**TEKNIK INFORMATIKA**

**RESUME TUGAS WEB PROGRAMING 2**

**PHP AJAX JAVASCRIPT JQUERY**



Nama Dosen : M. Muzaqi, S.Kom

Disusun Oleh :

Nama : Septiyan Adi Nugroho

Kelas : 5D

Nim : 17090105

**POLITEKNIK HARAPAN BERSAMA KOTA TEGAL**

Pengertian

**JSON** (JavaScript Object Notation) adalah format pertukaran data yang ringan, mudah dibaca dan ditulis oleh manusia, serta mudah diterjemahkan dan dibuat (generate) oleh komputer. Format ini dibuat berdasarkan bagian dari Bahasa Pemprograman JavaScript, Standar ECMA-262 Edisi ke-3 - Desember 1999. Format JSON sering digunakan untuk mentransmisikan data terstruktur melalui suatu koneksi jaringan pada suatu proses yang disebut [serialisasi](https://id.wikipedia.org/wiki/Serialisasi" \o "Serialisasi). Aplikasi utamanya adalah pada pemrograman [aplikasi web](https://id.wikipedia.org/wiki/Aplikasi_web" \o "Aplikasi web) [AJAX](https://id.wikipedia.org/wiki/AJAX) dengan berperan sebagai alternatif terhadap penggunaan tradisional format [XML](https://id.wikipedia.org/wiki/XML).

Contoh berikut menunjukkan representasi JSON untuk suatu objek yang mendeskripsikan seseorang. :

{

"namaDepan": "adi",

"namaBelakang": "nugroho",

"alamat": {

"namaJalan": "Jl. Sudirman 15A",

"kota": "tegal",

"provinsi": "jawa tengah",

"kodePos": 54219

},

"nomerTelepon": [

"021 555-1234",

"021 555-4567"

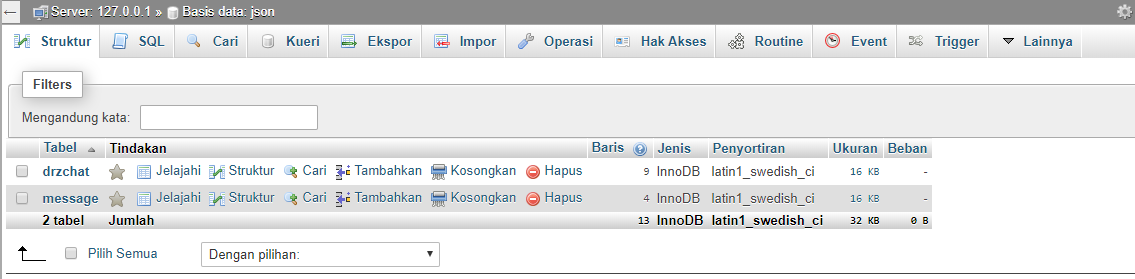
]

}

1. JSON AJAX WEB CHAT

Aplikasi Ajax yang menggunakan Json Sebagai output yang diberikan oleh client.

Berikut ini adalah Database dari Json Ajax web chat :



1. Berikut ini Adalah tampilan Ajax Web Chat

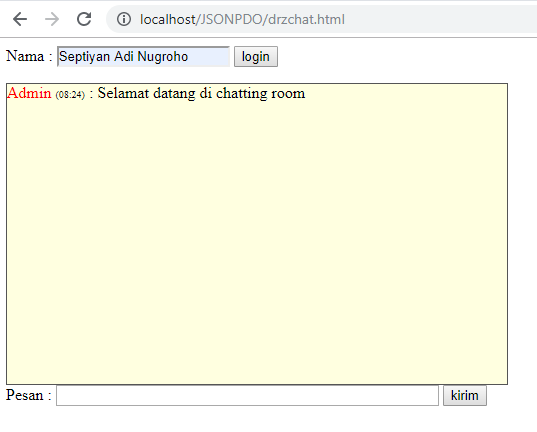
Ini adalah tampilan layout HTML dari Ajax wheb chat . Pertama-tama user harus memasukan nama mereka di suatu teks input. Setelah kita menekan tombol login, maka segera dijalankan fungsi ambil pesan.



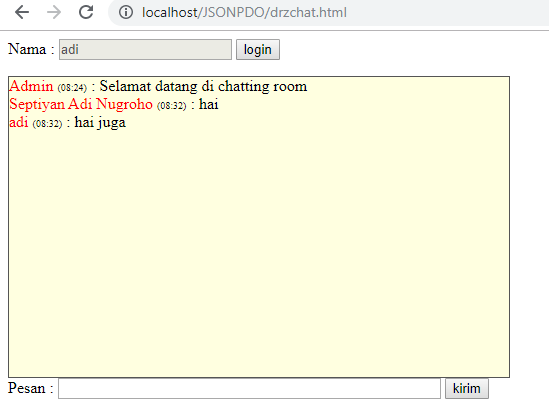
1. Berikut ini adalah tampilan saat user sudah login

Setelah menekan tombol Login akan otomatis melakukan request ke server untuk mengambil data dari server ambilchat.php .

Data yang diambil dari server berupa teks dengan format JSON.



Tampilan penggunaan Ajax Web Chat .



PENJELASAN CODDING

1. Berikut Ini adalah Penjelasan codding Database :

Pertama-tama mari kita buat tabel database dengan struktur tabel sebagai berikut :

CREATE TABLE ‘drzchat’(

‘nomor’ int(3) NOT NULL auto\_increment,

‘nama’ varchar(20) NOT NULL,

‘pesan’ varchar(200) NOT NULL,

‘waktu’ varchar(10) NOT NULL,

PRIMARY KEY (‘nomor’)

);

Kemudian mari kita buat tampilan layout HTML nya dan program ajax di sisi client :

<!DOCTYPE html>

<html>

<head>

<title>DRZ Chat 1.0</title>

<script>

var ajaxku = buatAjax();

var tnama = 0;

var pesanakhir = 0;

var timer;

function taruhNama(){

if(tnama==0){

document.getElementById("nama").disabled = "true";

tnama = 1;

}else{

document.getElementById("nama").disabled = "";

tnama = 0;

}

ambilPesan();

}

function buatAjax(){

if(window.XMLHttpRequest){

return new XMLHttpRequest();

}else if(window.ActiveXObject){

return new ActiveXObject("Microsoft.XMLHTTP");

}

}

function ambilPesan(){

namaku = document.getElementById("nama").value

if(ajaxku.readyState == 4 || ajaxku.readyState == 0){

ajaxku.open("GET","ambilchat.php?akhir="+pesanakhir+"&nama="+namaku+"&sid="+Math.random(),true);

ajaxku.onreadystatechange = aturAmbilPesan;

ajaxku.send(null);

}

}

function aturAmbilPesan(){

if(ajaxku.readyState == 4){

var chat\_div = document.getElementById("div\_chat");

var data = eval ("(" + ajaxku.responseText + ")");

for(i=0; i < data.messages.pesan.length;i++){

chat\_div.innerHTML += "<font color=red>"+data.messages.pesan[i].nama+"</font> ";

chat\_div.innerHTML += "<font size=1>("+data.messages.pesan[i].waktu+")</font> ";

chat\_div.innerHTML += " : "+data.messages.pesan[i].teks+"<br>";

chat\_div.scrollTop = chat\_div.scrollHeight;

pesanakhir = data.messages.pesan[i].id;

}

}

timer = setTimeout("ambilPesan()",10);

}

function kirimPesan(){

pesannya = document.getElementById("pesan").value

namaku = document.getElementById("nama").value

if(pesannya != "" && document.getElementById("nama").value !=""){

ajaxku.open("GET","ambilchat.php?akhir="+pesanakhir+"&nama="+namaku+"&pesan="+pesannya+"&sid="+Math.random(),true);

ajaxku.onreadystatechange = aturAmbilPesan;

ajaxku.send(null);

document.getElementById("pesan").value = "";

}else{

alert("Nama atau pesan masih kosong");

}

}

function aturKirimPesan(){

clearInterval(timer);

ambilPesan();

}

function blockSubmit(){

kirimPesan();

return false;

}

</script>

</head>

<body>

Nama : <input type=text name=nama id=nama>

<input type=button value=login id=tmbl\_login onclick="taruhNama()"><p>

<div id="div\_chat" style="height: 300px; width: 500px; overflow: auto; background-color: lightyellow; border: 1px solid #555555;">

</div>

<form onSubmit="return blockSubmit();">

Pesan : <input type=text name=pesan id=pesan size=50>

<input type=button value="kirim" onclick="kirimPesan()">

</form>

</body>

</html>

Pada program chat sisi client di atas, pertama-tama user harus memasukan nama mereka di suatu teks input. Setelah kita menekan tombol login, maka segera dijalankan fungsi ambilpesan().

Data yang diambil dari server berupa teks dengan format JSON sebagi berikut :

{“messages” :

{“pesan” : [

{“id” : “35”,

“nama” : “Batman”,

“teks” : “hallo Nurmi”,

“waktu”:”16:31”},

{“Id” : “36”,

“nama”:”Nurmi”,

“teks”:”halo juga”,

“waktu”:”16:31”},

{“id”: “37”,

“nama”:”desrizal”,

“teks”:”hai apa kabar”,

“waktu”:”16:47”}

]

}

}

Format JSON tersebut diambil dari server, yang dimasukkan ke dalam variable data, kemudian kita gunakan eval untuk mengubah JSON string menjadi objek javascript

Var data = eval (“(“+ajaxku.responseText+”)”);

Kemudian Objek JSON yang telah di parse di atas kita masukan ke dalam <div>

For (i=0;i<data.messages.pesan.length;i++){

Chat\_div.innerHTML += “<font color=red>”+data.messages.pesan[i].nama+”</font>”;

Chat\_div.innerHTML += “<font size =1>(“+data.messages.pesan[i].teks+”<br>”;

Chat\_div.innerHTML +=” : “+data.messages.pesan[i].teks+”<br>”;

Chat\_div.scrollTop = chat\_div.scrollHeight;

Pesanakhir = data.messages.pesan[i].id;

Supaya ajax hanya mengambil data dari pesan yang terakhir maka kita juga mengirimkan nomor pesan id terakhir

Pesanakhir = data.messages.pesan[i].id;

Supaya ajax mengambil data tiap satu detik maka kita gunakan fungsi setTimeout()

Timer = setTimeout(“ambilpesan()”,1000);

Biasanya kalau kita chat, kita ketik pesanya di input teks lalu kita tekan ENTER, biasanya kalua form di tekan ENTER akan otomatis ter-submit dan halaman akan bertukar, Untuk mencegah halaman ter-submit ketika di tekan ENTER, maka kita blok menggunakan fungsi berikut :

Function aturKirimPesan(){

ClearInterval(timer);

Ambilpesan();

}

Fungsi di atas dipanggil ketika ada event onSubmit.

<form onSubmit=”return blockSubmit();”>

Sekarang mari kita lihat kode PHP di sisi server, di mana di program PHP ini kita membuat output data berformat JSON

Codding Ambilchat.php

<?php

$con = new PDO("mysql:dbname=json;host=localhost", "root", "");

if (isset($\_GET['nama'])) {

@$nama = $\_GET['nama'];

}

if (isset($\_GET['pesan'])) {

$pesan = $\_GET['pesan'];

}

$waktu = date("H:i");

if (isset($\_GET['akhir'])) {

@$akhir = $\_GET['akhir'];

}

$json = '{"messages": {';

if (@$akhir == 0) {

$nomor = $con->query("select nomor from drzchat order by nomor desc limit 1");

$nomor->execute();

$n = $nomor->fetch(PDO::FETCH\_ASSOC);

$no = $n['nomor'] + 1;

$json .= '"pesan":[ {';

$json .= '"id":"' . $no . '",

"nama": "Admin",

"teks": "Selamat datang di chatting room",

"waktu": "' . $waktu . '"

}]';

$masuk = $con->query("insert into drzchat values(null, 'Admin', '$nama bergabung dalam chat', '$waktu')");

} else {

if (isset($pesan)) {

$masuk = $con->query("insert into drzchat values(null, '$nama','$pesan','$waktu')");

}

$query = $con->query("select \* from drzchat where nomor > $akhir");

$json .= '"pesan":[ ';

foreach ($query->fetchAll(PDO::FETCH\_ASSOC) as $x) {

$json .= '{';

$json .= '"id": "' . $x['nomor'] . '",

"nama": "' . htmlspecialchars($x['nama']) . '",

"teks": "' . htmlspecialchars($x['pesan']) . '",

"waktu": "' . $x['waktu'] . '"

},';

}

$json = substr($json, 0, strlen($json) - 1);

$json .= ']';

}

$json .= '}}';

echo $json;

logika dari program diatas adalah :

1. Lakukan koneksi ke server Mysql
2. Ambil nilai-nilai parameter yang diberikan oleh client dengan metode GET
3. Cek apakah user pertama kali login dengan melihat nilai variable $akhir, jika bernilai 0 artinya adalah user chat pertama kali, lalu kita masukan ke dalam database supaya user lain mengetahui ada user lain yang login
4. Apabila nilai $akhir besar dari 0 , artinya user bukan masuk untuk pertama kalinya, berikan data-data dari database setelah yang terakhir dia dapat, yang kita dapat dari variable $akhir

$query = mysql\_query(“select\*from drzchaat where nomor> $akhir”);

1. Output yang kita buat dengan format json, untuk mengirim data ke client cukup melakukan echo $json;.