

LARAVEL

STEP 1 – INSTALLATION

It was very easy to install (using composer) if you have the right PHP version. For me, I had to reinstall XAMPP, but this has nothing to do with Laravel.

STEP 2 – INTRO

The intro was great, it made it look easy – and it is. Also the entire design of the documentation website is something that was never seen when it comes to frameworks

STEP 3 – FIRST LINES

It's easy to make the first steps, mainly because of how easy routing works. However, it can result to following a pattern further from MVC, which is not good.

STEP 4 – DB

One of the greatest features that I know of: Migrations. Elegant and useful, easy to use! Speaking of db methods and data manipulation, everything is straight forward and intuitive. No problems whatsoever.

STEP 5 – GETTING DEEPER

Laravel does get complicated (as in advanced) if it has too, but in the same elegant manner. Validation is really nicely done, the Auth Class is superb and Blade is a really a great way to template. The templating system overall is very useful and I can see a lot of people ditching HTML tags in favor of Blade syntax.

PHALCON

STEP 1 – INSTALLATION

...

STEP 2 – SOME MORE HOURS OF INSTALLATION

It's frustrating because the documentation is so cold and support so little, that it makes you feel like things are going to break – and they do. Finally, I got everything working with quite some effort, especially for installing the Phalcon Dev Tools

STEP 2.1 – INTRO

LEARN . BY . SUFFERING.

...

SUFFER SOME MORE

...

The Phalcon dev tools (<https://github.com/phalcon/phalcon-devtools>) are awesome. Although nobody wants to really put constraints on the file/folder structure, by using the dev tools and executing 'create-project' a very nice structure is created. Dev tools also create models from database tables (really, really nice) and can create controllers and views. (has lots more features)

STEP 3 – FIRST LINES

Trying to figure out a good way to do things is impossible, documentation is poor and the community is busy coding, because they are not writing on forums. The only way to learn is to dig deep into the available examples (there are a few) and then suffer by trying to apply them in your app. They won't work and there will be 10 ways of doing the same thing, but at the end of the day you get considerably wiser.

STEP 4 – DB

Almost finished the application without knowing about '**useDynamicUpdate(true)**' that can be used on the init method of a model and allows fields of a row to be updated, rather than the entire row (are you fucking serious, this is not mentioned anywhere in the docs?! Almost threw my computer out of the window because of this)

STEP 5 GETTING DEEPER

..and it gets deep. So deep you can easily get lost and be on your own. A lot of things to explore: from ACL to Volt – everything has a lot of functionality which you have to.. guess. Some things I don't like in the end: Volt syntax is not intuitive (no closing tags), routing is not that good (or maybe I didn't even do it right, I will never know), validation on the Model (as they suggest a few times) is not a good idea either.

CONCLUSION

Phalcon is fast and permissive, while Laravel is intuitive, more precise and (very important) fun.

Blade wins by far the battle with Volt.

Overall Laravel makes it easier to create applications and it's only logical that it comes with a great cost: speed.

Therefore, for really resource intensive apps, Phalcon is still a good option and with time the community will grow.