

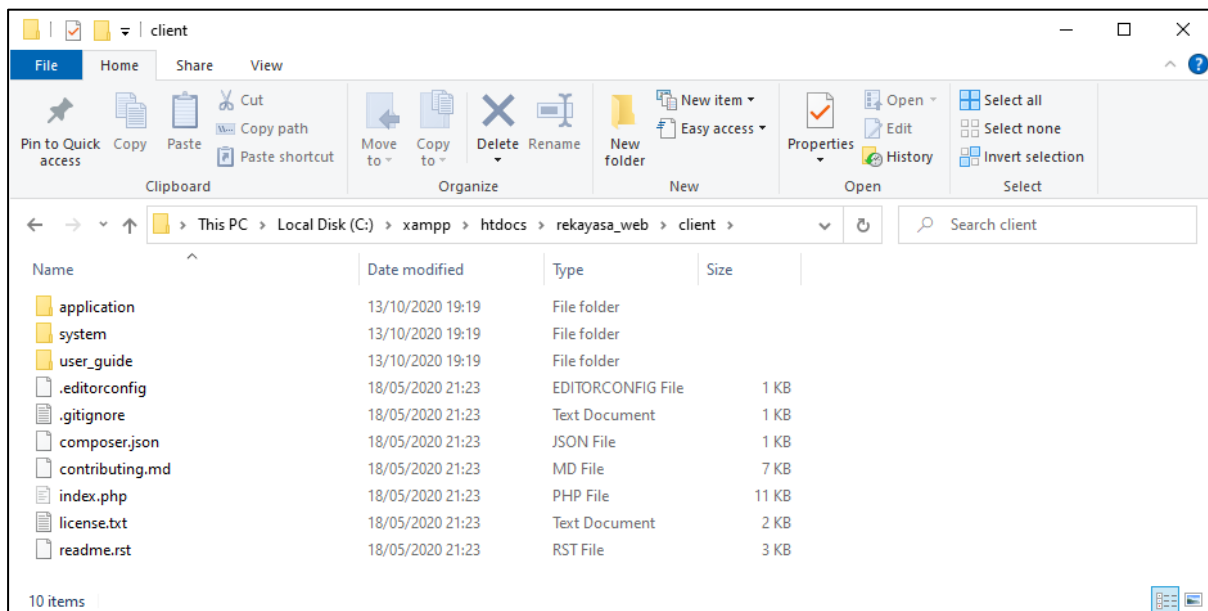
Praktikum Rekayasa Web
Membuat Client Server
Pertemuan ke 1

Disusun ulang oleh :
Galet Guntoro Setiaji

Yang harus disiapkan dalam praktikum pembuatan client yaitu download file-file didalam table dibawah ini :

No	Nama File	Url
1	Codeigniter 3	https://github.com/kuda-troya/RekayasaWeb/blob/master/Bahan%20Praktikum/CodeIgniter-3.1.10.zip
2	Codeigniter Curl Master	https://github.com/kuda-troya/RekayasaWeb/blob/master/Bahan%20Praktikum/codeigniter-curl-master.zip
3	Codeigniter Template Library	https://github.com/kuda-troya/RekayasaWeb/blob/master/Bahan%20Praktikum/ci_template_library.zip
4	Codeigniter RestServer Master	https://github.com/kuda-troya/RekayasaWeb/blob/master/Bahan%20Praktikum/codeigniter-restserver-master.zip
5	Bootstrap Versi 3	https://github.com/kuda-troya/RekayasaWeb/blob/master/Bahan%20Praktikum/bootstrap-3.3.7.zip
		https://github.com/kuda-troya/RekayasaWeb/blob/master/Bahan%20Praktikum/bootstrap-3.3.7-dist.zip

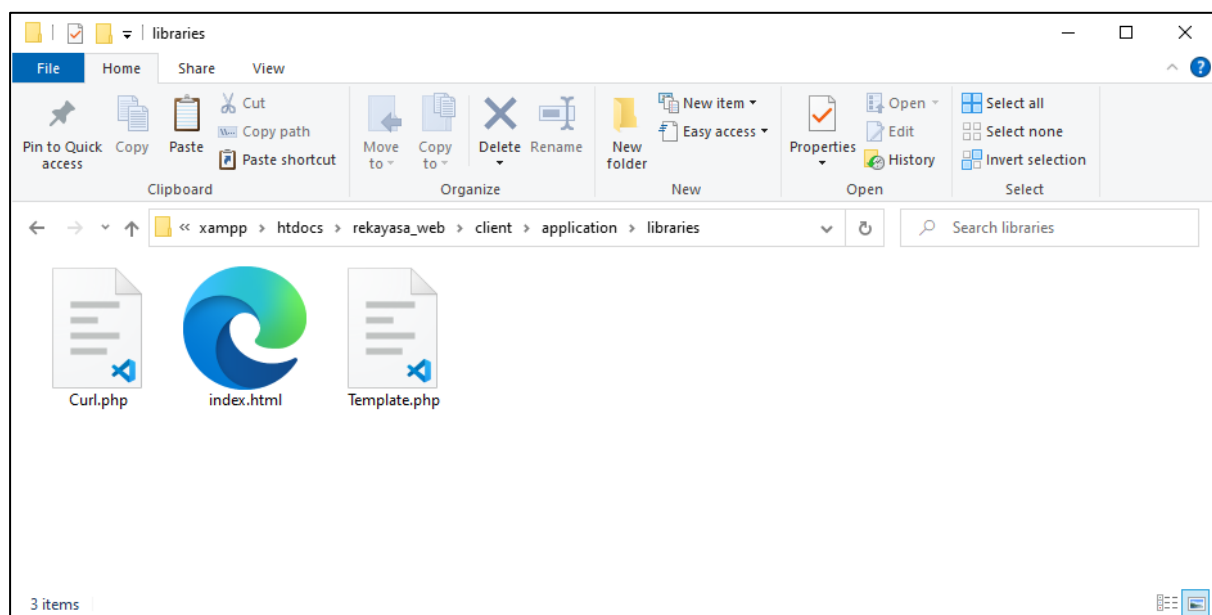
Buat folder **“client”** kemudian ekstrak **CodeIgniter-3.1.10.zip** dan copykan isi framework codeigniter 3 ke folder client, seperti gambar dibawah ini.



Langkah selanjutnya yaitu ekstrak codeigniter-curl-master.zip dan ci_template_library.zip, kemudian lakukan seperti arahan dari table dibawah ini:

Nama file	File zip name	Asal file	Copy ke
Curl.php	codeigniter-curl-master.zip	codeigniter-curl-master/ libraries/Curl.php	xampp/htdocs/rekayasa_web/client / application/ libraries/Curl.php
Template.php	ci_template_library. zip	ci_template_library /Template.php	xampp/htdocs/rekayasa_web/client / application/ libraries/Template.php

Hasilnya setelah dicopy akan menjadi seperti gambar dibawah ini :



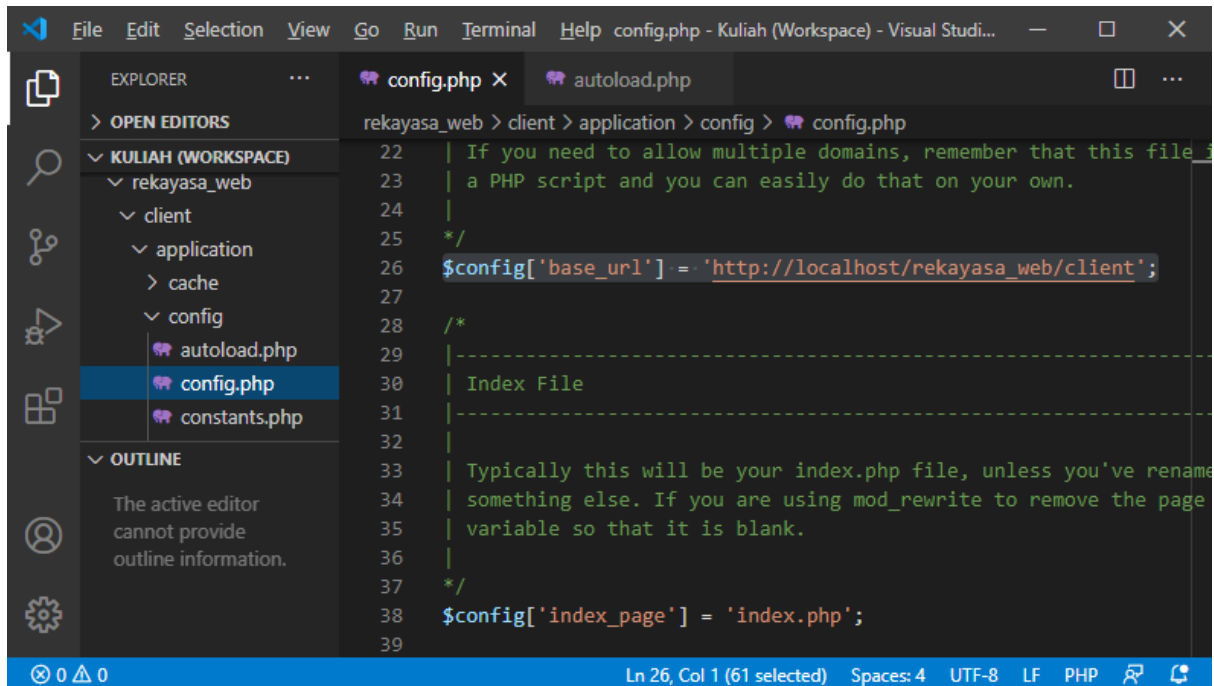
Kemudian ubahlah **“base_url”** yang berada di file **config.php** yang terletak di C:/xampp/htdocs/rekayasa_web/client/application/config/config.php seperti dibawah ini.

```
$config['base_url'] = '';
```

Diubah menjadi seperti dibawah ini

```
$config['base_url'] = 'http://localhost/rekayasa_web/client';
```

Seperti gambar dibawah ini:



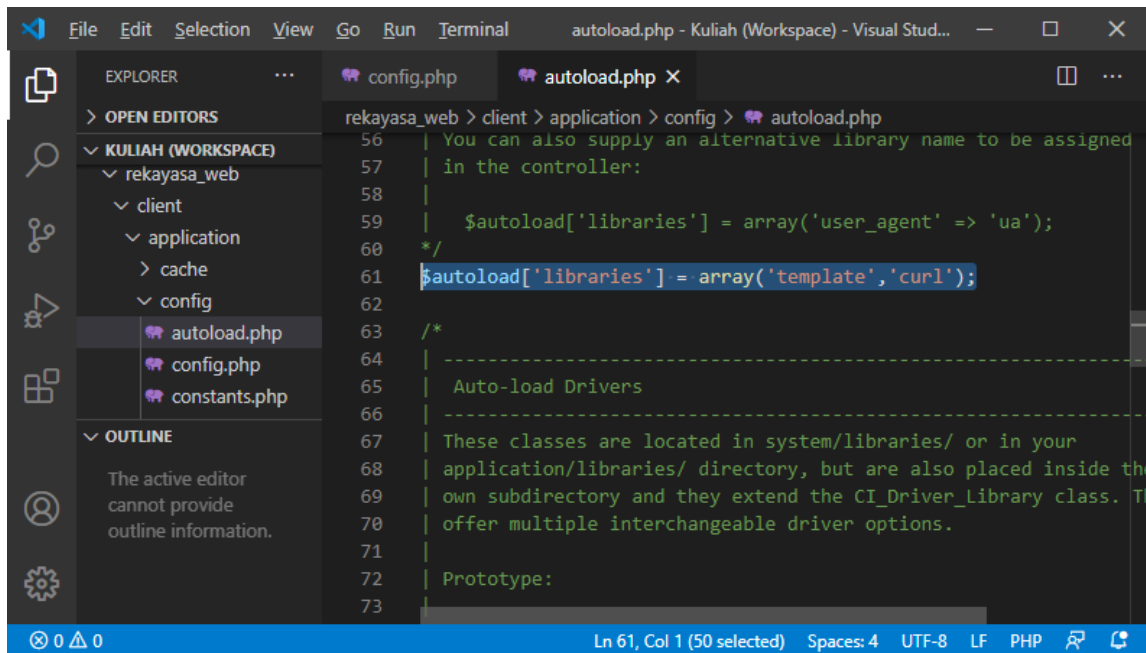
Kemudian kita tambahkan beberapa library di file autoload.php yang berada di C:/xampp/htdocs/rekayasa_web/client/application/config/autoload.php dari

```
$autoload['libraries'] = array('');
```

Menambahkan library **template** dan **curl** seperti dibawah ini

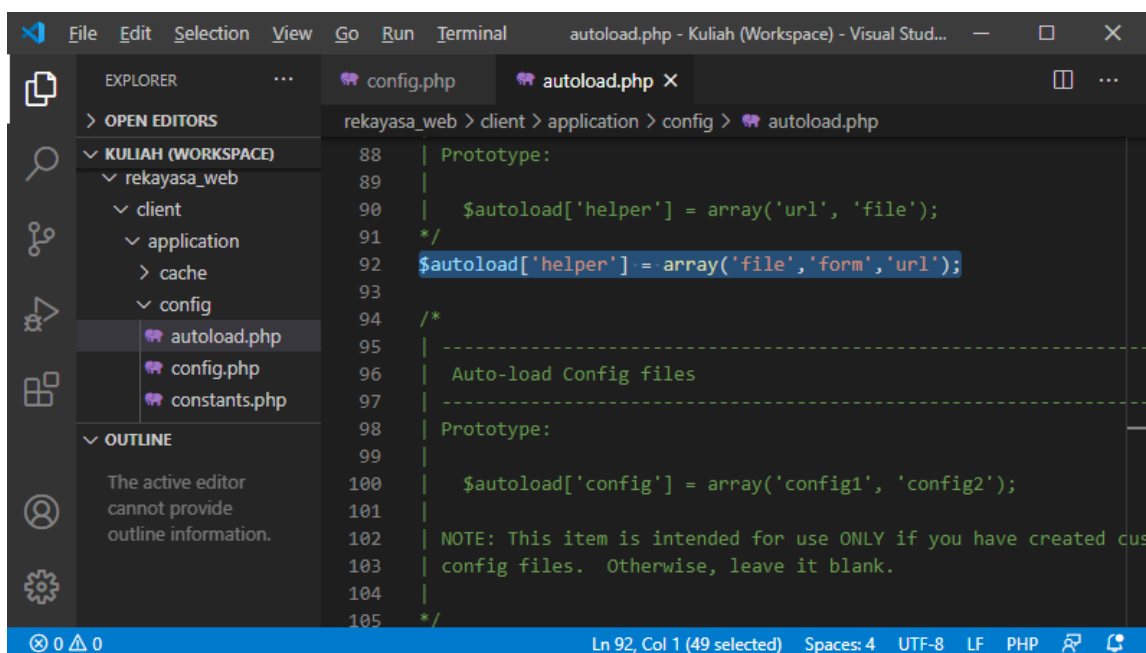
```
$autoload['libraries'] = array('template','curl');
```

Fungsi dari library ini nanti akan memanggil file template.php dan curl.php yang berada di folder libraries.



Masih di file autoload.php, tambahkan helper file, form dan url seperti dibawah ini

```
$autoload['helper'] = array('file','form','url');
```



Kemudian buat Script code untuk controller dengan nama **Customer.php**, dan letakkan di C:/xampp/htdocs/rekayasa_web/client/ application/controllers/Customer.php.

```
<?php
defined('BASEPATH') OR exit('No direct script access allowed');

class Customer extends CI_Controller{
    var $API="";

    function __construct(){
        parent::__construct();
        $this->API="http://localhost/rekayasa_web/server/index.php/api/customers/";
    }

    function index(){
        $data['customers']=$this->__execute($this->API,'get','');
        $this->template->load('template_master','customer/list',$data);
    }

    function add(){
        if(isset($_POST['submit'])){
            $params=array(
                'name'=>$this->input->post('name'),
                'phone'=>$this->input->post('phone'),
                'city'=>$this->input->post('city')
            );
            $this->__execute($this->API,'post',$params);
            redirect('customer');
        }else{
            $this->template->load('template_master','customer/add');
        }
    }

    function edit(){
        if(isset($_POST['submit'])){
            $params=array(
                'id'=>$this->input->post('id'),
                'name'=>$this->input->post('name'),
                'phone'=>$this->input->post('phone'),
                'city'=>$this->input->post('city')
            );
            $this->__execute($this->API,'put',$params);
            redirect('customer');
        }else{
            $id=$this->uri->segment(3);
            $data['customer']=$this->__execute($this->API.'id/'.$id,'get',array('id'=>$id));
            $this->template->load('template_master','customer/edit',$data);
        }
    }
}
```

```

    }
}

function delete(){
    $id=$this->uri->segment(3);
    $this->__execute($this->API,'delete',array('id'=>$id));
    redirect('customer');
}

function __execute($url,$method,$param){
    $this->curl->create($url);
    $this->curl->$method($param);
    $result=json_decode($this->curl->execute());
    return $result;
}
}

```

Penjelasan tentang code controller Customer.php :

Function __construct()

```

function __construct(){
    parent::__construct();
    $this->
    >API="http://localhost/rekayasa_web/server/index.php/api/customers/";
}

```

Fungsi pertama kali yang dijalankan dengan memanggil url webservice dari server **Api.php**.

Function index()

```

function index(){
    $data['customers']=$this->__execute($this->API,'get','');
    $this->template->load('template_master','customer/list',$data);
}

```

Fungsi index akan menampilkan tampilan data setelah fungsi **__construct** untuk mengecek webservice dari server dan dengan memanggil fungsi **__execute** untuk mengecek metode yang digunakan adalah **"GET"**. Dengan memanggil webservice server dari **Api.php** dengan function **customers_get()**.

Function add()

```

function add(){
    if(isset($_POST['submit'])){
        $params=array(
            'name'=>$this->input->post('name'),
            'phone'=>$this->input->post('phone'),
            'city'=>$this->input->post('city')
        );
        $this->__execute($this->API,'post',$params);
        redirect('customer');
    }
}

```

```

    }else{
        $this->template->load('template_master','customer/add');
    }
}

```

Fungsi add akan mengecek apabila tombol yang diklik tombol add, maka akan menampilkan form tambah. Namun apabila tombol submit button form maka akan mengeksekusi input dari form dan fungsi **__execute()** akan mengecek metode yang digunakan adalah **"POST"**. Kemudian **\$params** merupakan data dari post berbentuk array yang akan diinputkan atau insert. Dari **Api.php** dengan menggunakan **function customers_post()**.

Function edit()

```

function edit(){
    if(isset($_POST['submit'])){
        $params=array(
            'id'=>$this->input->post('id'),
            'name'=>$this->input->post('name'),
            'phone'=>$this->input->post('phone'),
            'city'=>$this->input->post('city')
        );
        $this->__execute($this->API,'put',$params);
        redirect('customer');
    }else{
        $id=$this->uri->segment(3);
        $data['customer']=$this->__execute($this->API.'id/'.$id,'get',array('id'=>$id));
        $this->template->load('template_master','customer/edit',$data);
    }
}

```

Didalam fungsi edit hampir sama dengan fungsion post, apabila di klik link edit makan akan mengarahkan ke form edit. Namun setelah dari form edit di ubah datanya kemudian di tekan tombol submit maka akan menjalankan perintah update dengan data array **\$params**, yang akan di eksekusi oleh **Api.php** dengan function **customers_put()**.

Function delete()

```

function delete(){
    $id=$this->uri->segment(3);
    $this->__execute($this->API,'delete',array('id'=>$id));
    redirect('customer');
}

```

Fungsi delete akan memangil service dari **Api.php** dengan fungsi **customer_delete()**, dan akan menghapus data tersebut.

Function __execute()

```
function __execute($url,$method,$param){  
    $this->curl->create($url);  
    $this->curl->$method($param);  
    $result=json_decode($this->curl->execute());  
    return $result;  
}
```

Fungsi execute merupakan fungsi yang akan menerapkan metode dari webservice (url api.php, metode yang digunakan, data yang dikirim), dengan librarie Curl.php.