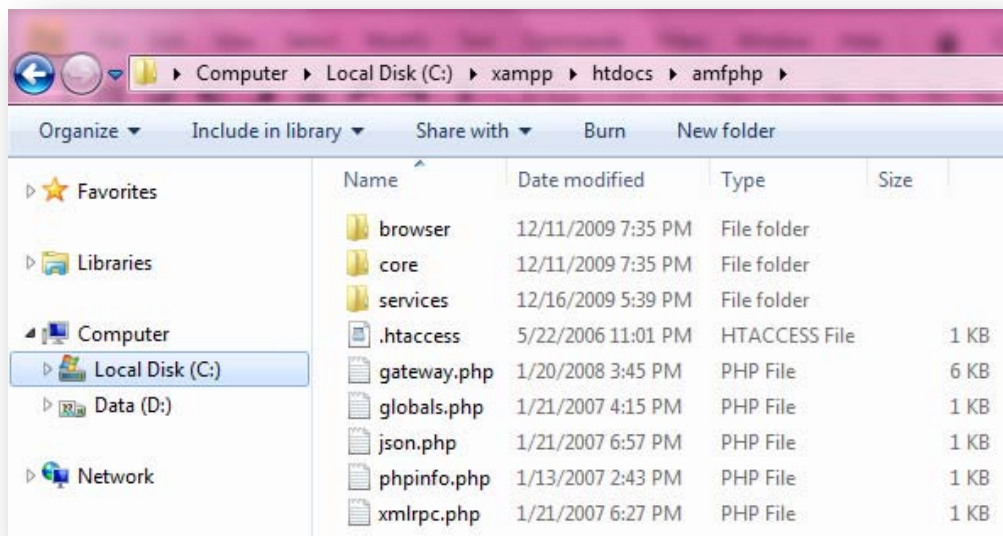


AMFPHP + Adobe Flex {Simple CRUD}

AMFPHP adalah Free Open Source implementasi PHP terhadap Action Message Format(AMF). AMFPHP sendiri bukan merupakan Framework, sehingga anda bebas menggunakan dengan cara anda. Versi terakhir dari amfphp ini merupakan versi 1.9 yang baru saja di release pada February 02, 2010. Pada tutorial ini saya akan memberikan Contoh membuat aplikasi sederhana menggunakan AMFPHP sebagai Backend dan Adobe Flex 3 sebagai Frontend, dengan database MySQL.

1. Download amfphp dari <http://waix.dl.sourceforge.net/project/amfphp/amfphp/amfphp%201.9.zip>
2. Extract file "amfphp 1.9.zip" yang sudah anda download ke folder dimana anda menyimpan sebagai document root. Misalkan C:\xampp\htdocs. Lalu ubah folder "amfphp 1.9" yang sudah anda extract tadi menjadi "amfphp". Sehingga anda akan mempunyai folder seperti ini "C:\xampp\htdocs\amfphp", yang jika anda buka folder tersebut, akan terlihat seperti gambar dibawah ini



3. Selanjutnya eksekusi perintah SQL dibawah ini :

```
CREATE DATABASE `adobeusergroup` ;  
USE adobeusergroup;
```

```
CREATE TABLE `adobeusergroup`.`user` (  
  `no` INT( 5 ) NOT NULL AUTO_INCREMENT PRIMARY KEY ,  
  `nama` VARCHAR( 55 ) NOT NULL ,  
  `alamat` VARCHAR( 80 ) NOT NULL ,  
  `email` VARCHAR( 40 ) NOT NULL ,  
  `telp` VARCHAR( 24 ) NOT NULL ,  
  UNIQUE ( `email` ) ) ENGINE = MYISAM COMMENT = 'Data User';
```

```
INSERT INTO `adobeusergroup`.`user` ( `no` , `nama` , `alamat` , `email`  
  , `telp` )
```

```
VALUES (NULL , 'Ahmad Fathi Hadi', 'Jl. Matraman Dalam II No:9b',  
'ahmadfathihadi@gmail.com', '0818084977xx' );
```

NB : Anda dapat menggunakan SQL Editor apapun, seperti phpmyadmin, MySqlFront, SQL
Yog,dll.

4. Selanjutnya kita harus menyiapkan VO (Value Object) yang akan dipakai oleh PHP. Prinsipnya adalah setiap Table akan menjadi Class, sedangkan field akan menjadi variable. Pada Table yang sudah kita miliki bernama User, dan memiliki field (no, nama, alamat, email dan telp). Sehingga akan terbentuk sebuah VO seperti berikut :

```
<?php  
class UserVO  
{  
    public $no;  
    public $nama;  
    public $alamat;  
    public $email;  
    public $telp;  
  
    var $_explicitType = "ifx.ifx.master.UserVO";  
}  
?>
```

Simpanlah file VO tersebut dengan nama UserVO.php pada folder sesuai var \$_explicitType di dalam folder Service/vo. Pada contoh kali ini berarti anda akan membuat folder C:\xampp\htdocs\amfphp\services\vo\ifx\ifx\master dan menyimpan file UserVO.php di dalamnya. Tujuan dibuatnya banyak folder seperti ifx/ifx adalah agar anda tidak bingung dan betrok jika anda bekerja dengan bayak service, sehingga anda tahu dimana service anda berada.

NB : Agar terstruktur dan tidak membingungkan saat bekerja tim, anda harus menyepakati aturan yang sudah umum, seperti menyimpan semua VO didalam folder service/vo

5. Selanjutnya buatlah Service yang akan mengolah Table tadi, seperti Select, Insert, dan Delete. Simpan code PHP dibawah ini di folder Service dengan nama User.php

```
<?php  
mysql_connect('localhost', 'root', '');  
mysql_select_db('adobeusergroup');  
require_once ('./vo/ifx/ifx/master/UserVO.php');  
  
class User  
{  
    public function getUser()  
    {  
        $query = "SELECT * from user";  
        $result = mysql_query($query);  
  
        $ret = array();  
        while ($row = mysql_fetch_object($result))
```

```

    {
        $tmp = new UserVO();
        $tmp->no = $row->no;
        $tmp->nama = $row->nama;
        $tmp->alamat = $row->alamat;
        $tmp->email = $row->email;
        $tmp->telp = $row->telp;
        $ret[] = $tmp;
    }
    mysql_free_result($result);
    return $ret;
}
public function saveUser($newUser)
    {
        $queryUser = "SELECT no from user where no='".$newUser->no.'";
        $resultUser = mysql_query($queryUser);
        $ketUser = mysql_fetch_array($resultUser);
        if($ketUser != NULL)
        {
            $query = "UPDATE user SET nama = '".$newUser->nama.'," .
                "alamat = '".$newUser->alamat.'," .
                "email = '".$newUser->email.'," .
                "telp = '".$newUser->telp.'";" .
                "WHERE no = '".$newUser->no.'";";
            $result = mysql_query($query);
            return NULL;
        }
        else
        {
            $query = "INSERT INTO user" .
                "(no," .
                "nama," .
                "alamat," .
                "email," .
                "telp)" .
                "VALUES" .
                "(".$newUser->no.'," .
                "".$newUser->nama.'," .
                "".$newUser->alamat.'," .
                "".$newUser->email.'," .
                "".$newUser->telp.'");";
            $result = mysql_query($query);
            return NULL;
        }
    }

    public function deleteUser($selectedUser)
    {

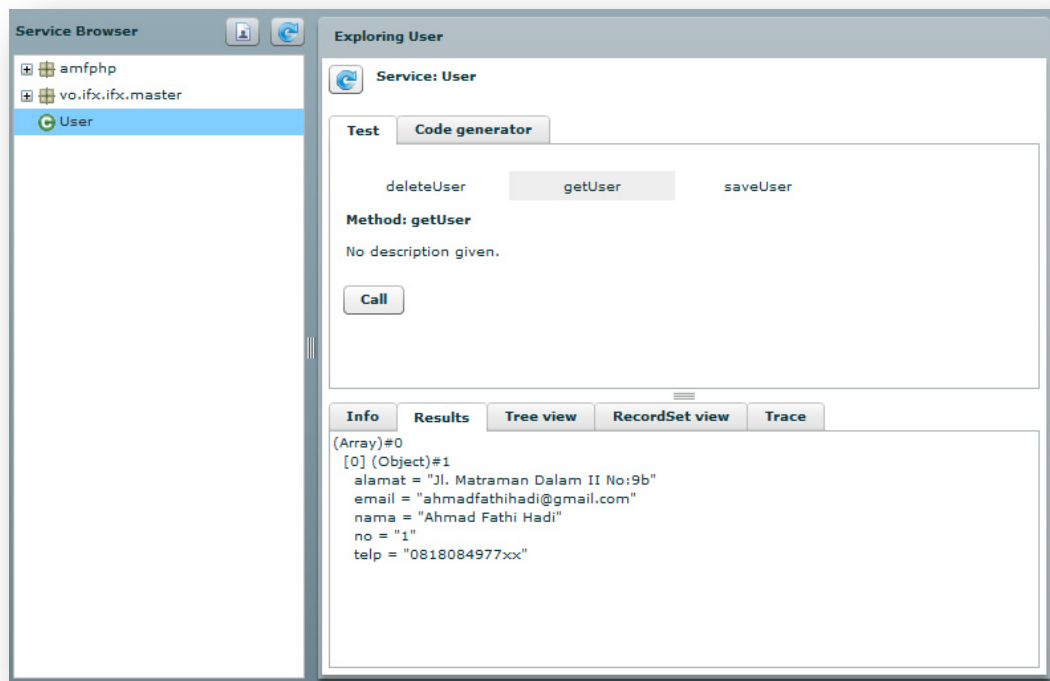
```

```

$query = "DELETE FROM user WHERE no ='$selectedUser";
$result = mysql_query($query);
return NULL;
}
}
?>

```

Sebelum maju kelangkah selanjutnya, silahkan anda cek service yang anda buat, apakah sudah berjalan dengan baik. Silahkan buka url pada browser anda <http://localhost/amfphp/browser/> Apakah sudah terdapat Class "User" pada service browser ?, apakah jika anda plih, ada dapat mengakses method yang ada seperti deleteUser, getUser, dan saveUser ?



6. Sekarang buatlah project baru pada Adobe Flex Builder 3 dengan nama AMFPHP Test
7. Langkah pertama adalah menyiapkan VO yang akan dipakai di Project Flex, sama seperti VO pada PHP, kita akan membuat sama, silahkan buat folder ifx/ifx/master pada folder scr di Flex Builder, lalu buatlah Class ActionScript baru dengan nama UserVO.as

```

package ifx.ifx.master
{
    [RemoteClass(alias="ifx.ifx.master.UserVO")]

    [Bindable]
    public class UserVO
    {

```

```

        public var no:Number;
        public var nama:String;
        public var alamat:String;
        public var email:String;
        public var telp:String;
    }
}

```

8. Sekarang pada main Application silahkan ketikkan code berikut :

```

<?xml version="1.0" encoding="utf-8"?>
<mx:Application xmlns:mx="http://www.adobe.com/2006/mxml" layout="absolute"
creationComplete="init();">
    <mx:Script>
        <![CDATA[
            import ifx.ifx.master.UserVO;
            import mx.rpc.events.ResultEvent;

            [Bindable]
            private var no:Number = new Number();
            [Bindable]
            private var dataUser:Array = new Array();

            private function init():void
            {
                myUser.getUser.addEventListener("result",getUserHandler);
                myUser.getUser();
            }

            private function getUserHandler(event:ResultEvent):void
            {
                dataUser = event.result as Array;
            }

            private function insert():void
            {
                var newUser:UserVO = new UserVO();
                newUser.nama = namaInput.text;
                newUser.alamat = alamatInput.text;
                newUser.email = emailInput.text;
                newUser.telp = telpInput.text;
                myUser.saveUser(newUser);
                myUser.getUser();
                cancelHandler();
            }

            private function selectHandler():void
            {
                noLabel.text = dataGridUser.selectedItem.no;
                namaInput.text = dataGridUser.selectedItem.nama;
                alamatInput.text = dataGridUser.selectedItem.alamat;
                emailInput.text = dataGridUser.selectedItem.email;
                telpInput.text = dataGridUser.selectedItem.telp;
            }

            private function cancelHandler():void
            {
                noLabel.text = "";
            }
        ]]>
    </mx:Script>

```

```

        namaInput.text = "";
        alamatInput.text = "";
        emailInput.text = "";
        telpInput.text = "";
        dataGridUser.selectedItem = null;
    }
    private function deleteUser():void
    {
        var selectedUser = noLabel.text;
        myUser.deleteUser(selectedUser);
        myUser.getUser();
    }
    ]]>
</mx:Script>
<mx:DataGrid x="10" y="243" width="627" dataProvider="{dataUser}"
id="dataGridUser" click="selectHandler();">
    <mx:columns>
        <mx:DataGridColumn headerText="No" dataField="no"
width="40"/>
        <mx:DataGridColumn headerText="Nama" dataField="nama"/>
        <mx:DataGridColumn headerText="Alamat" dataField="alamat"/>
        <mx:DataGridColumn headerText="Email" dataField="email"/>
        <mx:DataGridColumn headerText="telp" dataField="telp"/>
    </mx:columns>
</mx:DataGrid>
<mx:Form x="10" y="10" width="410" height="225"
horizontalScrollPolicy="off" verticalScrollPolicy="off" id="myForm">
    <mx:Label text="No : {no}" id="noLabel"/>
    <mx:FormItem label="Nama :" required="true">
        <mx:TextInput width="260" id="namaInput"/>
    </mx:FormItem>
    <mx:FormItem label="Alamat :" required="true">
        <mx:TextArea width="260" height="62" id="alamatInput"/>
    </mx:FormItem>
    <mx:FormItem label="Email :" required="true">
        <mx:TextInput width="260" id="emailInput"/>
    </mx:FormItem>
    <mx:FormItem label="Telp :" required="true">
        <mx:TextInput width="260" id="telpInput"/>
    </mx:FormItem>
    <mx:HBox width="100%" horizontalAlign="center">
        <mx:Button label="Save" click="insert()"/>
        <mx:Button label="Delete" click="deleteUser()"/>
        <mx:Button label="Cancel" click="cancelHandler();"/>
    </mx:HBox>
</mx:Form>
<mx:RemoteObject id="myUser" destination="User" source="User"
endpoint="http://localhost/amfphp/gateway.php" showBusyCursor="true"/>
</mx:Application>

```

Anda dapat mendownload Project ini di pada blog.fathihadi.net

Dibuat oleh Ahmad Fathi Hadi