



Tutorial CakePHP Dasar Part IV - Searching

Oleh: Cecep Yusuf

Masuk ke Serial Tutorial CakePHP bagian ke IV, kali ini kita akan belajar membuat **searching** dengan CakePHP. Level **Newbie** dan dianjurkan mengikuti tutorial sebelumnya.



Selamat pagi sahabat Jaco, kali ini saya akan melanjutkan tutorial CakePHP untuk bagian ke IV. Kali ini kita akan belajar membuat **searching** dengan **CakePHP**. Mengenai level kita set **Newbie** untuk level tutorial ini ya. Kita akan melanjutkan aplikasi yang sudah kita buat dan akan kita terapkan teknik pencarian untuk aplikasi kita dengan query dasar terlebih dahulu sebelum masuk ke pencarian *expert*.

Adapun teknik pencarian yang akan saya jelaskan adalah menggunakan metode `$_GET`, yaitu metode pencarian dengan membaca variable yang kita inisialisasikan di *address bar browser*. Manfaatnya dibanding dengan menggunakan `$_POST` yaitu **dapat dibookmark** dan meningkatkan SEO dari hasil pencarian itu sendiri, apalagi jika kita tambahkan meta description dan keyword, mantap pokoknya.

Ok langsung aja yuk bagaimana kita cara membuatnya. Kita mulai dengan melanjutkan materi tutorial sebelumnya, yaitu [Tutorial CakePHP Dasar Part III - Sorting, Pagination](#). Atau jika ingin langsung download file sourcena, bisa didownload di sini: <http://download.cheyuz.com/jagocoding.com/tutorial-dasar-cakephp-3-blogku.zip>.

Membuat Search Form

Langkah pertama yaitu kita membuat sebuah form yang berisi textbox untuk query pencarian beserta tombol submit/search. Kita akan membuatnya dengan menggunakan helper bawaan CakePHP, yaitu **FormHelper**. Silahkan buat kode seperti di bawah ini dan simpan di atas table.

```
<?php
echo $this->Form->create('', array('type' => 'get'));
echo $this->Form->input('q', array('label' => 'keyword'));
echo $this->Form->end('Search');
?>
```

Perlu diperhatikan bahwa di dalam function create() pada FormHelper kita mengisi parameter **type** dengan **get**, yang artinya method dari form yang kita submit adalah \$_GET.

Maka hasilnya adalah sebuah textbox berlabelkan "keyword" dan sebuah tombol search. Kamu dapat liat gambar di bawah ini:



Menambahkan Kode pada Controller

Setelah kita membuat form untuk pengisian *search keyword*, selanjutnya kita akan membuat kode untuk menangani apa yang akan terjadi jika kita submit query/keyword yang kita isi tersebut. Silahkan dibuka file **ArtikelController.php**, kemudian tambahkan kode di bawah ini sebelum pagination:

```
...  
  
        $conditions = array();  
  
        if (!empty($this->params['url']['q']))  
        {  
            $conditions['or']['judul like'] = '%' .  
$this->params['url']['q'] . '%';  
            $conditions['or']['konten like'] = '%' .  
$this->params['url']['q'] . '%';  
        }  
  
...
```

Seperti biasa, mari kita jelaskan satu persatu :D

Variable **\$conditions** kita set terlebih dahulu kosong, atau dalam hal ini adalah *array* kosong, karena jika tidak ada query, akan ditampilkan semua data, artinya jika **tidak ada conditions**, maka data akan ditampilkan semua.

Setelah itu program mengecek apakah ada variable **\$_GET** untuk **"q"** atau tidak? Jika **ADA**, maka tambahkan conditions dengan aturan **judul like** query yang diinputkan **ATAU konten like** query yang diinputkan. Pada contoh ini ditambahkan tanda persen (%). Jika kamu sudah pernah bergelut di database MySQL pasti tahu tentang hal ini. Ya, dengan menggunakan persen ini, kita tidak peduli apa karakter yang ada pada sebelum kata maupun setelah kata. Artinya, dengan menginput kata **"oto"**, kita akan mendapatkan kata **"motor"**, **"soto"**, dan sebagainya.

Okay selanjutnya kita ubah kode di bawah ini:

```
$this->paginate = array(  
    'limit' => 3
```

```
);
```

menjadi seperti ini:

```
$this->paginate = array(  
    'limit' => 3,  
    'conditions' => $conditions  
);
```

kita menambahkan key **conditions** dengan value / nilai **\$conditions** yang didapatkan berdasarkan pengecekan keyword di atas. Selanjutnya ayo kita coba dengan klik URL berikut: <http://localhost/blogku/artikel/>.

Dan HORE, kita berhasil membuat searching, mudah bukan?? Sejauh ini aplikasi blog sederhana kita sudah ada fitur mesin pencarinya, ya walaupun masih sederhana hehehe.

Eitt tapi tunggu dulu, tapi *kok* rasanya aneh ya, kok setelah klik tombol cari tulisan keyword yang ada di textbox kok malah hilang ya?



Menambahkan Set Value pada Input Text

Ya, karena kita belum memberikan value pada textbox berdasarkan keyword yang ada di URL. Cara menambahkannya adalah dengan menambahkan kode berikut ini:

```
'value' => (!empty($this->params['url']['q']) ? $this->params['url']['q']  
: '')
```

pada form input, sehingga form input untuk query/keyword menjadi seperti ini:

```
echo $this->Form->input('q', array('label' => 'keyword', 'value' =>  
(!empty($this->params['url']['q']) ? $this->params['url']['q'] : ''));
```

Silahkan dicoba kembali, refresh halaman. Sekarang value akan muncul dengan semestinya :)



"Segitu doank??? Gampang banget ya bikin search pake CakePHP???"

Ya, memang gampang, oleh karena itu framework ini jadi framework favorit saya, dengan keunggulan kecepatan developing yang mantap, CakePHP bisa membuat aplikasi yang rumit dengan mudah, sedikit sekali kode.

OK deh, kayaknya cukup untuk tutorial kali ini, nanti insya Allah akan saya update terus untuk [#serial-tutorial-cakephp](#) ini, pantengin aja terus pokoknya ya. Di Jagocoding.com tutorial akan selalu update ;)

Seperti biasa jika Anda "hoream" membaca dan mengikuti tutorial ini, silahkan download source dengan mengklik tombol di bawah, untuk nanti dilanjutkan pada tutorial berikutnya.

Stand by with me,

Follow saya supaya kamu tetap berlangganan dengan artikel saya.

Tentang Penulis



Cecep Yusuf

Hi, my name is Cecep Yusuf. However, in the virtual world I am more likely to use the name Cheyuz, which is an abbreviation of two words "Cecep" and "Yusuf". I am founder of Jagocoding.com, u can view more of me in [Cheyuz.id](#)