Report - Assignment III

Robin Bekkevold - 110065

Emma Johansen Nylund - 130712

Ferdinand Grüner - 150096

Siri Meen - 130703

Maksymilian Sadowski - 130705

Setup

Requirements:

- Composer
- PHP 5.5.15

Setting up the database:

- Creating a local database.
- Updating the database information in the .env file in the root directory.
- In the terminal navigate to the projects folder.
- Run:
    - composer dump-autoload
    - composer clear-cache
    - php artisan migrate
    - php artisan db:seed

And you should be good to go. Now simply go to your browser and enter the directory + /public and the page is displayed. In production this is the folder the url would point to.

Collaboration/Tools

To keep track of the task which had to be done, and the tasks someone was currently working on, we used a todo-list realized with Google Docs at first. But after a while it was kind of hard to keep track of the tasks everyone had to do. So we switched to a web application called Trello (trello.com).

For version control of the code we used Git and SourceTree to make it easier to use for us on our computers.

On top of all those virtual tools we met at least once a week to discuss about our current progress, the things that have to be done (updating trello) and current problems. All in all our organisation can be defined as "SCRUM-ish", with a leader and one week long sprints.

Architecture

Backend:

To realize our backend we chose the PHP framework Laravel 5. The reason for that was quite simple, after comparing it to other frameworks like Symfony or CakePHP it seemed to suit our requirements better. Especially the taskline tools like artisan came in handy when setting up the database. For the architecture we used MVC.

4.2 Frontend:

We used jQuery because we all had experience with that and it was easier to use with Ajax which we have used to provide a better user experience, through lazy loading posts without having to reload the entire page. As a templating engine we used the in laravel already included blade. Since we think the only thing changing about a post is its text, the user only has the possibility to change that afterwards. If the hashtag wants to be changed it is considered as a new event and therefore a new post.

Design

For the design of the website we made sketches early in the process, and had them in mind while working out the rest of the project. We decided to use Bootstrap since we have worked a lot with it in earlier projects. Bootstrap's grid system is neat and simple to work with and is fast to implement. The group agreed on having a simple flat design, so that the user can focus on the content more than being overwhelmed by too much eye candy.

As a start page we decided on a page, showing a word cloud with the most recent hashtags, so the user gets some inspiration what to look for and further a search field to search for stories and posts.

All the posts are always shown in chronological order, to make it easy for the user to follow the event in an effective way.

The userpage shows the posts and stories of a user. A user has the possibility to use another users posts to make a story out of them as long as they're under the same hashtag. To do so the user has to simply select the "+Add to story" button shown underneath a post and then either select an existing story or create a new one.

Attachments

https://trello.com/b/fuJhsf0f/mumbler
https://github.com/gruenerf/assignment3