

Does anyone write really long, complex PHP apps? [closed]

Asked 16 years, 2 months ago Modified 10 years, 5 months ago

Viewed 3k times  Part of [PHP](#) Collective



2



Closed. This question needs to be more [focused](#). It is not currently accepting answers.



Want to improve this question? Update the question so it focuses on one problem only by [editing this post](#).

Closed 10 years ago.

[Improve this question](#)

It's hard for me to imagine a php script that is more than a few hundred lines of code. It seems that, for a non-persistent environment, web-based scripting is usually done in small chunks and used for the purpose of delivering a portion of a website to the end user. I'd like to know if people are developing any type of large, or persistent, or complex apps with php, and what is it exactly you are working on. I've only done small projects for small websites, so I don't know what can be accomplished on a larger scale. It would also be nice to know what libraries you are using, and what other

technologies you are integrating with. Please enlighten me so I can start to dream of bigger things!

PHP

php

Share

Improve this question

Follow

edited Jun 27, 2014 at 5:18



tshepang

12.4k ● 25 ● 95 ● 139

asked Oct 6, 2008 at 2:31



Ixion

745 ● 1 ● 7 ● 8

10 Answers

Sorted by:

Highest score (default)



12



I would look at some of the well-known open source web apps that use PHP to get a good sense of what can be accomplished, and how PHP is used in each of them. The advantage is that since they are all open source, you can actually look at the PHP code to see how various functionality was implemented.



Some good examples to look at include:

- [WordPress](#)
- [TextPattern](#)
- [MediaWiki](#)
- [PhpBB](#) *questionable code quality

- [SugarCRM](#)
- [Joomla](#)
- [Drupal](#)

Also look at some of the popular frameworks to see what kind of functionality they offer (this should give you a good sense of what types of things PHP is most often used for):

- [Zend Framework](#)
- [CakePHP](#)
- [CodeIgniter](#)
- [Symfony](#)

Share Improve this answer

edited Oct 6, 2008 at 6:52

Follow

answered Oct 6, 2008 at 2:45



Wilco

33.3k ● 49 ● 132 ● 161

1 Ugh, PHP-BB should not be on that list. Never before have I seen so many security vulnerabilities packed into such few lines of incomprehensible spaghetti code. – [John Millikin](#) Oct 6, 2008 at 3:06

ROFL, that is SOOOO true, I have an app with 3000+ and I've had everything you can imagine thrown at it with no issues yet. – [UnkwnTech](#) Oct 6, 2008 at 3:36

2 Maybe move PHPBB to "Good to look at so you know what NOT to do" – [UnkwnTech](#) Oct 6, 2008 at 3:36

Haha, good to know - I'll add a note to that one. – [Wilco](#) Oct 6, 2008 at 6:51

1 I'll also say Wordpress' code isn't too crash hot. I'd say better to look at the Zend Framework. – [alex](#) Aug 24, 2009 at 5:32



2



For my day-job we run everything in PHP - our front-end website, our backend for agents and employees, inventory, server control interfaces, etc. These are everything from spiffy new AJAX-enabled Zend Framework apps to legacy code that we haven't ported yet. On top of that we use things like [Mantis](#) (bug tracking built in PHP), [Mediawiki](#), and [phpMyAdmin](#).

The only thing that isn't PHP are vendor apps because vendors love Java. The one ASP.NET application we have was actually abandoned by the vendor during the project (not really a knock against ASP.NET, that app was just the perfect definition of a runaway project and would have failed no matter what language it was written in).

With mature frameworks like [Zend Framework](#), [CodeIgnitor](#), and [CakePHP](#) creating just about anything in PHP is possible.

Share Improve this answer

answered Oct 6, 2008 at 4:02

Follow



[dragonmantank](#)

15.5k ● 22 ● 85 ● 92



2



The biggest problem developing large scale programs is definitely keeping them maintainable in the long-term.

Initially, a program starts out all full of ideal methods and ideas, but keeping the integrity intact, especially, over time fails, in my opinion, more often than not.

In addition, scope creep is your enemy. You HAVE to reign that in ASAP.

As far as large scale programs go the company I work for has a few internal programs constantly under development. One example is our proprietary website engine. It's a very large code-base that includes a dozen modules (user management, survey system, blogs, user galleries, etc) that allows us to build our clients sites rapidly.

We also develop our own internal project management program for managing our clients work.

You should definitely be thinking in terms of scale in the long term. In almost every project I've worked on there's a permission/group element for users involved. You might want to start thinking about the possibilities and issues involved in that and work up to more complicated functionality.

Share Improve this answer

answered Oct 6, 2008 at 4:10

Follow



[Eric Lamb](#)

1,434 ● 5 ● 17 ● 29



1

[MediaWiki](#) is one of the largest public PHP apps, and it's got very nice code. . I know some larger ones, but they're utterly awful and you'd learn nothing by reading them.



Share Improve this answer

answered Oct 6, 2008 at 2:34

Follow



[John Millikin](#)

201k ● 41 ● 215 ● 227



The MediaWiki code base is ok until you decide to extend it. I've been doing NLP research with it so i guess my view point is skewed by the complexity of my problem. – [gradbot](#) Oct 8, 2008 at 15:40



1

There are **lots** of complex OpenSource php applications. For example, the Drupal CMS, which can be considered a platform in its own right for developing other web sites.



You can browse through the source code online:

<http://cvs.drupal.org/viewvc.py/drupal/drupal/>



Share Improve this answer

answered Oct 6, 2008 at 2:37

Follow



[Rob Walker](#)

47.4k ● 15 ● 100 ● 137

I think magento is much better in terms of Awesome code and power, big project ! – [Pratik Joshi](#) May 19, 2015 at 14:01



+1 for Wilco

1

I have a software I use for some of my clients, it's a CMS, Blog, eCommerce beast, the code base is HUGE, but everything cooperates with each other nicely.

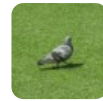


Share Improve this answer

answered Oct 6, 2008 at 2:48



Follow



UnkwnTech

90.7k ● 65 ● 191 ● 234



1

My company works on educational software. We've recently started doing web-based content delivery, including video and audio, with the backend written entirely in PHP using MySQL. We have two primary apps, one which lives on our servers and one which is delivered to the customer. One clocks in at ~42,000 lines of code (using a physical line count) and one at ~68,000 lines.



We use [PEAR](#) extensively and a recently started project is using the [Zend Framework](#).

Share Improve this answer

answered Oct 6, 2008 at 4:33

Follow



Michael Johnson

2,307 ● 16 ● 21

How much painful it was to convert from OOP PHP to Zend ?
– [Pratik Joshi](#) May 19, 2015 at 13:58



We use PHP at our company. (We do online language learning: <http://www.livemocha.com>. You should go take a

1

look at the site. Yes, it's sort of a shameless plug, but it's also topical. :-))



I can't give you a precise number of users, but we put out a press release a while back celebrating hitting the 3 million mark. That's a pretty large scale as web apps go.

We build on the CakePHP framework, which is based on an MVC architecture... at least in theory. In practice, they auto-generate certain methods for the models which tend to have the result of pushing some pieces of model code (caching, deciding which DB to use) into the controllers. They also have a few localization issues in 1.2 that make me think this part of the framework hasn't really reached maturity yet. That said, I find CakePHP pretty comfortable to work with overall, and you should at least take a look at it if you're considering implementing a large-scale web app in PHP. It has some excellent documentation available as well (google for "CakePHP bakery").

Share Improve this answer

answered Dec 24, 2009 at 6:23

Follow




Arkaaito

7,387 ● 3 ● 41 ● 57

Built with says it was made in WurdPress .

builtwith.com/livemocha.com – Pratik Joshi May 19, 2015 at 13:53

That's entirely possible - I know they went through a major rebuild (lasting multiple years) about 3 years after I posted this message. They've also since been bought by Rosetta Stone, so maybe RS pulled down the site altogether and

replaced it with an ad page. (I don't know - I no longer work there.) – [Arkaaito](#) May 21, 2015 at 22:26 



0



Get CodeIgniter and rebuild Amazon or Ebay. If you can dream it you can build it in PHP but you might not be able to maintain it because it is so easy to create bad code that works. PHP.net is your friend. Whatever framework you use make sure you read the User Guide and let it guide you.



Share Improve this answer

answered Oct 8, 2008 at 5:57

Follow



[Clutch](#)

7,582 ● 11 ● 47 ● 57



0



I can't believe nobody has mentioned the MVC pattern yet. IMO, it's one of the best things you can use to help you maintain large codebases.



Share Improve this answer

answered Aug 24, 2009 at 5:34

Follow



[alex](#)

490k ● 204 ● 889 ● 990