Tutorial phalcon

Apa itu phalcon?

Phalcon merupakan framework PHP paling cepat saat ini. Kecepatan eksekusi programnya mengalahkan beberapa framework terkenal. Phalcon juga sempat menjadi perbincangan menarik di salah satu forum PHP Indonesia. Diawali dari thread salah satu anggota yang membagikan hasil survey Popular PHP Framework oleh SitePoint (http://www.sitepoint.com/best-php-frameworks-2014), selanjutnya banyak member yang berkomentar dan tidak menyangka bahwa nama yang terasa asing ini menempati urutan kedua terpopular setelah [Laravel](http://laravel.com/) dengan perolehan 16.73%.

Project Phalcon sendiri sebenarnya sudah dimulai sejak 2012, kemudian baru meraih stable release  tidak lama ini tepatnya pada 6 June 2014. Framework ini ditulis dalam bahasa C, C++, dan PHP. Phalcon juga menggunakan pola MVC seperti halnya framework popular lainnya seperti Cake, Codeigniter, Yii, Laravel, dll. Catatan penting yang harus kita tahu, ternyata wujud Phalcon adalah PHP C-Extension. Phalcon tidak ditulis dalam plain PHP.

Kamu tidak akan menemukan folder berisi file .php seperti halnya frameworklain. Contoh extension sendiri seperti yang biasa kita gunakan BCMath, Ctype, FTP, MySQL, ODBC, Overload, PCRE, Session dan Curl.  Jika sudah terpasang, kita dapat memanggil fungsi tersebut langsung di source codephp.

Phalcon dibungkus dalam ekstensi C, bertujuan untuk menangani lebih banyak request. Jika ditulis dalam C maka kecepatan eksekusi program akan lebih cepat dan penggunaan resource juga berkurang.

Sebagai framework, Phalcon pun sudah menyediakan berbagai alat perang yang kita butuhkan seperti : ORM, Pagination, Cache, Form Builder, dan Template Engine bernama **Volt**. Kamu bisa lihat lebih lengkap disini <http://docs.phalconphp.com>

Download

Linux/Unix/Mac

#Ubuntu

sudo apt-get install php5-dev php5-mysql gcc

#Fedora

sudo yum install php-devel php-mysqlnd gcc libtool

#RHEL

sudo yum install php-devel php-mysql gcc libtool

#Suse

yast2 -i php5-pear php5-devel php5-mysql gcc

#FreeBSD

pkg\_add -r phalcon

#Debian

wget -O – http://debrepo.frbit.com/frbit.gpg | sudo apt-key add – aptitude update

aptitude install php5-igbinary php5-mongo php5-oauth php5-phalcon php5-runkit php5-stats php5-stomp php5-yaf php5-yaml

windows

<http://phalconphp.com/en/download/windows>

Instalasi

Linux/Unix/Mac

Untuk menggunakan Phalcon di linux,kita hanya tinggal meng-compile dan menginstall ekstensinya dari source codenya.  
Membutuhkan:

– PHP 5.3 development resources / versi yang lebih baru

– GCC compiler

– Git

Tahap selanjutnya,merupakan tahap Compile: yang dilakukan pertama kali adalah membuat extensi :

git clone –depth=1 git://github.com/phalcon/cphalcon.git

cd cphalcon/build

sudo ./install

Tambahkan Extensi ke konfigurasi Php :  
#Suse: Add this line in your php.ini

extension=phalcon.so

#Centos/RedHat/Fedora

Add a file called phalcon.ini in /etc/php.d/ with this content: extension=phalcon.so

#Ubuntu/Debian

Add a file called 30-phalcon.ini in /etc/php.d/ with this content: extension=phalcon.so

Restart WebServer

Windows

Untuk menggunakan Phalcon di windows, kita dapat mendownload DLL library di <http://phalconphp.com/en/download/windows>

Selanjutnya Lakukan Instalasi Phalcon di XAMPP  
Setelah berhasil mendownload Library, Extract library tersebut untuk mendapatkan Phalcon.dll

Selanjutnya, Copy library tersebut ke folder xampp/php/ext/

Kemudian Edit file php.ini , dan tambahkan di akhir :  
extension=php\_phalcon.dll

Restart Apache Web Server dari XAMPP Control panel. Ini akan memproses Configurasi PHP baru

Pada browser ketik http://localhost. Kemudian buka phpinfo() , dan cari informasi tentang Phalcon ,bjika sudah ketemu berarti Phalcon Framework telah berhasil terpasang

Membuat project baru

Setelah berhasil [meng-install Phalcon Framework](http://www.klikpedia.com/2014/08/05/spesifikasi-kebutuhan-dan-instalasi-phalcon/), maka selanjutnya kita sudah dapat memulai membuat sebuah project aplikasi baru. Berikut ini adalah penjelasan langkah-langkah dasar untuk membuat sebuah project aplikasi baru menggunakan Phalcon Framework.

Jika kita telah menggunakan framework PHP lain, kita akan mengharapkan beberapa file framework terkait di suatu tempat dalam struktur folder proyek kita. Dengan Phalcon, semua file ini sudah tersedia dalam memori, sehingga semua yang perlu kita lakukan untuk memulainya adalah membuat struktur folder kosong di suatu tempat di dalam Apache document root (defaultnya /var/www). Cara yang disarankan untuk melakukannya adalah sebagai berikut:

1. **Membuat Struktur Folder**  
   Untuk membuat sebuah project baru dengan Phalcon Framework kita harus membuat foldernya terlebih dahulu. Masuk ke dalam htdocs apache anda, lalu buatlah folder baru dengan struktur sebagai berikut:

project\_name/

app/

controllers/

models/

views/

public/

css/

img/

js/

Jadi apa yang kita miliki di sini adalah folder proyek yang memiliki 2 folder utama: app dan publik. Yang pertama akan menjadi rumah logika aplikasi kita (kebanyakan PHP) sedangkan yang kedua adalah di mana browser kita akan mengarahkan dan diarahkan ke sumber daya dalam folder app di satu sisi, dan memiliki akses ke semua aset frontend, di sisi lain.

1. **Bootstrap File**

File Yang pertama dan paling penting yang kita butuhkan untuk membuat adalah file index.php aplikasi kita akan digunakan untuk bootstrap. Buat file ini di public/folder aplikasi kita:  
nano /var/www/project\_name/public/index.php

Bootstrap file berisi konfigurasi aplikasi dan service dari Phalcon yang kita gunakan. Konfigurasi ini dibuat dalam file bernama index.php dan disimpan dalam direktori application-name/public/. Untuk contoh kali ini konfigurasi yang digunakan cukup sederhana yakni memuat konfigurasi autoloader, dependency injection(DI), komponen view, dan URL. Script pada file index.php adalah sebagai berikut:

***<?php***

***try***

***{***

***//Register an autoloader***

***$loader = new \Phalcon\Loader();***

***$loader->registerDirs(array(***

***'../app/controllers/',***

***'../app/models/'***

***))->register();***

***//Create a DI***

***$di = new \Phalcon\DI\FactoryDefault();***

***//Setup view component***

***$di->set('view', function(){***

***$view = new \Phalcon\Mvc\View();***

***$view->setViewsDir('../app/views');***

***return $view;***

***});***

***//Setup base URI***

***$di->set('url', function(){***

***$url = new \Phalcon\Mvc\Url();***

***$url->setBaseUri('/tesphalcon/');***

***return $url;***

***});***

***//Handle the request***

***$application = new \Phalcon\Mvc\Application($di);***

***echo $application->handle()->getContent();***

***}***

***catch(\Phalcon\Exception $e)***

***{***

***echo "PhalconException: ", $e->getMessage();***

***}***

Penjelasan kode:

***Autoloader***

Terdapat pada baris ke 5-9. Pada bagian ini kita akan mendaftarkan direktori dimana controller dan model akan dibuat. Caranya adalah dengan memanggil classPhalcon\Loader() lalu mendaftarkan direktori tempat controller dan model berada.

**Dependency Injection (DI)**

DI adalah sebuah konsep design pattern dalam OOP. Dengan menggunakan DI kita dapat mendaftarkan komponen-komponen yang terlibat dalam aplikasi. DI berperan dalam pembuatan konfigurasi aplikasi.

Buatlah sebuah variabel yang akan meng-instance pada class Phalcon\DI\FactoryDefault(). Phalcon\DI\FactoryDefault() adalah sebuah varian dari Phalcon\DI. Dengan mengakses class tersebut kita dapat menggunakan service yang disediakan oleh Phalcon.

Untuk komponen-komponen yang disediakan dalam contoh ini adalah view dan URL. Pengaturan masing-masing komponen dengan menggunakan DI adalah sebagai berikut:

a. View

Pengaturan komponen view dilakukan dengan cara mendaftarkan direktori dimana file-file view disimpan. Script untuk pengaturan komponen view terdapat pada baris 15-19.

b. URL

Script untuk pengaturan URL terdapat pada baris 22-26. Pada baris tersebut kita dapat mendaftarkan base URL dari aplikasi. Pada contoh ini saya menggunakan ‘/tesphalcon/’ sebagai base URL sesuai dengan nama aplikasi yang dibuat dan nama folder dimana aplikasi ini tersimpan.

Pemanggilan tiap-tiap class juga dapat dilakukan dengan menggunakan namespace. Sehingga script pada index.php dapat ditulis seperti ini:

<?php

use Phalcon\Loader;

use Phalcon\DI\FactoryDefault;

use Phalcon\Mvc\Url;

use Phalcon\Mvc\View;

use Phalcon\Mvc\Application;

try

{

//Register an autoloader

$loader = new Loader();

$loader->registerDirs(array(

'../app/controllers/',

'../app/models/'

))->register();

//Create a DI

$di = new FactoryDefault();

//Setup view component

$di->set('view', function(){

$view = new View();

$view->setViewsDir('../app/views');

return $view;

});

//Setup base URI

$di->set('url', function(){

$url = new Url();

$url->setBaseUri('/tesphalcon/');

return $url;

});

//Handle the request

$application = new Application($di);

echo $application->handle()->getContent();

}

catch(\Phalcon\Exception $e)

{

echo "PhalconException: ", $e->getMessage();

}

Pause.