

LAPORAN PEMROGRAMAN WEB FRAMEWORK

“Login & API”



DISUSUN OLEH :

Ratih Cahyani Apriliya

E31191660

GOLONGAN C

MANAJEMEN INFORMATIKA

TEKNOLOGI INFORMASI

POLITEKNIK NEGERI JEMBER

2021

BAB I

PENDAHULUAN

1.1 Latar Belakang

Dalam dunia industri yang maju sangat pesat ini, tidak menutup kemungkinan dari hasil karya sekumpulan manusia yang mempunyai suatu tujuan. Di masa yang sudah canggih ini kita sering menggunakan Web (website atau situs) dapat diartikan sebagai kumpulan halaman yang menampilkan informasi data teks, data gambar diam atau gerak, data animasi, suara, video dan atau gabungan dari semuanya, baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait dimana masing-masing dihubungkan dengan jaringan-jaringan halaman (hyperlink). Bersifat statis apabila isi informasi website tetap, jarang berubah, dan isi informasinya searah hanya dari pemilik website. Bersifat dinamis apabila isi informasi website selalu berubah-ubah, dan isi informasinya interaktif dua arah berasal dari pemilik serta pengguna website. Contoh website statis adalah berisi profil perusahaan, sedangkan website dinamis adalah seperti Friendster, Multiply, dll. Dalam sisi pengembangannya, website statis hanya bisa diupdate oleh pemiliknya saja, sedangkan website dinamis bisa diupdate oleh pengguna maupun pemilik. Website ini digunakan untuk mencari tahu informasi yang kita butuhkan, bahkan kita bisa berkomunikasi dengan orang lain pun lewat jejaring sosial. Di dalam kita berselancar di dunia maya kita memerlukan browser untuk mencari informasi apa yang kita inginkan di dunia maya. Lewat browser terdapat script – script yang di mengerti oleh computer. Dan penulis membahasnya dalam laporan praktikum ini yaitu mengenai HTML (Hyper Text Markup Language), CSS (Cascading Style Sheets), dan PHP (HyperText Preprocessor)

1.2 Rumusan Masalah

- 1.2.1 Bagaimana konsep Login dan Session pada CI?
- 1.2.2 Bagaimana menggunakan Login dan Session pada CI?
- 1.2.3 Bagaimana konsep Web Service (REST API) pada CI?
- 1.2.4 Bagaimana Web Service (REST API) pada CI?

1.3 Tujuan

- 1.3.1 Mahasiswa dapat mengetahui konsep Login dan Session pada CI.
- 1.3.2 Mahasiswa dapat menggunakan Login dan Session pada CI.
- 1.3.3 Mahasiswa dapat mengetahui konsep Web Service (REST API) pada CI.
- 1.3.4 Mahasiswa dapat menggunakan Web Service (REST API) pada CI.

BAB II

DASAR TEORI

2.1 Sesssion

Sebuah web server tidak dapat mengenali siapa saja yang sedang mengakses dan apa yang dilakukan oleh user karena protokol HTTP bersifat stateless. Sebuah protokol dinyatakan stateless karena tidak menyimpan informasi mengenai interaksi yang terjadi sebelumnya, sehingga setiap interaksi yang terjadi dianggap sebagai interaksi yang independen dan tidak terkait satu sama lain. Ketika user mengakses suatu halaman website, misalnya index, dan berpindah ke halaman lain di website yang sama, webserver akan menganggap setiap interaksi yang bersifat request-response tersebut sebagai interaksi tunggal. Tidak ada kaitannya antara satu halaman dengan halaman yang lain. Agar webserver dapat 'mengingat' informasi pengakses, PHP menyediakan sebuah mekanisme yang disebut session. Dengan adanya session, setiap pengakses diberi pengenal (ID) yang akan digunakan selama mengakses sebuah website. Pengenal tersebut dapat didefinisikan oleh pembuat programer. Informasi pada session disimpan di server. Oleh sebab itu, session akan hilang secara otomatis jika user menutup browser atau meninggalkan (ditutup/ke luar) website tersebut. User akan mendapatkan pengenal yang baru ketika mengakses kembali halaman website yang sama.

Session memiliki fungsi seperti tiket di bioskop atau di tempat-tempat yang membutuhkan tiket. Fungsi tiket untuk memastikan bahwa seseorang berhak untuk masuk ke dalam bioskop. Petugas bioskop hanya dapat mengizinkan seseorang untuk masuk ke bioskop, apabila dapat menunjukkan tiket yang telah dibelinya. Apabila di tengah pertunjukan seorang penonton ke luar sebentar, misal untuk membeli makanan atau ke toilet, maka tiket yang sama harus ditunjukkan kepada petugas penjaga agar dapat masuk kembali ke bioskop. Jika ternyata tiket hilang, maka petugas penjaga berhak melarang penonton tersebut untuk masuk kembali. Pemanfaatan session sangat terasa ketika menggunakan aplikasi berbasis web seperti ecommerce, media sosial seperti Twitter dan Facebook, atau blog.

2.2 Rest API

API merupakan singkatan dari *application programming interfaces*, yang secara singkatnya API bisa digunakan oleh developer untuk mengizinkan dan mengintegrasikan aplikasi yang berada di platform yang berbeda ataupun perangkat yang berbeda untuk bisa saling terkoneksi antara satu dengan yang lainnya. Tujuan utama kenapa API ini digunakan oleh developer dalam pembuatan sebuah perangkat lunak adalah untuk saling berbagi data antara aplikasi yang sama dari platform yang sama ataupun berbeda. Tentu saja penggunaan API tidak hanya sesederhana itu,

API juga bisa digunakan untuk menyediakan function sehingga para developer tidak perlu membuat fungsi lagi dalam coding nya hanya perlu memanggilnya saja. Sedangkan REST merupakan kependekan dari *Representational State Transfer*. REST merupakan sebuah web service yang berjalan di client dan server dimana yang memiliki sifat stateless. Yang mempunyai arti bahwa setiap request yang dikirim oleh aplikasi maka harus menyertakan semua parameter dan datanya dengan lengkap.

Secara singkat cara kerjanya adalah client REST (aplikasinya) akan menjalankan request ke server REST, ketika request terkirim ke server selanjutnya REST server akan memberikan respon. Kemudian client REST akan menampilkan responnya atau jika tidak, client akan melakukan pemrosesan yang lain. Respon dari REST Server ke REST client dapat diberikan dalam berbagai format, yaitu ada HTML, XML, Json, atau beberapa format yang lain. Namun format yang paling sering dipakai oleh para developer adalah Json, karena formatnya lebih mudah untuk digunakan dan dipelajari. Untuk bisa menggunakan REST ini, REST memiliki sebuah standarisasi yang harus dipakai dalam implementasinya, standarisasi REST ini menggunakan URL dan juga HTTP verb. Dengan menggunakan standarisasi yang menggunakan URL maka developer dapat melakukan beberapa operasi berdasarkan HTTP verbs. Secara teknis HTTP verbs ini berupa `$_SERVER['REQUEST_METHOD']`

Ada 5 method request HTTP yang sangat umum digunakan pada arsitektur REST ini, yaitu:

1. GET, digunakan untuk mengakses data atau membaca data yang ada pada resource.
2. PUT, digunakan untuk men-create atau membuat sebuah resource baru.
3. DELETE, digunakan untuk menghapus resource.
4. PATCH, untuk mengupdate kumpulan data (field) yang ada di dalam resource.
5. POST, untuk memperbaharui sebuah resource, atau menambahnya.

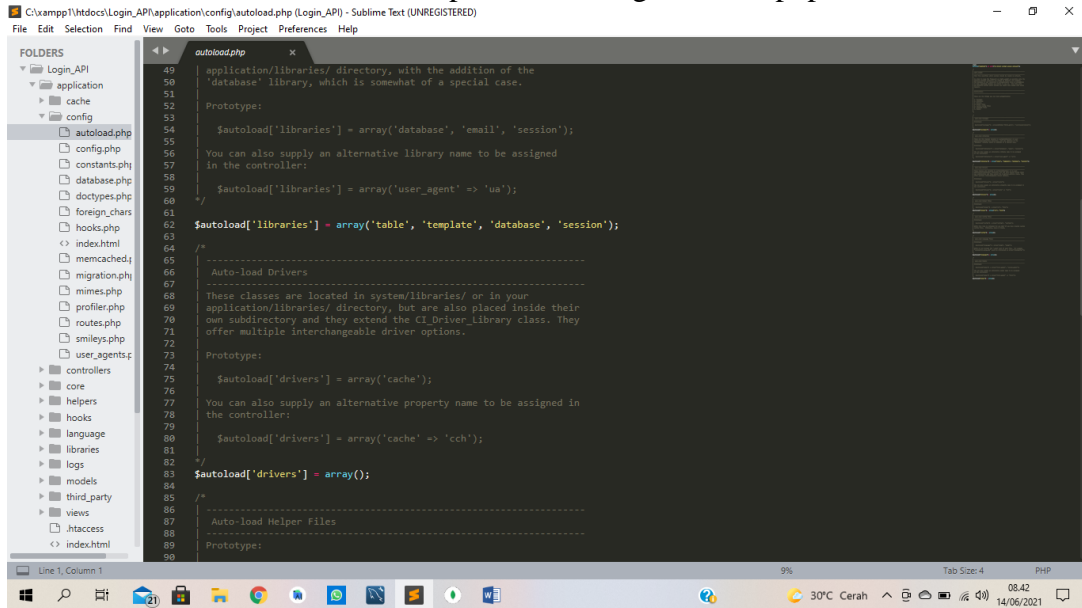
Namun sebenarnya HTTP request method ini ada 9, yaitu 4 yang lainnya adalah connect, option, trace dan head. Dengan adanya standarisasi URL ini maka developer yang menggunakan REST API akan lebih sedikit untuk membuat dokumentasinya, karena ini merupakan sebuah standar yang mutlak sehingga tidak diperlukan lagi penjelasan mengenai membaca, menambah, mengupdate dan beberapa operasi lainnya, menganggap para pengguna sudah paham mengenai request HTTP verb yang digunakan. Jadi pada dasarnya menggunakan REST API ini akan lebih mempermudah developer dalam pembuatan project-nya, karena semuanya sudah terstandarisasi sehingga dalam sebuah tim semua akan berbentuk dan lebih mudah untuk dimengerti satu sama lainnya.

BAB III

HASIL DAN PEMBAHASAN

3.1 Praktikum Login & Session

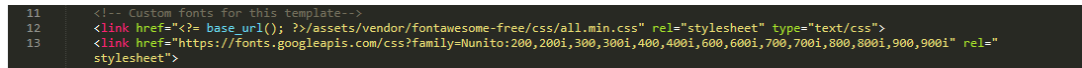
a. Tambahkan librari session pada file config/autoload.php



```
49 application/libraries/ directory, with the addition of the
50 'database' library, which is somewhat of a special case.
51
52 Prototype:
53
54 $autoload['libraries'] = array('database', 'email', 'session');
55
56 You can also supply an alternative library name to be assigned
57 in the controller:
58
59 $autoload['libraries'] = array('user_agent' => 'ua');
60
61
62 $autoload['libraries'] = array('table', 'template', 'database', 'session');
63
64
65 -----
66 Auto-load Drivers
67 -----
68 These classes are located in system/libraries/ or in your
69 application/libraries/ directory, but are also placed inside their
70 own subdirectory and they extend the CI_Driver_Library class. They
71 offer multiple interchangeable driver options.
72
73 Prototype:
74
75 $autoload['drivers'] = array('cache');
76
77 You can also supply an alternative property name to be assigned in
78 the controller:
79
80 $autoload['drivers'] = array('cache' => 'cch');
81
82
83 $autoload['drivers'] = array();
84
85 -----
86 Auto-load Helper Files
87 -----
88 Prototype:
```

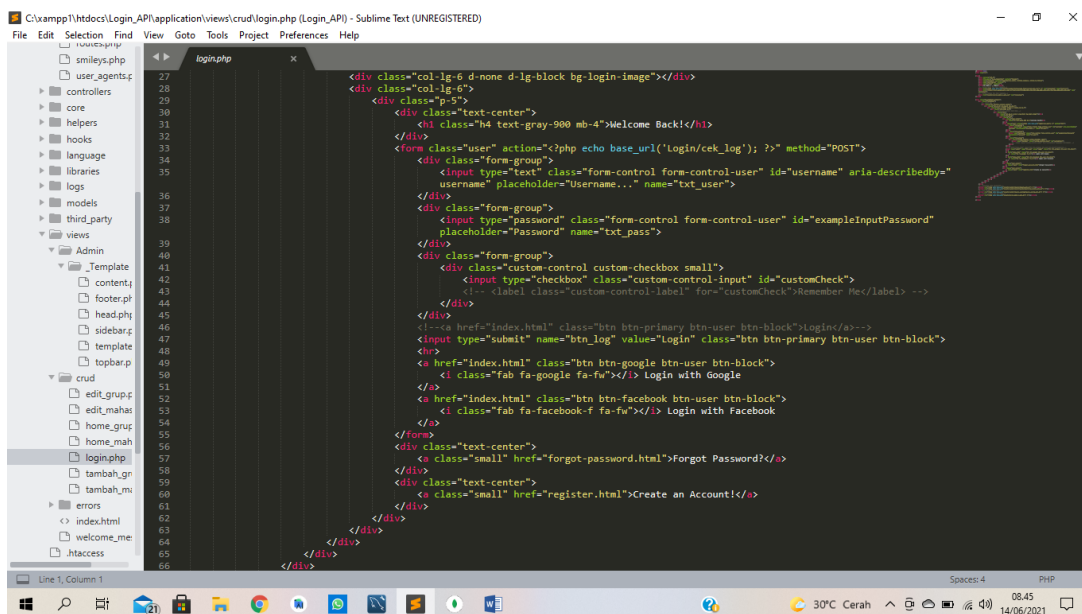
b. Buat file view login.php dari file login.html dari sbadmin dirubah ekstensinya kemudiandiedit pada bagian :

head :

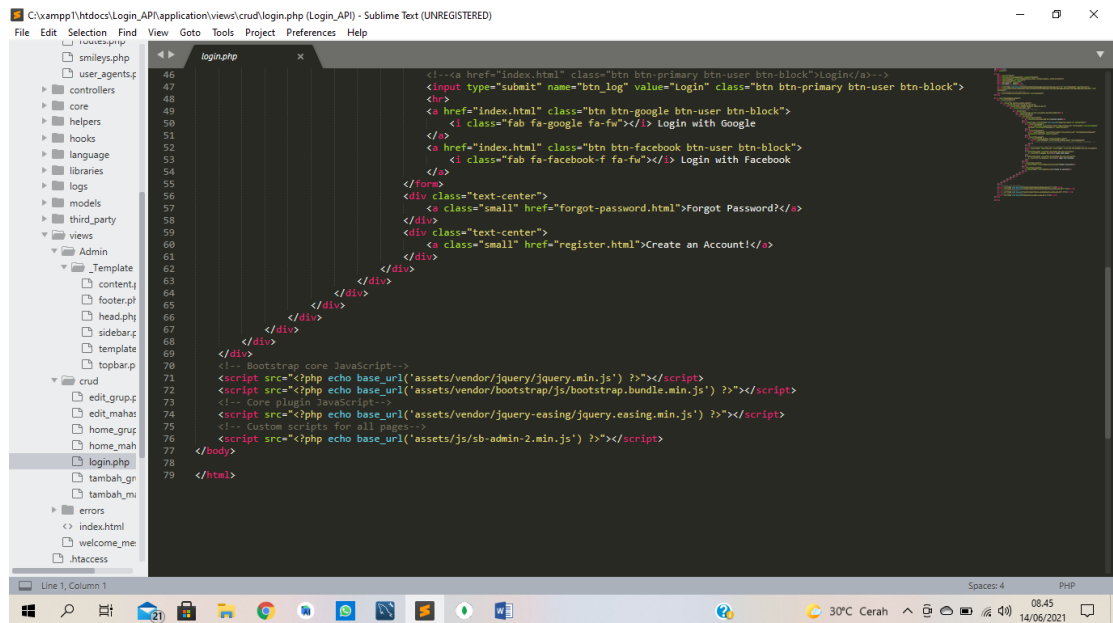


```
11 <!-- Custom fonts for this template -->
12 <link href="<? base_url(); ?>assets/vendor/fontawesome-free/css/all.min.css" rel="stylesheet" type="text/css">
13 <link href="https://fonts.googleapis.com/css?family=Nunito:200,200i,300,300i,400,400i,600,600i,700,700i,800,800i,900,900i" rel="stylesheet">
```

Line number 42

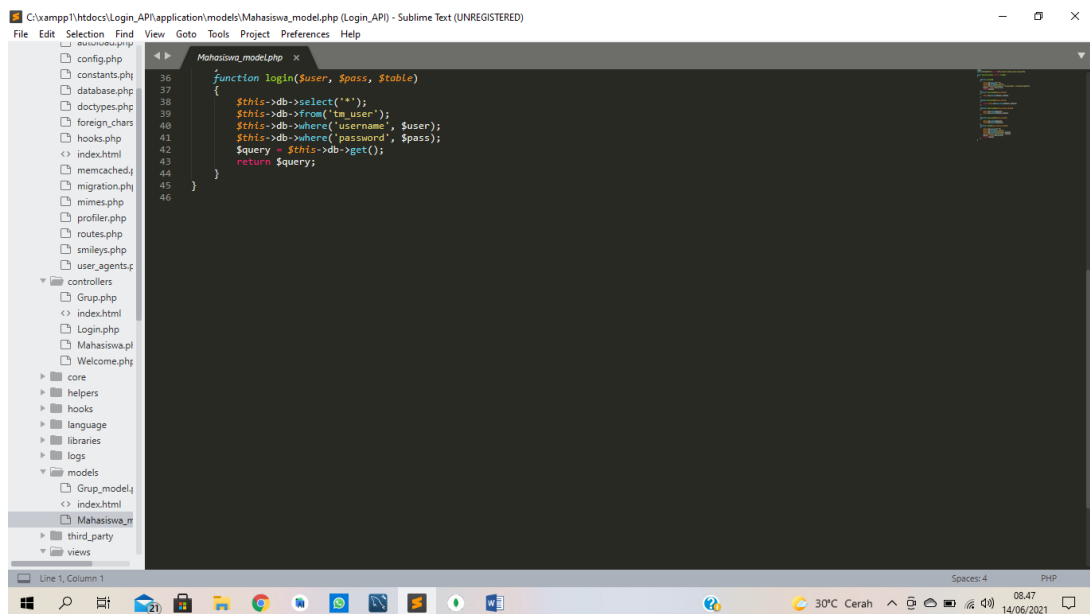


```
27 <div class="col-lg-6 d-none d-lg-block bg-login-image"></div>
28 <div class="col-lg-6">
29 <div class="p-5">
30 <div class="text-center">
31 <h1 class="h4 text-gray-900 mb-4">Welcome Back!</h1>
32 </div>
33 <form class="user" action="<?php echo base_url('login/cek_log'); ?>" method="POST">
34 <div class="form-group">
35 <input type="text" class="form-control form-control-user" id="username" aria-describedby="
36 username" placeholder="Username..." name="txt_user">
37 </div>
38 <div class="form-group">
39 <input type="password" class="form-control form-control-user" id="exampleInputPassword"
40 placeholder="Password" name="txt_pass">
41 </div>
42 <div class="form-group">
43 <input type="checkbox" class="custom-control-input" id="customCheck">
44 <label class="custom-control-label" for="customCheck">Remember Me</label> -->
45 </div>
46 </div>
47 <a href="index.html" class="btn btn-primary btn-user btn-block">Login</a>-->
48 <input type="submit" name="btn_log" value="Login" class="btn btn-primary btn-user btn-block">
49 </div>
50 <div class="text-center">
51 <a href="index.html" class="btn btn-google btn-user btn-block">
52 <i class="fab fa-google fa-fw"></i> Login with Google
53 </a>
54 <a href="index.html" class="btn btn-facebook btn-user btn-block">
55 <i class="fab fa-facebook-f fa-fw"></i> Login with Facebook
56 </a>
57 </div>
58 <div class="text-center">
59 <a class="small" href="forgot-password.html">Forgot Password</a>
60 </div>
61 <div class="text-center">
62 <a class="small" href="register.html">Create an Account</a>
63 </div>
64 </div>
65 </div>
66 </div>
```



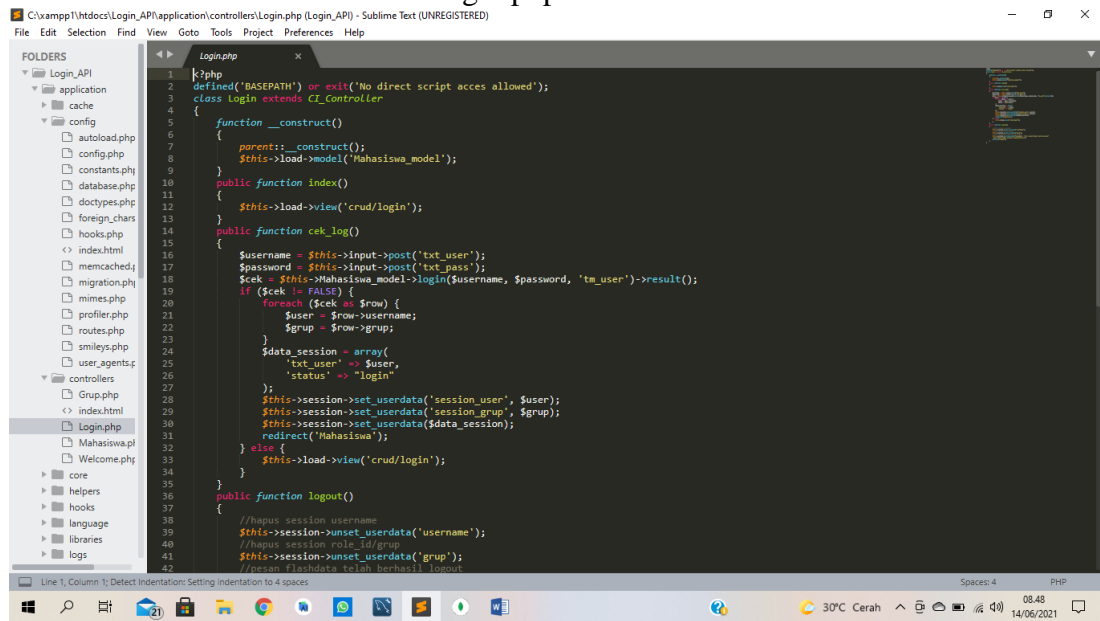
```
46 <!--<a href="index.html" class="btn btn-primary btn-user btn-block">login</a-->
47 <input type="submit" name="btn_log" value="Login" class="btn btn-primary btn-user btn-block">
48 <hr>
49 <a href="index.html" class="btn btn-google btn-user btn-block">
50 <i class="fab fa-google fa-fw"></i> Login with Google
51 </a>
52 <a href="index.html" class="btn btn-facebook btn-user btn-block">
53 <i class="fab fa-facebook fa-fw"></i> Login with Facebook
54 </a>
55 </form>
56 <div class="text-center">
57 <a class="small" href="forgot-password.html">Forgot Password?</a>
58 </div>
59 <div class="text-center">
60 <a class="small" href="register.html">Create an Account!</a>
61 </div>
62 </div>
63 </div>
64 </div>
65 </div>
66 </div>
67 </div>
68 </div>
69 </div>
70 <!-- Bootstrap core JavaScript-->
71 <script src="<?php echo base_url('assets/vendor/jquery/jquery.min.js') ?>"></script>
72 <script src="<?php echo base_url('assets/vendor/bootstrap/js/bootstrap.bundle.min.js') ?>"></script>
73 <!-- Core plugin JavaScript-->
74 <script src="<?php echo base_url('assets/vendor/jquery-easing/jquery.easing.min.js') ?>"></script>
75 <!-- Custom scripts for all pages-->
76 <script src="<?php echo base_url('assets/js/sb-admin-2.min.js') ?>"></script>
77 </body>
78 </html>
```

c. Pada Mahasiswa_model.php tambahkan baris berikut :

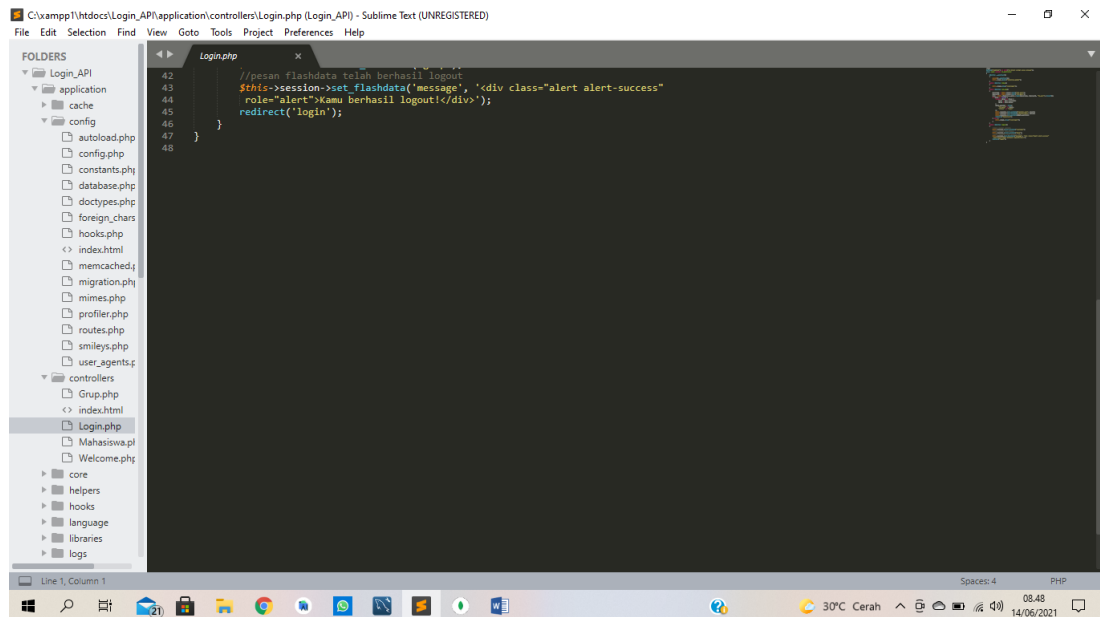


```
36 function login($user, $pass, $table)
37 {
38     $this->db->select('*');
39     $this->db->from('m_user');
40     $this->db->where('username', $user);
41     $this->db->where('password', $pass);
42     $query = $this->db->get();
43     return $query;
44 }
45
46 }
```

d. Buatlah controller baru : Login.php



```
1 k?php
2 defined('BASEPATH') or exit('No direct script acces allowed');
3 class Login extends CI_Controller
4 {
5     function __construct()
6     {
7         parent::__construct();
8         $this->load->model('Mahasiswa_model');
9     }
10    public function index()
11    {
12        $this->load->view('crud/login');
13    }
14    public function cek_log()
15    {
16        $username = $this->input->post('txt_user');
17        $password = $this->input->post('txt_pass');
18        $cek = $this->Mahasiswa_model->login($username, $password, 'ta_user')->result();
19        if ($cek != FALSE) {
20            foreach ($cek as $row) {
21                $user = $row->username;
22                $grup = $row->grup;
23            }
24            $data_session = array(
25                'txt_user' => $user,
26                'status' => "login"
27            );
28            $this->session->set_userdata('session_user', $user);
29            $this->session->set_userdata('session_grup', $grup);
30            $this->session->set_userdata($data_session);
31            redirect('Mahasiswa');
32        } else {
33            $this->load->view('crud/login');
34        }
35    }
36    public function logout()
37    {
38        //hapus session username
39        $this->session->unset_userdata('username');
40        //hapus session role_id/grup
41        $this->session->unset_userdata('grup');
42        //pesan flashdata telah berhasil logout
```



```
42 //pesan flashdata telah berhasil logout
43 $this->session->set_flashdata('message', '<div class="alert alert-success"
44     role="alert">Kamu berhasil logout!</div>');
45 redirect('login');
46 }
47 }
48 }
```

- e. Pada file view Admin/_Template/template.php tambahkan baris berikut pada bagian atas file:

```
1 <!DOCTYPE html>
2 <html lang='en'>
3
4 <?php echo $head; ?>
5 <?php
6 $getUser = $this->session->userdata('session_user');
7 $getGrup = $this->session->userdata('session_grup');
8 ?>
```

- f. Tambahkan juga baris seperti poin no 5 pada file view crud/home_mahasiswa.php

```
1 <?php
2 $getUser = $this->session->userdata('session_user');
3 $getGrup = $this->session->userdata('session_grup');
4 ?>
```

- g. Lalu pada file yang sama dengan poin 6 rubah baris berikut :

```
32 if ($getGrup == 1) {
33     echo '<a href=' . base_url('Mahasiswa/edit/' . $baris->id) . ' ' class='fa fa-edit'></a>';
34     echo ' ';
35     echo '<a href=' . base_url('Mahasiswa/hapus' . $baris->id) . ' ' class='fa fa-times'></a>';
36 }
```

- h. Tambahkan juga baris seperti poin no 5 pada file view Admin/_Template/sidebar.php

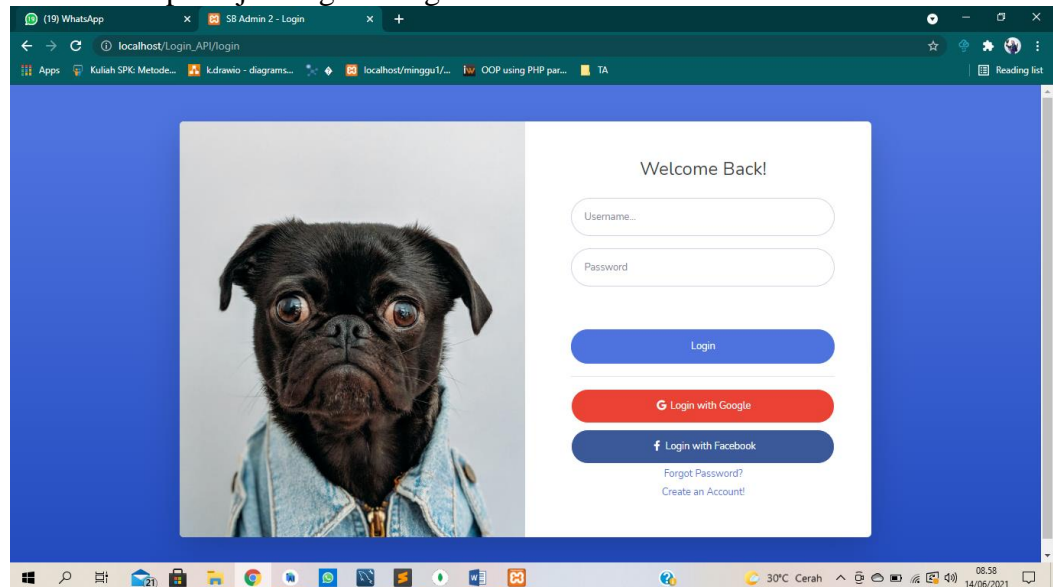
```
1 <!DOCTYPE html>
2 <html lang='en'>
3
4 <?php echo $head; ?>
5 <?php
6 $getUser = $this->session->userdata('session_user');
7 $getGrup = $this->session->userdata('session_grup');
8 ?>
```


- i. Lalu pada file yang sama dengan pon 8, rubah kode seperti pada bagian dibawah ini :

```
6 $getUser = $this->session->userdata('session_user');  
7 $getGrup = $this->session->userdata('session_grup');
```

- j. Coba akses file controller Login melalui browser. Jika login dan session berhasil maka akan ada kondisi seperti berikut

1. Tampilan jika login sebagai admin

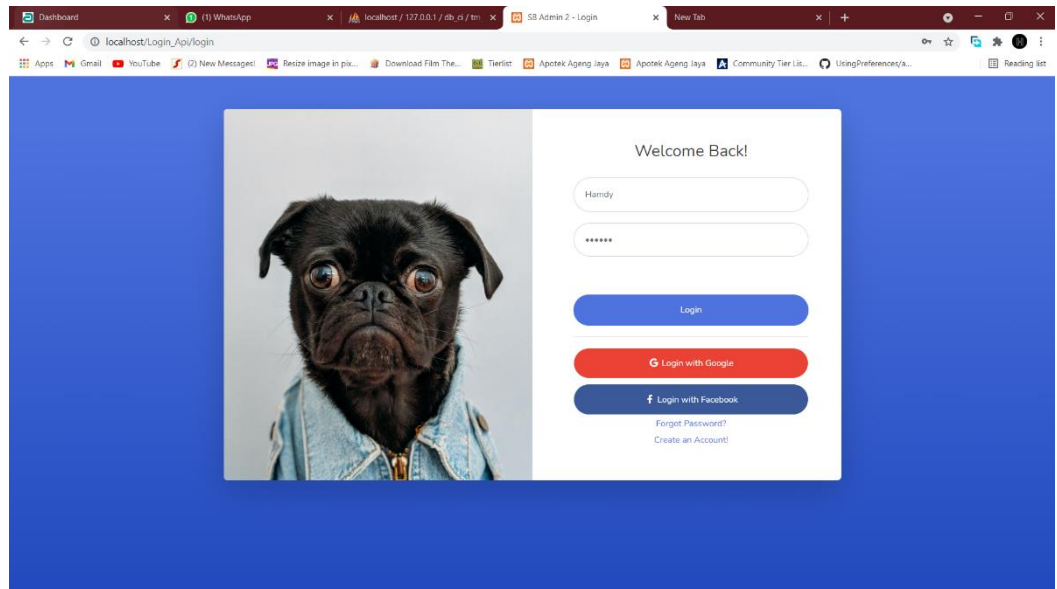


2. Tombol edit dan hapus muncul pada halaman home_mahasiswa.

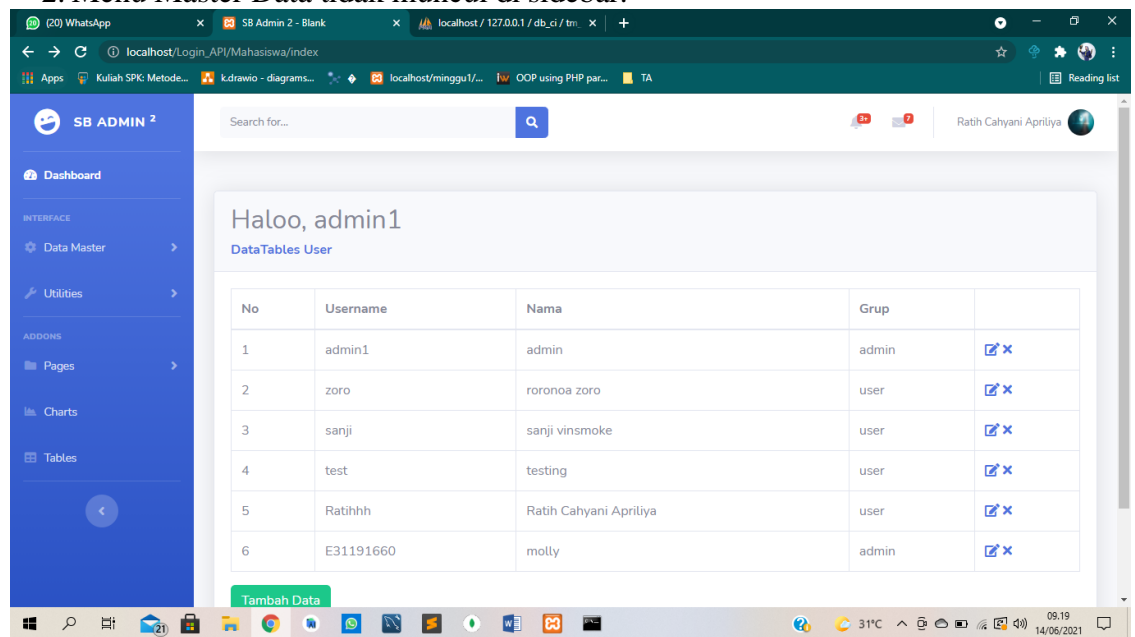
3. Menu Master Data muncul di sidebar

k. Tampilan jika login sebagai user

1. Tombol edit dan hapus tidak muncul pada halaman home_mahasiswa.

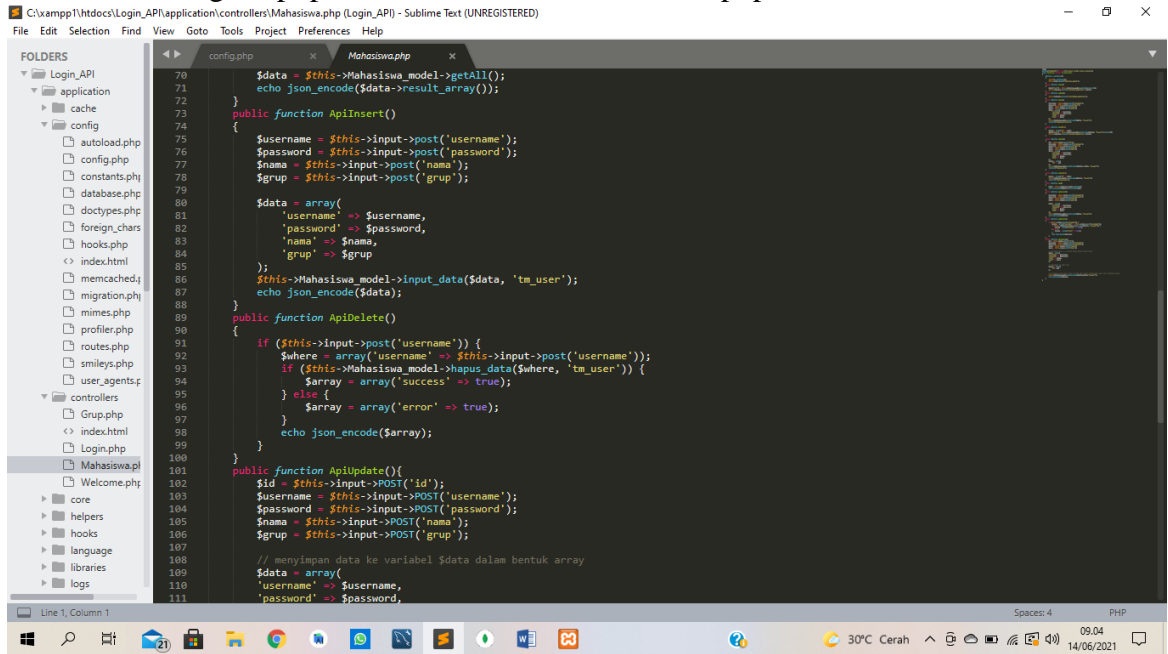


2. Menu Master Data tidak muncul di sidebar.



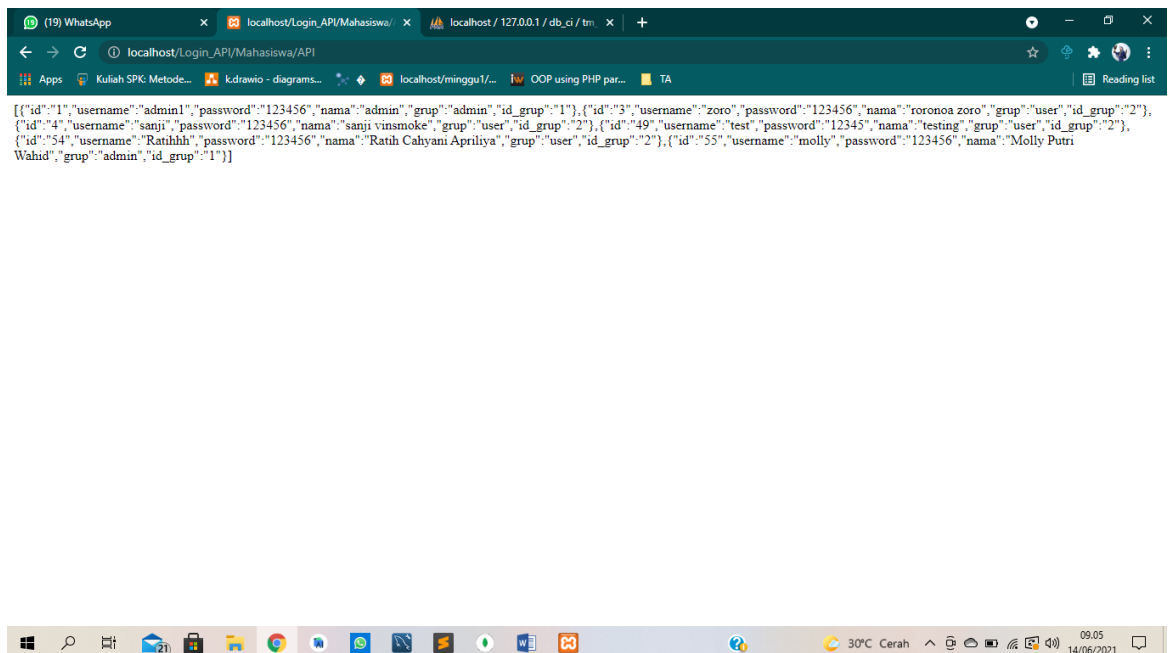
3.2 Praktikum API

1. Tambahkan fungsi Api pada file Controller/Mahasiswa.php



```
70 $data = $this->Mahasiswa_model->getAll();
71 echo json_encode($data->result_array());
72 }
73 public function ApiInsert()
74 {
75     $username = $this->input->post('username');
76     $password = $this->input->post('password');
77     $nama = $this->input->post('nama');
78     $grup = $this->input->post('grup');
79
80     $data = array(
81         'username' => $username,
82         'password' => $password,
83         'nama' => $nama,
84         'grup' => $grup
85     );
86     $this->Mahasiswa_model->input_data($data, 'tm_user');
87     echo json_encode($data);
88 }
89 public function ApiDelete()
90 {
91     if ($this->input->post('username')) {
92         $where = array('username' => $this->input->post('username'));
93         if ($this->Mahasiswa_model->hapus_data($where, 'tm_user')) {
94             $array = array('success' => true);
95         } else {
96             $array = array('error' => true);
97         }
98         echo json_encode($array);
99     }
100 }
101 public function ApiUpdate()
102 {
103     $id = $this->input->post('id');
104     $username = $this->input->post('username');
105     $password = $this->input->post('password');
106     $nama = $this->input->post('nama');
107     $grup = $this->input->post('grup');
108
109     // menyimpan data ke variabel $data dalam bentuk array
110     $data = array(
111         'username' => $username,
```

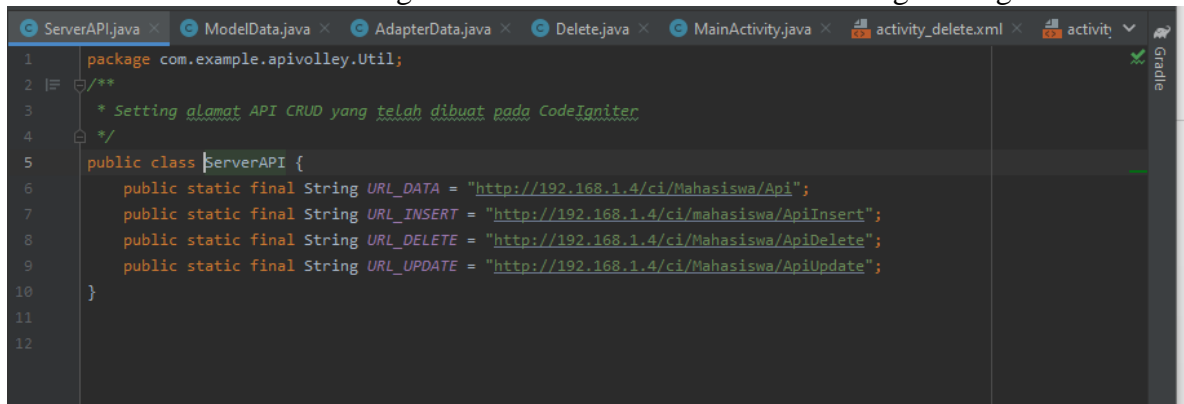
Sehingga jika dipanggil di Web Brower, maka akan muncul data keseluruhan dari database dalam bentuk JSON.



```
[{"id":"1","username":"admin","password":"123456","nama":"admin","grup":"admin","id_grup":"1"}, {"id":"3","username":"zoro","password":"123456","nama":"roronoa zoro","grup":"user","id_grup":"2"}, {"id":"4","username":"sanji","password":"123456","nama":"sanji vinsmoke","grup":"user","id_grup":"2"}, {"id":"49","username":"test","password":"12345","nama":"testing","grup":"user","id_grup":"2"}, {"id":"54","username":"Ratihhh","password":"123456","nama":"Rath Cahyani Apriliya","grup":"user","id_grup":"2"}, {"id":"55","username":"molly","password":"123456","nama":"Molly Putri Wahid","grup":"admin","id_grup":"1"}]
```

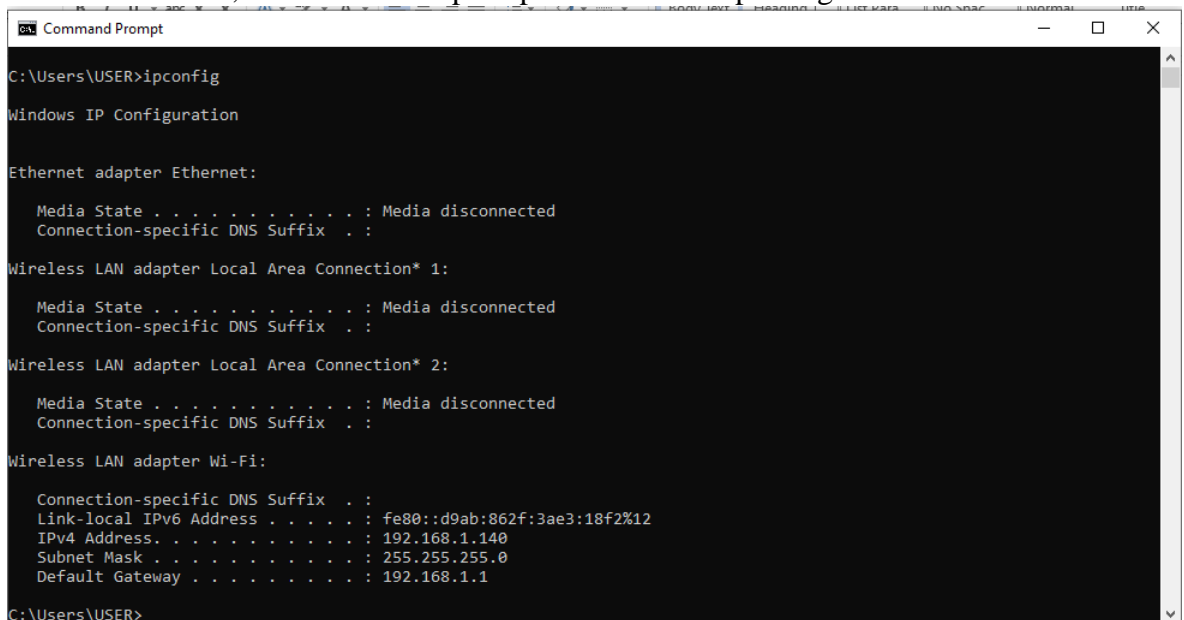
2. Mengimplementasikan hasil API yang telah dibuat dilakukan dengan cara mengkoneksikan dengan device client (Android).
3. Buka Android Studio dan buka project yang telah kita kerjakan sebelumnya kemudian lakukan Sync gradle.

4. Rubah alamat URL sesuai dengan IP dan alamat dari folder CI masing-masing.



```
1 package com.example.apivolley.Util;  
2 /**  
3  * Setting alamat API CRUD yang telah dibuat pada CodeIgniter  
4  */  
5 public class ServerAPI {  
6     public static final String URL_DATA = "http://192.168.1.4/ci/Mahasiswa/Api";  
7     public static final String URL_INSERT = "http://192.168.1.4/ci/mahasiswa/ApiInsert";  
8     public static final String URL_DELETE = "http://192.168.1.4/ci/Mahasiswa/ApiDelete";  
9     public static final String URL_UPDATE = "http://192.168.1.4/ci/Mahasiswa/ApiUpdate";  
10 }  
11  
12
```

Untuk Melihat IP, buka command prompt lalu ketikkan ipconfig.



```
C:\Users\USER>ipconfig  
  
Windows IP Configuration  
  
Ethernet adapter Ethernet:  
  
Media State . . . . . : Media disconnected  
Connection-specific DNS Suffix . :  
  
Wireless LAN adapter Local Area Connection* 1:  
  
Media State . . . . . : Media disconnected  
Connection-specific DNS Suffix . :  
  
Wireless LAN adapter Local Area Connection* 2:  
  
Media State . . . . . : Media disconnected  
Connection-specific DNS Suffix . :  
  
Wireless LAN adapter Wi-Fi:  
  
Connection-specific DNS Suffix . :  
Link-local IPv6 Address . . . . . : fe80::d9ab:862f:3ae3:18f2%12  
IPv4 Address. . . . . : 192.168.1.140  
Subnet Mask . . . . . : 255.255.255.0  
Default Gateway . . . . . : 192.168.1.1  
  
C:\Users\USER>
```

5. Coba jalankan aplikasi android jika berhasil maka akan tampil seperti berikut :
6. Tampilan Awal (Mengambil data dari JSON web)



7. Penambahan data melalui Android

CRUD_Volley

Username
E31192060

Nama
Edo Kristian Yusak

Password
232323

Grup
1

BATAL **SIMPAN**

CRUD_Volley

INSERT **DELETE**

Edo
Edo Kristian
123456
admin

edo mif
edo kristian yusak
jitpol
admin

chelsea
islan
232343
user

manchester
united
232323
user

E31192060
Edo Kristian Yusak
232323
admin

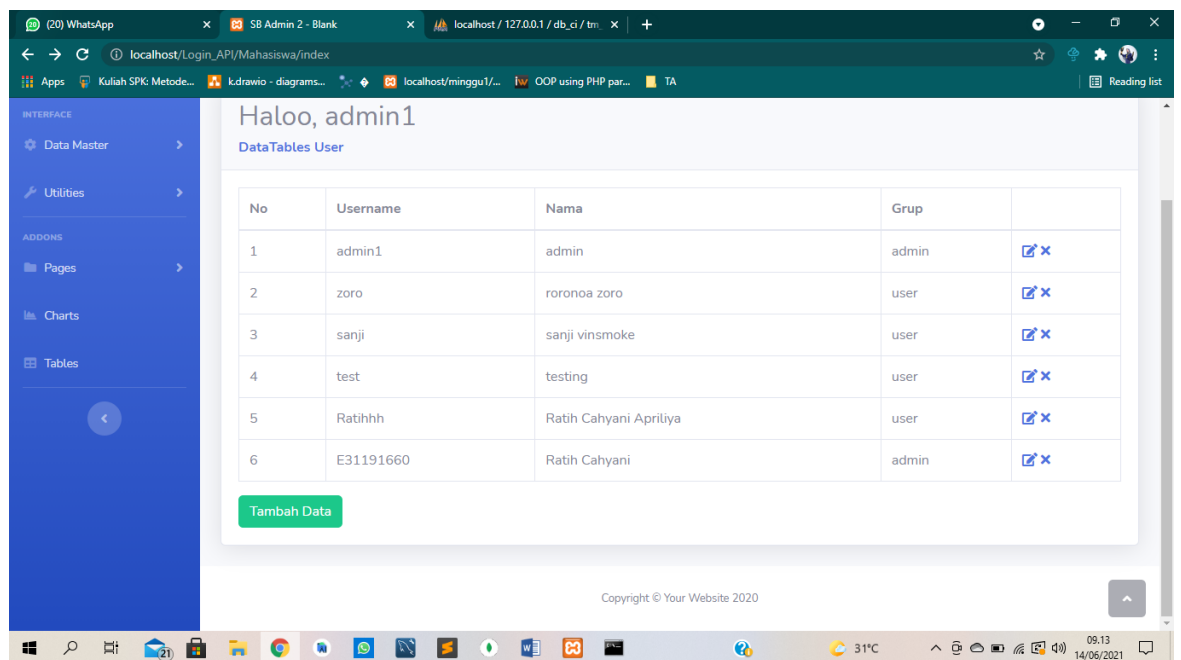
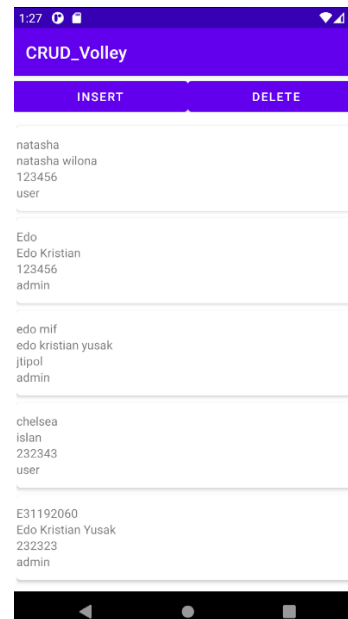
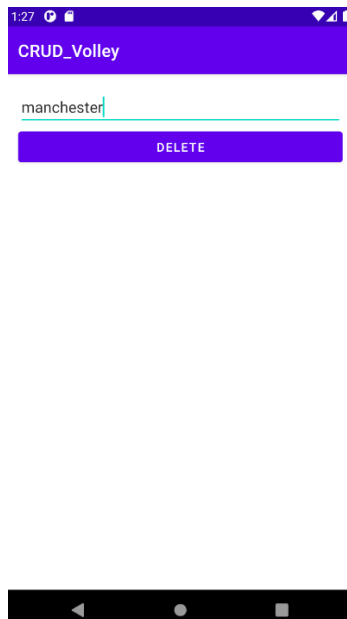
Hasil di Web,

The screenshot shows a web application interface with a sidebar menu and a main content area. The sidebar menu includes options like Dashboard, INTERFACE, Data Master, Utilities, ADDONS, Pages, Charts, and Tables. The main content area displays a greeting 'Haloo, admin1' and a table titled 'DataTables User'. The table contains 7 rows of user data, including No, Username, Nama, and Grup. Each row has edit and delete icons. A 'Tambah Data' button is located at the bottom of the table.

No	Username	Nama	Grup	
1	admin1	admin	admin	✎ ✕
2	zoro	rوروناo zoro	user	✎ ✕
3	sanji	sanji vinsmoke	user	✎ ✕
4	test	testing	user	✎ ✕
5	Ratihhh	Ratih Cahyani Apriliya	user	✎ ✕
6	molly	Molly Putri Wahid	admin	✎ ✕
7	E31191660	Ratih Cahyani	admin	✎ ✕

Tambah Data

Penghapusan data melalui Android



Update data melalui android

1:36

CRUD_Volley

Nama
Polije 19

Password
121212

Grup
admin

BATAL UPDATE DATA

1:38

CRUD_Volley

INSERT DELETE

admin1
admin
123456
admin

natasha
natasha wilona
123456
user

Edo
Edo Kristian
123456
admin

edo mif
Polije 19
121212
admin

chelsea
islan
232343
user

Hasil di Web

20 WhatsApp SB Admin 2 - Blank localhost / 127.0.0.1 / db_ci / tm

localhost/Login_API/Mahasiswa/index

Apps Kuliah SPK: Metode... k.drawio - diagrams... localhost/minggu1/... OOP using PHP par... TA

INTERFACE

- Data Master
- Utilities

ADDONS

- Pages
- Charts
- Tables

Haloo, admin1

DataTables User

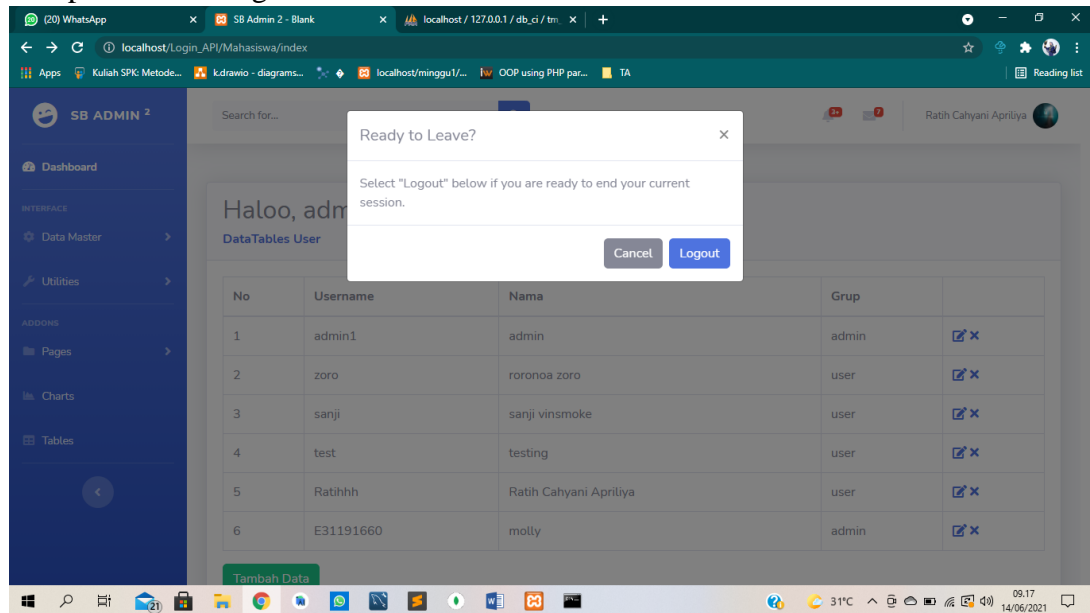
No	Username	Nama	Grup	
1	admin1	admin	admin	✎ ✕
2	zoro	roronoa zoro	user	✎ ✕
3	sanji	sanji vinsmoke	user	✎ ✕
4	test	testing	user	✎ ✕
5	Ratihhh	Ratih Cahyani Apriliya	user	✎ ✕
6	E31191660	molly	admin	✎ ✕

Tambah Data

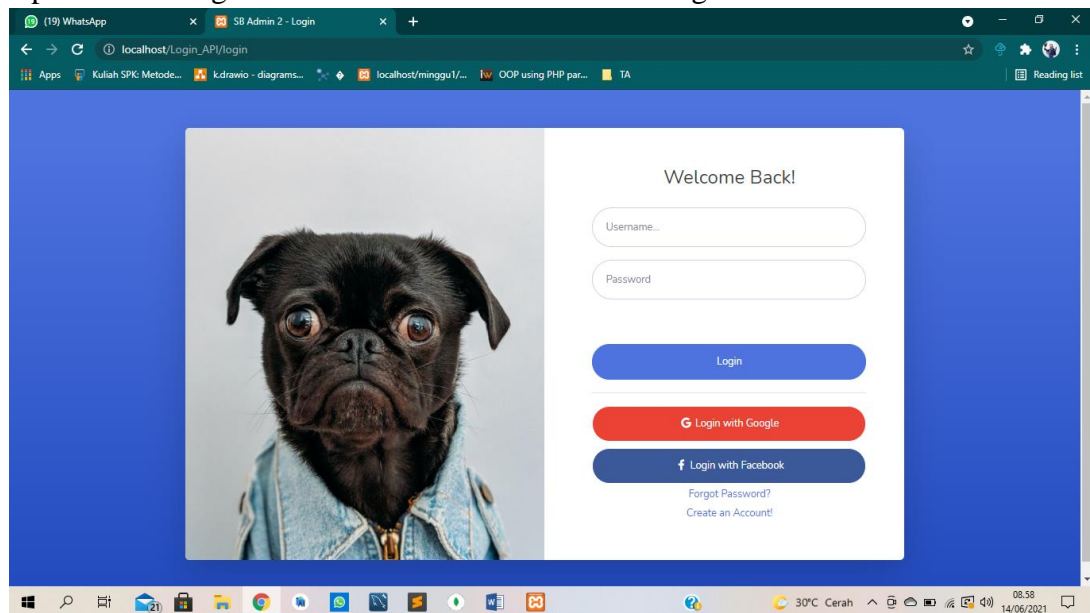
Copyright © Your Website 2020

09.15 14/06/2021

Tampilan Form Logout



Apabila klik Logout maka akan dialihkan ke menu login



BAB IV

PENUTUP

4.1 Kesimpulan

Ketika user mengakses suatu halaman website, misalnya index, dan berpindah ke halaman lain di website yang sama, webserver akan menganggap setiap interaksi yang bersifat request-response tersebut sebagai interaksi tunggal. Tidak ada kaitannya antara satu halaman dengan halaman yang lain. Agar webserver dapat 'mengingat' informasi pengakses, PHP menyediakan sebuah mekanisme yang disebut session. Dengan adanya session, setiap pengakses diberi pengenal (ID) yang akan digunakan selama mengakses sebuah website. Pengenal tersebut dapat didefinisikan oleh pembuat programer. Informasi pada session disimpan di server. Oleh sebab itu, session akan hilang secara otomatis jika user menutup browser atau meninggalkan (ditutup/ke luar) website tersebut. User akan mendapatkan pengenal yang baru ketika mengakses kembali halaman website yang sama.

Secara singkat cara kerjanya client REST (aplikasinya) akan menjalankan request ke server REST, ketika request terkirim ke server selanjutnya REST server akan memberikan respon. Kemudian client REST akan menampilkan responnya atau jika tidak, client akan melakukan pemrosesan yang lain. Respon dari REST Server ke REST client dapat diberikan dalam berbagai format, yaitu ada HTML, XML, Json, atau beberapa format yang lain. Namun format yang paling sering dipakai oleh para developer adalah Json, karena formatnya lebih mudah untuk digunakan dan dipelajari. Untuk bisa menggunakan REST ini, REST memiliki sebuah standarisasi yang harus dipakai dalam implementasinya, standarisasi REST ini menggunakan URL dan juga HTTP verb. Dengan menggunakan standarisasi yang menggunakan URL maka developer dapat melakukan beberapa operasi berdasarkan HTTP verbs. Secara teknis HTTP verbs ini berupa `$_SERVER['REQUEST_METHOD']`.

DAFTAR PUSTAKA

Damar, R. 2013. *Rancang Bangun REST Web Service untuk Perbandingan harga pengiriman dengan Metode Web Scrapping dan Pemanfaatan API*. Yogyakarta : STMIK AMIKOM YOGYAKARTA.

Indrawan, M.E. dan Adil, A. 2016. *Implementasi Restful Web Service One Chip Multi-Client untuk Mengoptimalkan Penjualan Pulsa All Operator*. Mataram : STMIK Bumigora Mataram.

Wiyono, D.S. Dan Wijayanto A. 2012. *Implementasi Rest Web Service Dengan Menggunakan Json Pada Aplikasi Mobile Enterprise Resource Planning*. Surakarta : Universitas Sebelas Maret.

Link Github :