

PEMROGRAMAN WEB BERBASIS SERVICES TUGAS AKHIR – AWONAPAKARYA

Wahyu Pambudi – 19312302 (Backend)
Muhammad Ali Nasir – 19312248 (FrontEnd)

PENJELASAN PROGRAM

Team AwonapaKarya membuat Website Berbasis Services dengan study Kasus Website Website Inventaris Barang Sekolah.

Penjelasan Program Backend

- Get Data Barang

```

1  GET http://localhost/TA/Server/index.php/Barang/
2  Status: 200 OK Size: 590 Bytes Time: 90 ms
3  Response
4  {
5    "barang": [
6      {
7        "id_barang": "4",
8        "kd_barang": "008",
9        "nm_barang": "Komputer Betul",
10       "spek_barang": "Keren betul",
11       "jml_barang": "10",
12       "kondisi_barang": "Bagus",
13       "tgl_bay": "2022-12-13",
14       "harga_barang": "99999"
15     },
16     {
17       "id_barang": "3",
18       "kd_barang": "brg01",
19       "nm_barang": "Monitor Samsung 5248350 ",
20       "spek_barang": "24 Inch, IPS, 16 : 9",
21       "jml_barang": "10"
22     }
23   ]
24 }

```

```

1  <?php
2  defined('BASEPATH') OR exit('No direct script access allowed');
3
4  require APPPATH . 'libraries/Server.php';
5
6  class Barang extends Server {
7
8  // membuat fungsi get data
9  function service_get()
10 {
11 // test menggunakan kd_brg
12 $kd_brg = $this->get('kd_brg');
13
14 // panggil model Mbarang
15 $this->load->model('Mbarang', 'model', TRUE);
16
17 if ($kd_brg == '') {
18 // panggil fungsi get_data
19 $brg = $this->model->get_data();
20 $this->response(array('barang' => $brg), 200);
21 } else if ($kd_brg != '') {
22 // query where
23 $this->db->where('kd_brg', $kd_brg);
24 // panggil fungsi get_data
25 $brg = $this->model->get_data();
26 $this->response(array('barang' => $brg), 200);
27 } else {
28 $this->response(array('status' => "Data Tidak Ditemukan",
29 404));
30 }
31 }
32 }

```

```

1  <?php
2  defined('BASEPATH') OR exit('No direct script access allowed');
3
4  class Mbarang extends CI_Model {
5
6  // membuat model untuk tampil data
7  function get_data() {
8    $id as id_barang,
9    kd_brg as kd_barang,
10   nm_brg as nm_barang,
11   spek_brg as spek_barang,
12   jml_brg as jml_barang,
13   kondisi_brg as kondisi_barang,
14   tgl_bay_brg as tgl_bay,
15   harga_brg as harga_barang
16 };
17
18 $this->db->from('tb_barang');
19 $this->db->order_by('kd_brg', 'ASC');
20
21 $query = $this->db->get()->result();
22 return $query;
23 }
24
25 // membuat model untuk post data barang
26 function save_data($kd_brg, $nm_brg, $spek_brg, $jml_brg,
27 $kondisi_brg, $tgl_bay_brg, $harga_brg, $token)
28 {
29 // check kd_brg apakah ada di table

```

- Mendapatkan data barang dari table barang, dengan proses select dari table barang.
- nama kolom di buat menjadi alias dari **kd_brg** menjadi **kd_barang** dan lainnya.
- Membuat kondisi dengan parameter kd_brg

```

1  GET http://localhost/TA/Server/index.php/Barang/?kd_brg=brg01
2
3  Query Parameters
4  kd_brg brg01
5
6  parameter value

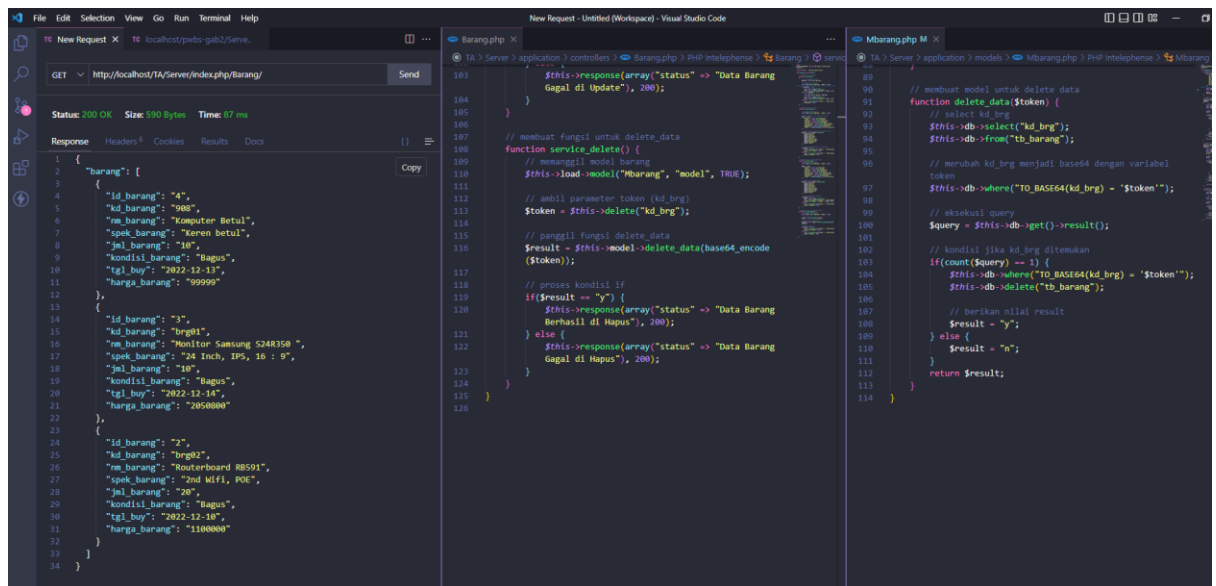
```

Status: 200 OK Size: 217 Bytes Time: 67 ms

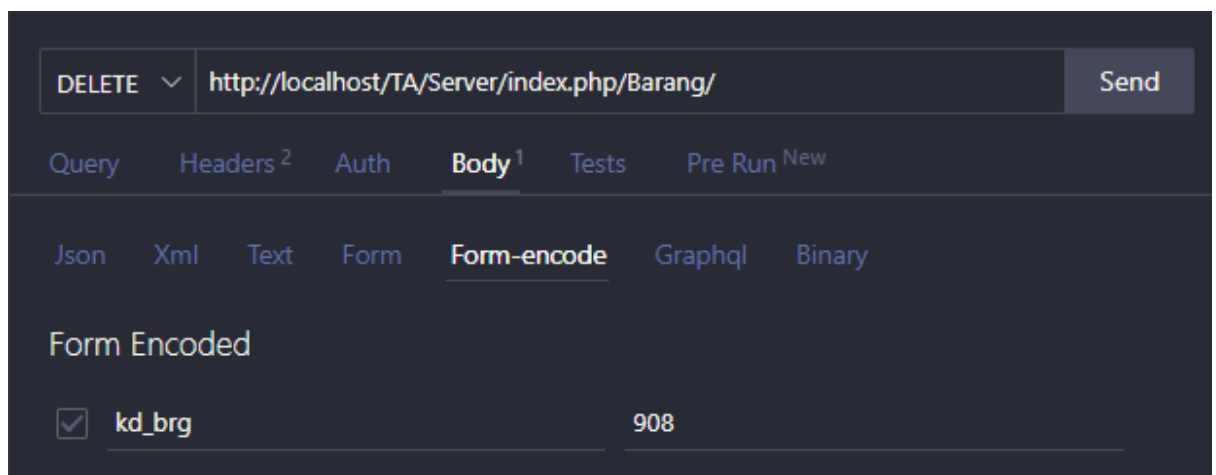
Response Headers⁶ Cookies Results Docs {} ≡

```
1  {
2    "barang": [
3      {
4        "id_barang": "3",
5        "kd_barang": "brg01",
6        "nm_barang": "Monitor Samsung S24R350 ",
7        "spek_barang": "24 Inch, IPS, 16 : 9",
8        "jml_barang": "10",
9        "kondisi_barang": "Bagus",
10       "tgl_buy": "2022-12-14",
11       "harga_barang": "2050800"
12     }
13   ]
14 }
```

• Hapus / DELETE Data Barang



- Proses hapus data barang, dengan mendapatkan terlebih dahulu `kd_brg` dari table barang.
 - Pada model barang, `kd_brg` di konversi ke base64 dengan variable token, kemudian pada controller barang, di kembali di encode dari base64.
 - Jika kondisi terpenuhi maka proses query berjalan.
 - Dan jika berhasil akan menampilkan pesan berhasil dihapus, jika gagal akan menampilkan pesan gagal dihapus.
- Hasil hapus data barang



DELETE ⌵ http://localhost/TA/Server/index.php/Barang/ Send

Query Headers ² Auth Body ¹ Tests Pre Run ^{New}

Json Xml Text Form Form-encode Graphql Binary

Form Encoded

☒ kd_brg 908

—

Status: 200 OK Size: 42 Bytes Time: 154 ms

Response Headers ⁶ Cookies Results Docs {} ≡

```
1 {
2   "status": "Data Barang Berhasil di Hapus"
3 }
```

Copy

- Add / POST Data Barang

POST ☐ http://localhost/TA/Server/index.php/Barang/ Send

Query Headers² Auth **Body¹** Tests Pre Run^{New}

Json Xml Text Form **Form-encode** GraphQL Binary

| | | |
|-------------------------------------|----------|-------------|
| <input checked="" type="checkbox"/> | kd_brg | 908 |
| <input checked="" type="checkbox"/> | nm_brg | Mikrotik RB |
| <input checked="" type="checkbox"/> | spek_brg | LAN 5 PORT |

⋮

Status: 200 OK **Size: 43 Bytes** **Time: 95 ms**

Response Headers⁶ Cookies Results Docs {} ≡

```

1 {
2   "status": "Data Barang Berhasil di Simpan"
3 }
```

```

{
  "id_barang": "6",
  "kd_barang": "908",
  "nm_barang": "Mikrotik RB",
  "spek_barang": "LAN 5 PORT",
  "jml_barang": "15",
  "kondisi_barang": "Bagus",
  "tgl_buy": "2022-12-04",
  "harga_barang": "1100000"
},
```

- Proses simpan data dengan cara menampung hasil inputan user, dan di tampung pada variable data berupa array.
- Dimulai dari select kd_brg dari tb_barang, dimana token konversi ke base64.
- Jika kd_brg kosong, maka melakukan proses query untuk insert data ke tb_barang.
- Setiap hasil akan menampilkan response sesuai dari controller barang pada function service_post.

• Edit / PUT Data Barang

The screenshot shows a web browser on the left displaying a REST client interface. The URL is `http://localhost/TA/Server/index.php/Barang/`. The response status is 200 OK, and the response body is a JSON array of barang data. On the right, two PHP files are shown: `Barang.php` and `Mbarang.php`. The `Barang.php` file contains a `function service_put()` that handles the PUT request, updates the data in the database, and returns a response. The `Mbarang.php` file contains a `function update_data()` that updates the data in the database.

```

// Barang.php
function service_put() {
    // membuat fungsi untuk update data
    // mengambil model barang
    $this->load->model('Mbarang', 'model', TRUE);

    $data = array(
        "kd_brg" => $this->put("kd_brg"),
        "nm_brg" => $this->put("nm_brg"),
        "spek_brg" => $this->put("spek_brg"),
        "jml_brg" => $this->put("jml_brg"),
        "kondisi_brg" => $this->put("kondisi_brg"),
        "tgl_buy_brg" => $this->put("tgl_buy_brg"),
        "harga_brg" => $this->put("harga_brg"),
        "token" => base64_encode($this->put("kd_brg"))
    );

    // panggil model update_data
    $result = $this->model->update_data(
        $data["kd_brg"],
        $data["nm_brg"],
        $data["spek_brg"],
        $data["jml_brg"],
        $data["kondisi_brg"],
        $data["tgl_buy_brg"],
        $data["harga_brg"],
        $data["token"]
    );

    // kondisi penyesuaian Mbarang
    if($result == 0) {
        $brg = $this->model->get_data();
        $this->response(array("status" => "Data Barang
        Berhasil di Update", "barang" => $brg), 200);
    } else {
        $this->response(array("status" => "Data Barang
        Gagal di Update", 200);
    }
}

// Mbarang.php
function update_data($kd_brg, $nm_brg, $spek_brg,
    $jml_brg, $kondisi_brg, $tgl_buy_brg, $harga_brg,
    $token) {
    // membuat model untuk input data
    $this->db->select("kd_brg");
    $this->db->from("tb_barang");
    $this->db->where("TO_BASE64(kd_brg) != 'Token' AND
    kd_brg = '$kd_brg'");

    // eksekusi query
    $query = $this->db->get()->result();
    // kondisi jika kd_brg tidak di temukan
    if(count($query) == 0) {
        $data = array(
            "kd_brg" => $kd_brg,
            "nm_brg" => $nm_brg,
            "spek_brg" => $spek_brg,
            "jml_brg" => $jml_brg,
            "kondisi_brg" => $kondisi_brg,
            "tgl_buy_brg" => $tgl_buy_brg,
            "harga_brg" => $harga_brg
        );
        $this->db->where("TO_BASE64(kd_brg) = 'Token'");
        $this->db->update("tb_barang", $data);
        $result = 0;
    } else {
        $result = 1;
    }
    return $result;
}

// membuat model untuk delete data
function delete_data($token) {
    // ambil kd_brg
    $this->db->select("kd_brg");
}

```

```

"status": "Data Barang Berhasil di Update",
"barang": [
{
    "id_barang": "6",
    "kd_barang": "908",
    "nm_barang": "Mikrotik RB 954",
    "spek_barang": "LAN 5 PORT",
    "jml_barang": "20",
    "kondisi_barang": "Bagus",
    "tgl_buy": "2022-12-04",
    "harga_barang": "1100000"
},

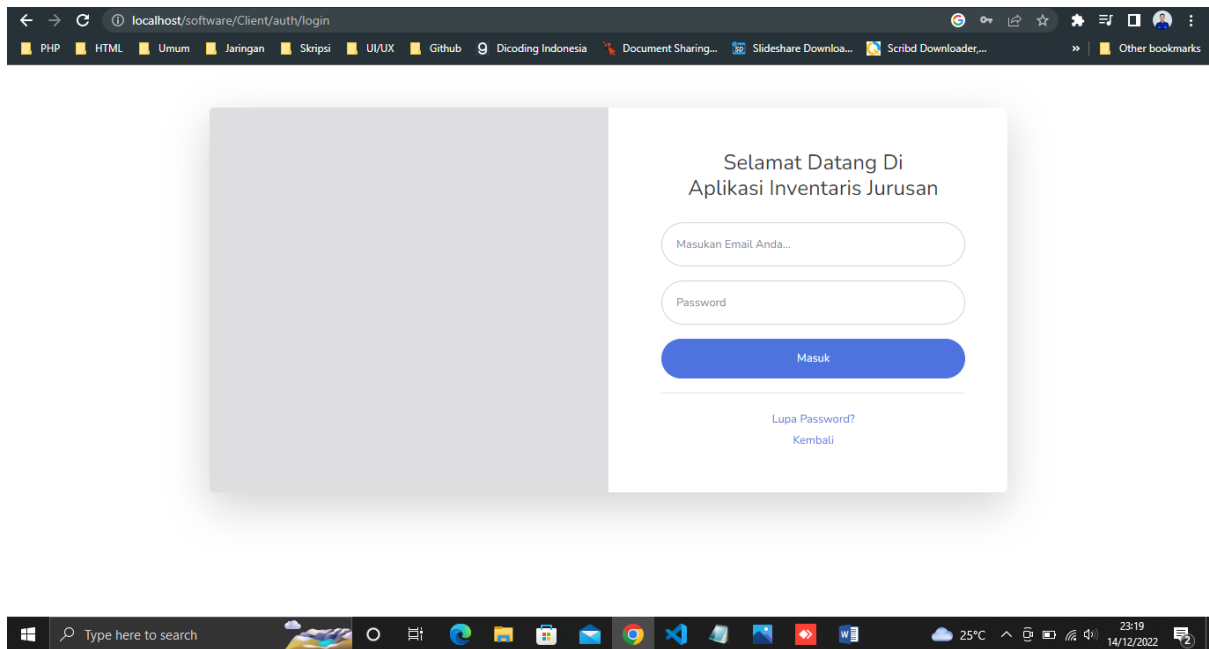
```

- Proses update / put data barang, dengan menerima parameter dari masing-masing inputan
- Kemudian melakukan select data dari table
- Membaca kondisi dari query, jika terpenuhi maka kondisi akan dijalankan dan data akan di update sesuai dengan controller barang.

Penjelasan Program Front End



Pada gambar diatas Terdapat halaman Homepage yang terdapat tulisan dibagian Navbar Nama Sekolah, Home dan Tentang dan terdapat dibagian Body “ Selamat Datang di Sistem Aplikasi Inventaris Barang Jurusan Sekolah” serta aset background dan terdapat Button masuk yang mengarahkan ke Halaman Login.



Selanjutnya pada gambar diatas adalah Halaman Login dimana halaman tersebut dikemas dengan Card menggunakan Framework CSS yaitu Bootstrap 4 dan terdapat tulisan dan Form untuk memasukan Email dan Password dan juga terdapat Menu Lupa Password untuk mereset password serta Menu Kembali ke Halaman Homepage, selain itu terdapat Button Masuk untuk masuk ke Halaman Dashboard. Halaman Dashboard dibuat untuk Proses CRUD pada Aplikasi tersebut. Yang memiliki beberapa Fitur seperti:

1. Menu Belanja
 - Barang Modal
 - Barang dan Jasa
2. Menu User
 - Admin
 - Kepala Jurusan

The screenshot shows a web application interface with a sidebar on the left and a main content area. The sidebar has a logo 'TEKAJE' and a 'Dashboard' link. The main content area has a search bar and a title 'Tabel Barang Modal'. Below the title is a paragraph explaining DataTables. A 'DataTables Example' box contains a table with 7 columns: Kode Barang, Nama Barang, Spesifikasi, Jumlah, Kondisi, Tanggal Pembelian, and Harga Barang. The table has 2 data rows. Below the table is a pagination bar showing 'Showing 1 to 2 of 2 entries' and 'Previous 1 Next'.

Search for...

Tabel Barang Modal

DataTables is a third party plugin that is used to generate the demo table below. For more information about DataTables, please visit the [official DataTables documentation](#).

DataTables Example

Show entries Search:

| Kode Barang | Nama Barang | Spesifikasi | Jumlah | Kondisi | Tanggal Pembelian | Harga Barang |
|-----------------|------------------|-------------|--------|------------|-------------------|--------------|
| Garrett Winters | Accountant | Tokyo | 63 | 2011/07/25 | \$320,800 | \$170,750 |
| Tiger Nixon | System Architect | Edinburgh | 61 | 2011/04/25 | \$320,800 | \$320,800 |

Showing 1 to 2 of 2 entries Previous 1 Next

