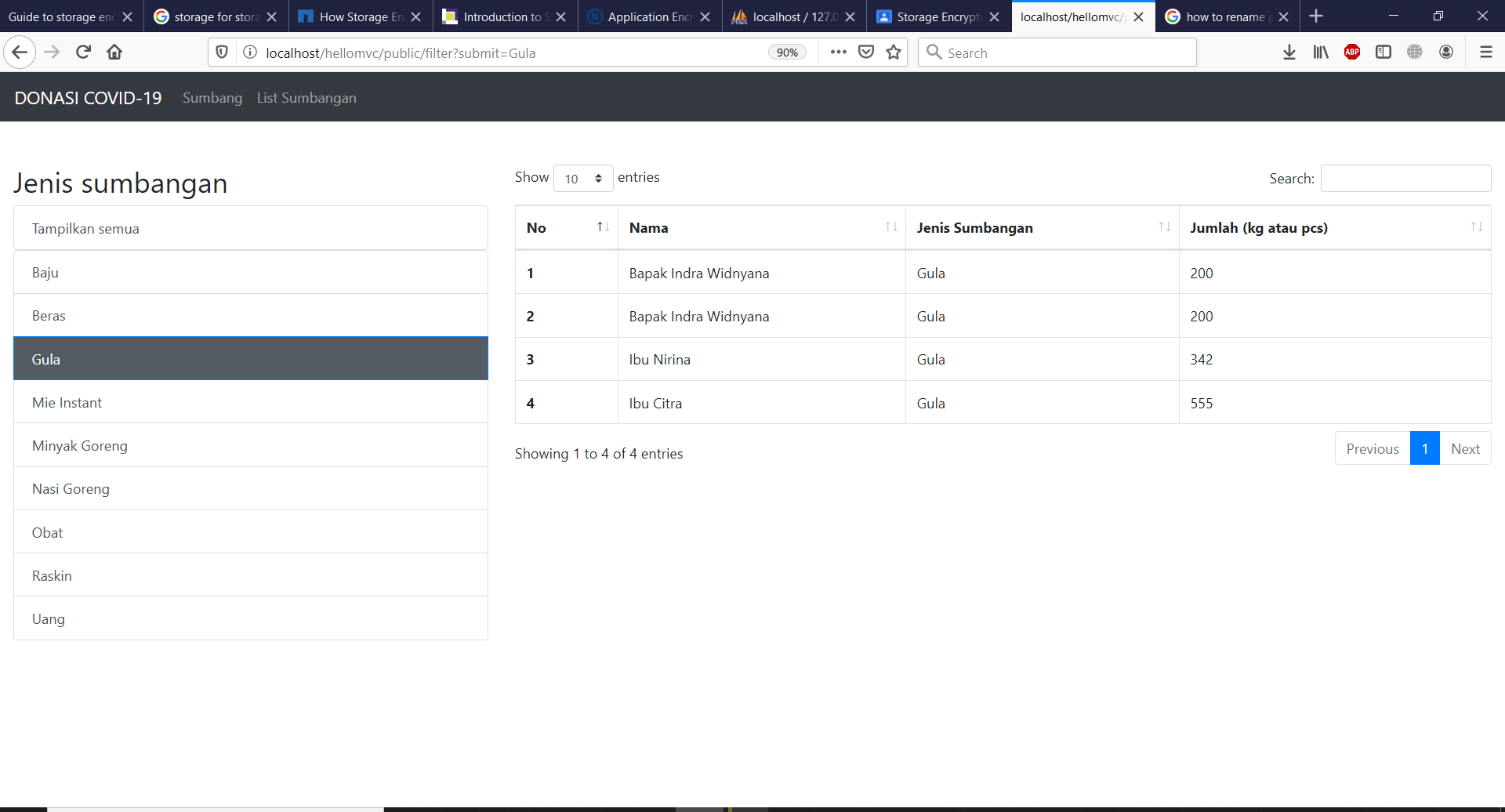
* Berikut ini adalah screenshot dari website **DONASI COVID-19**

****

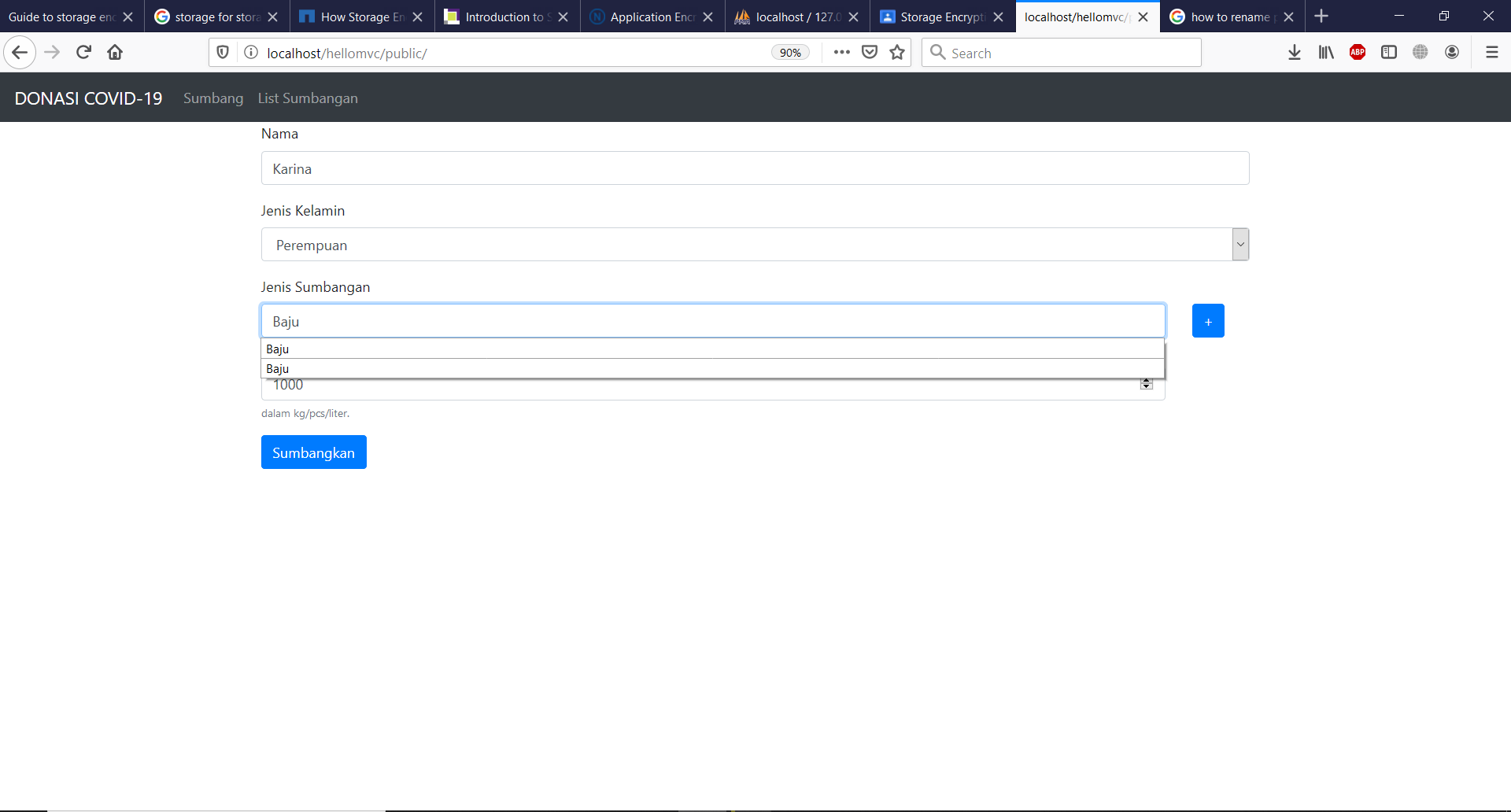
**Gambar diatas adalah tampilan awal website DONASI COVID-19 yang dimana terdiri dari navbar sumbang (untuk menyumbang) dan List sumbangan. Tampilan ini sekaligus menjadi tampilan pada navbar sumbang yang merupakan form untuk input.**



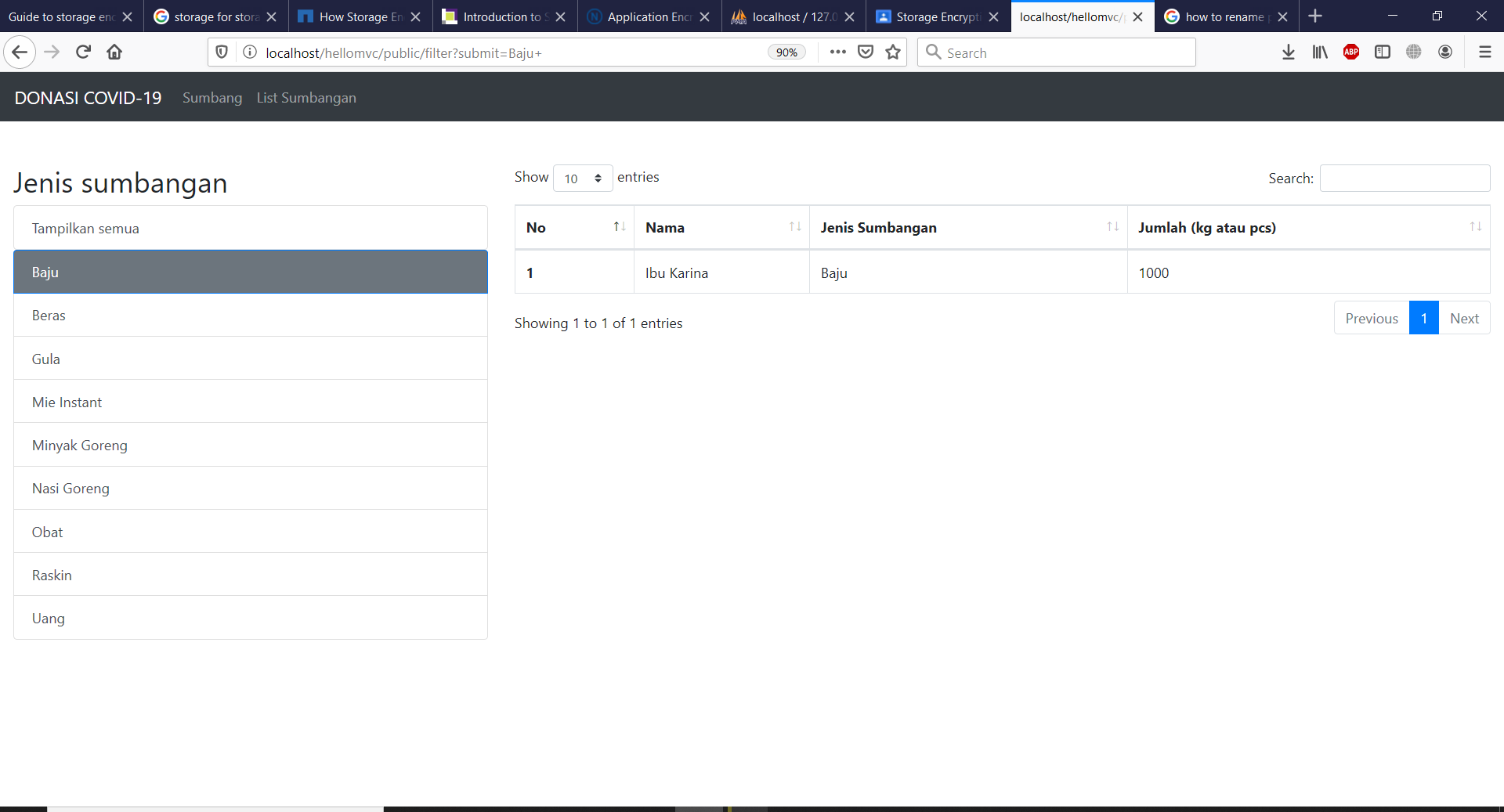
**Gambar diatas adalah tampilan dari List Sumbangan yang salah satu fiturnya adalah sorting sesuai nomor dan filter berdasarkan Jenis Sumbangan**

****

**Gambar diatas adalah tampilan dari List Sumbangan dengan menggunakan filter jenis sumbangan GULA**



**Gambar diatas adalah tampilan percobaan untuk input data yaitu (Karina, dengan jenis kelamin perempuan, jenis sumbangan baju, dan bobot sumbangan yaitu 1000 pcs**

****

**Gambar diatas adalah tampilan konfirmasi bahwa data masuk ke website dengan nama dan ketentuan sesuai yang diinput, dan dicek dalam fitur filter dengan jenis sumbangan yaitu Baju**