

A new comments system for my static Jekyll site

15 September 2019

I love WordPress, but I strongly believe there's a right tool for every job. This site is not a WordPress site. It's built on Jekyll and hosted on Github Pages for free. I write my posts using simple Markdown and it's just fast and convenient. However, what I always found annoying is the lack of a comment system. Until now I was using disqus for comments on this site, but I never really liked it. It was a last-resort solution. One I hated because Disqus loads a lot of assets and a lot of trackers.

Those who know me personally, know that I am concerned with online privacy and web-sustainability. I don't even have analytics on my sites simply because I consider the policies of tech giants like Google abhorable, and the fact that almost all sites load all these extra JS assets is insanely constly for the environment.

Disqus in particular was loading assets and trackers from Google, Facebook, Twitter and its own things. So I could no longer stand the thought of using

Disqus, a custom solution had to be implemented. Today I finally managed to do that and implement my new comments system.

Since this site is hosted on github pages there is already a repository. And since most of what I write is tech-related, it's safe to assume that people reading these posts have an account on GitHub. So leveraging the **GitHub Issues API**, whenever I want to enable comments on a post I just create a new issue on the repository, and then in the headers of my post just add this:

```
comments id: 5
```

Of course `5` in the above example is just the ID of the issue in the repository and each tickets has its own.

When someone wants to add a comment on a post, they click the link and they get taken to the corresponding issue on the repository where they can leave their comment. As soon as a comment is posted it appears on the page.

The Code

I created a new `comments.html` file in my `_includes` folder, and wrote this in there:

```
<script>
const GH_API_URL = 'https://api.github.com/repos/aristath/aristath.github.com/';

let request = new XMLHttpRequest();
request.open( 'GET', GH_API_URL, true );
request.onload = function() {
    if ( this.status >= 200 && this.status < 400 ) {
        let response = JSON.parse( this.response );

        for ( var i = 0; i < response.length; i++ ) {
            document.getElementById( 'gh-comments-list' ).appendCh
        }
    }

    if ( 0 === response.length ) {
        document.getElementById( 'no-comments-found' ).style.d
    }
    } else {
        console.error( this );
    }
}
```

```

};
|
function createCommentEl( response ) {
    let user = document.createElement( 'a' );
    user.setAttribute( 'href', response.user.url.replace( 'api.github.com/'
    user.classList.add( 'user' );
    |
    let userAvatar = document.createElement( 'img' );
    userAvatar.classList.add( 'avatar' );
    userAvatar.setAttribute( 'src', response.user.avatar_url );
    |
    user.appendChild( userAvatar );
    |
    let commentLink = document.createElement( 'a' );
    commentLink.setAttribute( 'href', response.html_url );
    commentLink.classList.add( 'comment-url' );
    commentLink.innerHTML = '#' + response.id + ' - ' + response.created_at;
    |
    let commentContents = document.createElement( 'div' );
    commentContents.classList.add( 'comment-content' );
    commentContents.innerHTML = response.body;
    |
    let comment = document.createElement( 'li' );
    comment.setAttribute( 'data-created', response.created_at );
    comment.setAttribute( 'data-author-avatar', response.user.avatar_url );
    comment.setAttribute( 'data-user-url', response.user.url );
    |
    comment.appendChild( user );
    comment.appendChild( commentContents );
    comment.appendChild( commentLink );
    |
    return comment;
}
request.send();
</script>
|
<hr>
|
<div class="github-comments">
    <h2>Comments</h2>
    <ul id="gh-comments-list"></ul>
    <p id="no-comments-found">No comments found for this article. Comments
    <p id="leave-a-comment">Join the discussion for this article on <a href
</div>

```

Depending on the number of requests you have on your site you may want to add these at the end of the URL in the `GH_API_URL` constant in the above script:

```
?client_id=MY_CLIENT_ID&client_secret=MY_CLIENT_SECRET
```

To get the client-ID and client-secret you'll have to create a new application from your **GitHub Profile > Developer Settings > OAuth Apps**. Don't worry, it only requires an application name and a URL. You can use the same URL for the callback, so it only takes 10-20 seconds to create your app and get the keys.

You can see how that file is then loaded on the template for posts **here**.

The logic is simple: We make a request to the GitHub API, we get the comments from the issue and render them. The code may seem a bit long but that's only because I'm using Vanilla JS instead of libraries like jQuery, so elements have to be created individually and then added to the DOM. The above code will get the user avatar, create a link to the user's profile, add the comment, add a link to the comment, the date and everything a comment usually has.

It's fast, there's no privacy concerns, no cookies, no nothing. And as an additional bonus you can just style it the way you want it.

I hope that helps someone out there, it took me quite a while to figure this one out.

Comments



This is a demo comment posted on the issue on GitHub.

#531526145 - 2019-09-15T01:18:59Z



Thanks

#531616593 - 2019-09-16T01:10:08Z



Ohh..! Thanks...

#534866123 - 2019-09-25T05:57:15Z

Testing your comment section. Nicely done!



#553926553 - 2019-11-14T15:02:59Z



Thank you for this!

#562585412 - 2019-12-06T14:09:43Z



Very cool!

#563466541 - 2019-12-09T22:19:39Z



Looks simple and elegant

#563899915 - 2019-12-10T07:17:02Z



Very cool!

Could I reply?

#566757849 - 2019-12-17T21:31:11Z



Hmmm it appears I'm not formatting the comments on my site... Should be relatively easy to fix though, I'll investigate ways to do that soon.

#566765477 - 2019-12-17T21:52:39Z



testing...

#568282314 - 2019-12-22T17:02:45Z



Thanks for posting this. After coming across various similar solutions which all failed, this actually worked for my blog (with some minor modifications)!

#585813825 - 2020-02-13T15:25:55Z



testing...

#586666939 - 2020-02-16T04:13:54Z



Thanks for posting this. After coming across various similar solutions which all failed, this actually worked for my blog (with some minor modifications)!

I'd love to hear about the modifications!

#586725052 - 2020-02-16T16:22:48Z



Minor update: **the latest commit** on this repo adds markdown parsing for comments. :)

Update: Switched to **showdown.js** in this **updated commit**.

#586726502 - 2020-02-16T16:38:28Z



@aristath thanks for the update, I had noticed Markdown wasn't rendering, but with these changes it now works perfectly!

#586731388 - 2020-02-16T17:28:11Z



@jackkeller only some really minor things, e.g. in the blog post the issue IDs are hardcoded, so I changed that into liquid variables. But looking at @aristath's code I now see he also use liquid variables, so that's not really a modification after all. Anyway, if you want to have a look my site is **here**, and **this is the source repo**.

#586731447 - 2020-02-16T17:28:43Z

Join the discussion for this article on **this ticket**. Comments appear on this page instantly.

© 2020 Ari Stathopoulos