best language / framework for a web CRUD app with roles on Linux

Asked 15 years, 11 months ago Modified 11 years ago Viewed 4k times

▲

0

▼

I have a Linux web server and I'd like to make some database tables (currently in Access) available on the web for CRUD. There needs to role-based security. What's the quickest path to develop this?

Also, which database would be best? I already have mySQL running on that box if it makes any difference.

linux crud rad security-roles

Share

Improve this question

Follow

edited Apr 8, 2009 at 20:47

George Stocker
57.9k ● 29 ● 180 ● 238

asked Jan 15, 2009 at 3:11

JoelFan
38.6k ● 38 ● 141 ● 213

4 Answers

Sorted by: Highest score (default)

▲

I agree with Chuck, the question shouldn't really be about the language, but about the framework you choose.

1

✓

I did something similar to you a while back, and ended up using Ruby on Rails, and the activescaffold plugin (http://www.activescaffold.com/) to provide a pretty front end. The actual code I ended up writing was extremely minimal. There are other plugins for Rails which provide role based security too (which I didn't bother with, I just had "you're either logged in and have write access, or you're not logged in and you don't") and which also mean you don't have to write much stuff yourself.

So put me in the camp for Rails come the religious war ;)

Edit: MySQL is a perfect database to use, so you don't have to worry there.

Share Improve this answer Follow

edited Jan 15, 2009 at 14:14

answered Jan 15, 2009 at 7:29

GodEater

3,707 ● 2 ● 29 ● 30

1

This will turn into a religious war between the Ruby on Rails camp and the Python camp, with a good smattering of the PHP and Perl. You should evaluate the langauges yourself and decide what is best for you. There are, of course, other choices, however listing those would just elicit mroe religious battles. Although, I would say all of those I listed would be reasonable choices. You can usually create a good design in spite of any shortcommings your chosen language may have.

Share Improve this answer Follow

edited Jan 15, 2009 at 3:25

answered Jan 15, 2009 at 3:20

BobbyShaftoe

28.5k ● 7 ● 54 ● 74

1

If you want a solid, clean, stable CRUD web app that can be maintained and expanded for years to come, stick with the standards: PHP, Perl, JavaScript, CSS, and HTML. Learn those roots languages well. Take the time to do it right and focus on good coding habits like clarity, consistency, and organization. Practice good reuse of code, good naming, good commenting, and good database design. Test, document, and refactor. Take pride in the craftsmanship of your CRUD app. Learn it inside and out. Set the stage so you can later maintain and expand it. Your goal is to build something that will work well, last a long time, and make a great return on the business investment. Someone once said that it takes 10 years to become a good coder.

As for frameworks, plugins, and external libraries, that's wonderful icing to put on your cake. But never confuse the icing with the cake. If you want to learn to code, take the time to learn it right. If you're not comfortable coding a simple CRUD app, you'll be even less comfortable trying to navigate the framework-generated code. Coding is a wonderful gig. But never mistake the sizzle for the steak.

Share Improve this answer Follow

answered Dec 6, 2013 at 1:59

John Doe
973 ● 9 ● 10

▲

0

▼

I don't think language is the question you should be asking. There's no language particularly well-suited to CRUD Web apps. There are many *frameworks* designed for that sort of thing, though, in many different languages, and those are probably what you'll want to look at.

I think Rails is the best in general, and that's what I use for most projects. It's very well-suited to CRUD apps (to the point where it allows you to create a simple one without writing any code at all). But if there really were an undisputed "best" choice, you'd probably already know about it. Instead, some apps are made with Rails, some with Django, some with Cake, and so on and so forth.

Share Improve this answer Follow

answered Jan 15, 2009 at 3:48

Chuck
237k ● 30 ● 304 ● 392