

Cara Membuat API Login dan Registrasi Menggunakan PHP

BY: FARIZDOTID | ON: MAY 13, 2017 | IN: API |
TAGGED: API, BACKEND PHP | WITH: 1 COMMENT |

FOLLOW ME ON:



CARI ARTIKEL

🔍 Type Search Term ..



Cara Membuat API Login dan Registrasi Menggunakan PHP –

Halo teman-teman dev. Di kesempatan nulis artikel kali ini saya ingin membuat sebuah API sederhana yang terkoneksi dengan server lokal atau biasa kita sebut localhost. API itu apa sih ? API ini berupa singkatan dari *Application Programming Interface*, Kegunaannya apa ? Kegunaannya API ini jika dalam kasus develop aplikasi android adalah sebagai penghubung antara android dan server. API ini berada di tengah-tengah antara android dan server, Yang dimana android mengirim ke API lalu API akan meneruskan ke server. Jadi kurang lebih seperti itu

ARTIKEL TERBARU

- [Cara Mengambil Nilai di Radio Button Android Dengan Mudah](#)

pemahaman mengenai API yang saya tau. Nah sekarang kita akan membuat sebuah API yang berfungsi untuk user login dan juga user bisa men-registrasi. Yuk simak Cara Membuat API Login dan Registrasi Menggunakan PHP.



Pertama,

Download terlebih dahulu xampp. Jika agan sudah punya silahkan skip step pertama ini.

[Download xampp](#)

Kedua,

Buatlah database dengan nama terserah agan. Contoh disini dengan nama database db_mahasiswa.

Ketiga,

Buatlah tabel dengan nama terserah agan. Contoh disini dengan nama tbl_user dan berisi field-field id, unique_id, nama, email, encrypted_password, salt. Agan bisa mengcopy – paste code sql berikut ini :

```
1 create table tbl_user(  
2     id int(11) primary key auto_increment,  
3     unique_id varchar(23) not null unique,  
4     nama varchar(50) not null,  
5     email varchar(100) not null unique,  
6     encrypted_password varchar(80) not null,  
7     salt varchar(10) not null  
8 );
```

- [Cara Menambahkan dan Menggunakan Library Butterknife](#)
- [Tutorial Login dan Register Menggunakan Retrofit2 dan API Server](#)
- [Cara Membuat API Login dan Registrasi Menggunakan PHP](#)
- [Referensi Library Textview/EditText Android](#)
- [Referensi Library ImageView Android](#)
- [Jasa 200.000 Backlink Berkualitas Murah untuk Website, Kontes Seo dan Youtube](#)
- [Beberapa Aplikasi GPS Offline Android Terbaik](#)

KOMENTAR TERBARU

- [farizdotid on Cara Membuat CRUD Database Sqlite dan RecyclerView Android Part 2](#)
- [farizdotid on Cara Mengambil Nilai di Radio Button Android Dengan Mudah](#)
- [irgi on Cara Membuat CRUD Database Sqlite dan RecyclerView Android Part 2](#)
- [irgi on Cara Membuat CRUD Database Sqlite dan RecyclerView Android Part 2](#)

Keempat,

Buat folder dengan nama terserah agan di dalam folder htdocs.
Contoh disini saya membuat folder dengan nama mahasiswa.

Kelima,

Didalam folder mahasiswa tersebut buat folder kembali yang bernama *include*. Isi folder include disini ialah untuk mengisi file-file helper saja. Jadi kurang lebih menjadi direktori seperti ini :

-htdocs

 -mahasiswa

 —include

Keenam,

Didalam folder include buatlah file dengan nama **Config.php** .
Kelas ini berisi variabel-variabel informasi server kita.

```
1 <?php
2 define("DB_HOST", "localhost");
3 define("DB_USER", "root");
4 define("DB_PASSWORD", "");
5 define("DB_DATABASE", "db_mahasiswa");
6 ?>
```

Ketujuh,

Didalam folder include buatlah file dengan nama **DB_Connect.php** . Kelas ini berfungsi untuk menghubungkan kita ke server.

```
1 <?php
2 class DB_Connect {
3     private $conn;
4
5     // koneksi ke database
6     public function connect() {
7         require_once 'include/Config.php';
8
9         // koneksi ke mysql database
10        $this->conn = new mysqli(DB_HOST, DB_USER, DB_
11
12        // return database handler
13        return $this->conn;
14    }
15 }
16
17 ?>
```

Kedelapan,

Didalam folder include buatlah file dengan nama **DB_Function.php** . Kelas ini berfungsi untuk menyimpan suatu fungsi yang didalam fungsi tersebut akan mengirim data ke server.

```
1 <?php
2
3 class DB_Functions {
4
5     private $conn;
6
7     // constructor
8     function __construct() {
9         require_once 'DB_Connect.php';
10        // koneksi ke database
11        $db = new Db_Connect();
12        $this->conn = $db->connect();
13    }
14
15    // destructor
16    function __destruct() {
17
18    }
19
20    public function simpanUser($nama, $email, $password
21        $uuid = uniqid('', true);
22        $hash = $this->hashSHA($password);
23        $encrypted_password = $hash["encrypted"]; //
24        $salt = $hash["salt"]; // salt
25
26        $stmt = $this->conn->prepare("INSERT INTO tbl
27        $stmt->bind_param("sssss", $uuid, $nama, $emc
```

```

28     $result = $stmt->execute();
29     $stmt->close();
30
31     // cek jika sudah sukses
32     if ($result) {
33         $stmt = $this->conn->prepare("SELECT * FROM t
34         $stmt->bind_param("s", $email);
35         $stmt->execute();
36         $user = $stmt->get_result()->fetch_assoc();
37         $stmt->close();
38
39         return $user;
40     } else {
41         return false;
42     }
43 }
44
45 /**
46  * Get user berdasarkan email dan password
47  */
48 public function getUserByEmailAndPassword($email,
49
50     $stmt = $this->conn->prepare("SELECT * FROM t
51
52     $stmt->bind_param("s", $email);
53
54     if ($stmt->execute()) {
55         $user = $stmt->get_result()->fetch_assoc();
56         $stmt->close();
57
58         // verifikasi password user
59         $salt = $user['salt'];
60         $encrypted_password = $user['encrypted_password'];
61         $hash = $this->checkhashSHA($salt, $password);
62         // cek password jika sesuai
63         if ($encrypted_password == $hash) {
64             // autentikasi user berhasil
65             return $user;
66         }
67     } else {
68         return NULL;
69     }
70 }
71
72 /**
73  * Cek User ada atau tidak
74  */
75 public function isUserExisted($email) {
76     $stmt = $this->conn->prepare("SELECT email fr
77
78     $stmt->bind_param("s", $email);
79
80     $stmt->execute();
81
82     $stmt->store_result();
83
84     if ($stmt->num_rows > 0) {
85         // user telah ada
86         $stmt->close();
87         return true;
88     } else {
89         // user belum ada
90         $stmt->close();
91         return false;
92     }
93 }
94
95 /**
96  * Encrypting password

```

```

97      * @param password
98      * returns salt and encrypted password
99      */
100     public function hashSSHA($password) {
101
102         $salt = sha1(rand());
103         $salt = substr($salt, 0, 10);
104         $encrypted = base64_encode(sha1($password . $salt));
105         $hash = array("salt" => $salt, "encrypted" => $encrypted);
106         return $hash;
107     }
108
109     /**
110      * Decrypting password
111      * @param salt, password
112      * returns hash string
113      */
114     public function checkhashSSHA($salt, $password) {
115
116         $hash = base64_encode(sha1($password . $salt));
117
118         return $hash;
119     }
120 }
121 }
122
123 ?>

```

Kesembilan,

Didalam folder mahasiswa buatlah filde dengan nama **login.php** . Kelas ini berfungsi untuk login dengan parameternya email dan password.

```

1  <?php
2  require_once 'include/DB_Functions.php';
3  $db = new DB_Functions();
4
5  // json response array
6  $response = array("error" => FALSE);
7
8  if (isset($_POST['email']) && isset($_POST['password'])) {
9
10     // menerima parameter POST ( email dan password )
11     $email = $_POST['email'];
12     $password = $_POST['password'];
13
14     // get the user by email and password
15     // get user berdasarkan email dan password
16     $user = $db->getUserByEmailAndPassword($email, $password);
17
18     if ($user != false) {
19         // user ditemukan
20         $response["error"] = FALSE;
21         $response["uid"] = $user["unique_id"];
22         $response["user"]["nama"] = $user["nama"];
23         $response["user"]["email"] = $user["email"];
24         echo json_encode($response);
25     } else {
26         // user tidak ditemukan password/email salah

```

```

27     $response["error"] = TRUE;
28     $response["error_msg"] = "Login gagal. Passwor
29     echo json_encode($response);
30 }
31 } else {
32     $response["error"] = TRUE;
33     $response["error_msg"] = "Parameter (email atau p
34     echo json_encode($response);
35 }
36 ?>

```

Kesepuluh,

Didalam folder mahasiswa buatlah file dengan nama **register.php** . Kelas ini berfungsi untuk menerima data dari Android yang lalu diteruskan menyimpan data ke server.

```

1 <?php
2
3 require_once 'include/DB_Functions.php';
4 $db = new DB_Functions();
5
6 // json response array
7 $response = array("error" => FALSE);
8
9 if (isset($_POST['nama']) && isset($_POST['email']) &&
10
11     // menerima parameter POST ( nama, email, password
12     $nama = $_POST['nama'];
13     $email = $_POST['email'];
14     $password = $_POST['password'];
15
16     // Cek jika user ada dengan email yang sama
17     if ($db->isUserExisted($email)) {
18         // user telah ada
19         $response["error"] = TRUE;
20         $response["error_msg"] = "User telah ada denga
21         echo json_encode($response);
22     } else {
23         // buat user baru
24         $user = $db->simpanUser($nama, $email, $password
25         if ($user) {
26             // simpan user berhasil
27             $response["error"] = FALSE;
28             $response["uid"] = $user["unique_id"];
29             $response["user"]["nama"] = $user["nama"];
30             $response["user"]["email"] = $user["email"];
31             echo json_encode($response);
32         } else {
33             // gagal menyimpan user
34             $response["error"] = TRUE;
35             $response["error_msg"] = "Terjadi kesalahan
36             echo json_encode($response);
37         }
38     }
39 } else {
40     $response["error"] = TRUE;
41     $response["error_msg"] = "Parameter (nama, email,
42     echo json_encode($response);
43 }

```

Kesebelas,

Silahkan coba terlebih dahulu API Login dan Registrasi tersebut di Aplikasi POSTMAN dengan method **POST**. Jika agan belum punya Aplikasi Postman agan bisa mendownloadnya disini ([download postman](#)).

API Login : <http://localhost/mahasiswa/login.php>

API Register : <http://localhost/mahasiswa/register.php>

Respon API Login :

Login berhasil

```
1 {
2   "error": false,
3   "uid": "5916cf58792ac2.94574612",
4   "user": {
5     "nama": "fariz",
6     "email": "farizdotid@mail.com"
7   }
8 }
```

Login gagal

```
1 {
2   "error": true,
3   "error_msg": "Login gagal. Password/Email salah"
4 }
```

Respon API Register :

Register berhasil

```
1 {
2   "error": false,
3   "uid": "5916cf58792ac2.94574612",
4   "user": {
5     "nama": "fariz",
6     "email": "farizdotid@mail.com"
7   }
}
```


Register Gagal (email sudah ada di database)

```
1 {  
2   "error": true,  
3   "error_msg": "User telah ada dengan email farizdotid",  
4 }
```

Keduabelas,

Selesai dan API Login dan Register tersebut sudah bisa digunakan. Agan bisa men-download project tersebut disini :

BACA DULU GAN/SIST!

Untuk meng-support blog ini, silahkan Like terlebih dahulu fanpage Farizdotid untuk membuka URL Download
:) Terimakasih :)

LIKE :)

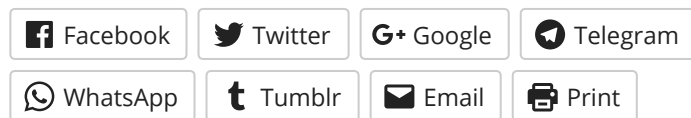
Sekian artikel *API* mengenai **Cara Membuat API Login dan Registrasi Menggunakan PHP**. Semoga berguna dan bermanfaat. Seperti biasa jika ada yang ingin ditanyakan bisa berkomentar dibawah ataupun PM ke salah satu media sosial saya.

Hatur nuhun! 😊

Sumber Referensi :

<http://www.androidhive.info/2012/01/android-login-and-registration-with-php-mysql-and-sqlite/>

SHARE THIS:



Previous Post: [Referensi Library Textview/EditText Android](#)

Next Post: [Tutorial Login dan Register Menggunakan Retrofit2 dan API Server](#)

1 COMMENT

[Tutorial Login dan Register Menggunakan Retrofit2 dan API Server](#)

MAY 14, 2017 | [PERMALINK](#)

LEAVE A REPLY

Your email address will not be published. Required fields are marked *

Comment

Name *

Email *

Website

POST COMMENT

☐ Confirm

you are
NOT a
spammer

☐ Notify me of follow-up comments by email.

☐ Notify me of new posts by email.

[About](#) / [Privacy Policy](#) / [Term of Use](#) / [Sitemap](#) / [Portofolio](#) / [Jasa Konsultasi](#)
[Pemrograman Android](#)

Designed using [Brigsby](#). Powered by [WordPress](#).