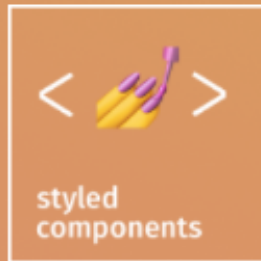


# Contribution to an Open Source project

*Styled components*



**Visual primitives for the component age.**

Use the best bits of ES6 and CSS to style your apps without stress 🙌

Nils Goksör

Fall 2017

# Introduction

This report is done in the course “Lingi2401 - Open Source strategy for software development”, where the student got to choose an open source software of their liking and then contribute to it somehow.

In this report I describe my contribution to the open source software “Styled Components”. In the first part of the report, it’s explained what the software is and how they work within the community. In the second part, I describe my contribution to the project in the form of a journal.

## About Styled Components

### What is “Styled Components”?

To understand the purpose of Styled Components, one must first be familiar with React.

React is used to create UI for web applications and what make’s it different from other alternatives is the usages of components, which are like HTML elements but with customised functionality. The advantage with React is that it makes reading and creating web applications faster and more simple.

There are some best practises that one should follow when using React. One is to only use a class for one purpose. This is good because if you have an error with a button it’s most likely something in the button-class. Another practise is to use components for styling, because a component is often used multiple times in a program so it will be easier to change things and look for errors if all design of all buttons are gathered at the same place.

This is where Styled components becomes important! It’s used together with React, and it basically removes classes because according to the first practise we don’t need them anyway.

Style components enables the user to “style without stress”. Instead of mapping components to styles the user creates the component while defining the style. *Figure 1* shows an example, and what happens is that `styled.h1`` calls a function and returns a React component that renders a `<h1>` with the styling stated (``` states that a function is called).

```
// Create a Title component that'll render an <h1>
tag with some styles
const Title = styled.h1`
  font-size: 1.5em;
  text-align: center;
  color: palevioletred;
`;

// Create a Wrapper component that'll render a
<section> tag with some styles
const Wrapper = styled.section`
  padding: 4em;
  background: papayawhip;
`;

// Use Title and Wrapper like any other React
component - except they're styled!
```

*Figure 1: the basics of styled components*

## Vision and challenges for the near future

Styled Components is quite new still but it's been very successful. It's being used by people at some big companies like Reddit and with over half a million downloads a month, the user base is growing rapidly.

I think the vision for the near future is to continue growing and making the software more known! With version 3 coming out soon the greatest challenge is to make it as successful as previous versions have been and to make people talk about it and thus increasing the awareness for the software...

## Main contributors

If you take a look at the github of Styled Components you'll see that there are quite a few contributors to the project. However, there are three contributors that are contributing more than the rest. These are the main contributors Glen Maddern ([geelen](#)), Max Stoiber ([mxstbr](#)) and Phil Plückthun ([philpl](#)). They've all made over a hundred commits each the past month and a half. They are also listed as the creators of Styled Components on the website.

They've done far from everything. During the same period of time, over 100 different contributors have made commits to the project. In the README many contributors are acknowledged for their help with ideas, code or inspiration.

## The bug tracker

The community is doing a great work keeping the issues down to a low number in the bug tracker. There's currently 85 active issues and over 700 hundred solved. Almost every issue has been assigned one or more labels to make it easy to keep track of what's going on and what needs to be done. Additionally, most issues have been commented by one of the main contributor, and it's common for them to discuss the issue with the one reporting it and give their vision on the issue.

## Communication channels and decision making

As mentioned above, there's a lot of discussions going on in the bug tracker and the main contributors are happy to share their view on things. However, for more general discussion, Styled Components use "Spectrum" to chat. Here, people can create topics which are discussed by the community.

I was curious to how decisions are made within the project so I decided to ask in the spectrum-chat, and I learned the following. It's the three main contributors who makes all decisions but they are happy to listen to suggestions from the rest of the community, and if there is something they decide they can't or won't to, they try to explain why.

It rarely happens that there are any disagreements between them because most decisions are solved purely following the core ideas. Only once have they not managed to find a solution by discussion, so an explicit vote was held between them.

# Contribution journal

## Finding a project

I began by trying to find out if I was using any open source project daily. It would have been fun to contribute to something I'm already using and care about. That way it would have been easier to find a good way to contribute but unfortunately, it turned out I was not using many open source applications.

Instead, I started browsing various websites with suggestions about good open source projects to contribute to for beginners, hoping I would find something that would interest me.

I've always enjoyed developing web applications and working with styling, and I've been wanting to learn React for quite some time. When I came across the open source project "styled-components" it got my attention! After having a quick look at their github I figured it would be a good place to make my first contribution. The community seemed very friendly, and it's not too big nor too small.



**Nils Goksör**  
@nilsgoksor  
36 rep



### First contributor looking for advise & question about decision making

October 21, 2017 · 3:16pm

Hey guys,

I'm a computer science and engineering student and I've recently began learning React (and of course styled-components) on my free time.

I've been wanting to contribute to an open source project for a long time and this seem like a great community to start! I'm currently looking for a way to contribute, I hope to find some issue on github I can dig my teeth into. If you have any recommendations for a first-timer, let me know!

Lastly, I'm curious about challenges that styled-components are facing and who's got the final say on decisions that has to be made. Is it just done between the three creators or is the community involved? What if they disagree?

Cheers guys!

*Figure 2: Introducing myself to the community*

## Getting familiar with the project

After choosing to contribute to Styled Components, I wanted to get acquainted to the community. What's currently happening with the project? What's the vision? Who are the main contributors? Which communication channels are used? Is the bug tracker well managed?

Many of these questions were answered by browsing the website and github. After learning about the spectrum chat which is used within the project, I decided to introduce myself (*figure 2*).

## Finding a way to contribute

Finding a way to contribute wasn't as easy as one might think. Since I never used the software before I didn't really know exactly how it works and what's missing. I figured my best shot would be to find a bug reported at the github for either the project or the website of the project. I wanted to do something challenging, but still in a field I'm somewhat familiar with.

I struggled finding something I was really happy with. Meanwhile, I started writing about what I had learned so far in the report.

## My first contribution


After browsing the bug tracker I discovered something I could dig my teeth into! After commenting on the post one of the main contributors assigned it to me. Time to get to work! (*figure 3*)


## Understanding what to do

The issue is perhaps not the most challenging one programming wise but there was still a lot of work to be done. As mentioned earlier in the report, I've no experience in React nor Styled Components, so it took some time to realise exactly what it is that needed to be done.

I followed some tutorials and did some basic exercises to get used to React and Styled components. I also experimented with the different classes of a DOM node which helped me a lot when I had to think about how to answer the FAQ.

# Add "Why do my DOM nodes have two classes?" to the FAQ #147


 Open mxstbr opened this issue 3 hours ago · 2 comments




mxstbr commented 3 hours ago

Owner + 🗨️

This is copied from a recent chat log, we should clean this up and add it to the FAQ:  
  
The first class ( `sc-bfhe32` ) is static per-component, meaning every instance of a `<Button />` will have that class. The second is per-dynamic-stylr-fragmet, so all `<Button primary />` will have the same one, all `<Button secodary />` will have the same one etc.  
Note that the first one doesn't actually have any styles attached to it because it's only used for when you interpolate the component ( ``${Button}`` ) and so that in the DevTools you can quickly tweak all `Button` s no matter if they're primary, secondary or whatever else




mxstbr added `enhancement` `good first issue` `help wanted` labels 3 hours ago




nilsgoksor commented 3 hours ago

+ 🗨️ ✎️ ✕️

I could give this a go as my first contribution! Just to clarify, I'm supposed to make a new question in the FAQ explaining this in a good way and using an example?

 1



mxstbr commented 2 hours ago

Owner + 🗨️

Yes, that's exactly it! Looking forward to your PR 😊

Figure 3: A post in the bug tracker that later became assigned to me

## My first pull request

When I knew how I wanted to answer the FAQ-question I located the files I needed to change. It took some time to get acquainted to all classes and the structure of the program, probably because it was the first time I had to jump straight into such a big program.

When I felt familiar enough with the structure I made changes to appropriate files. After making sure there were no errors and the website was running correctly I was ready to push my changes from my branch.

I'm not very experienced with github and I struggled a bit before I finally managed to do the PR correctly.

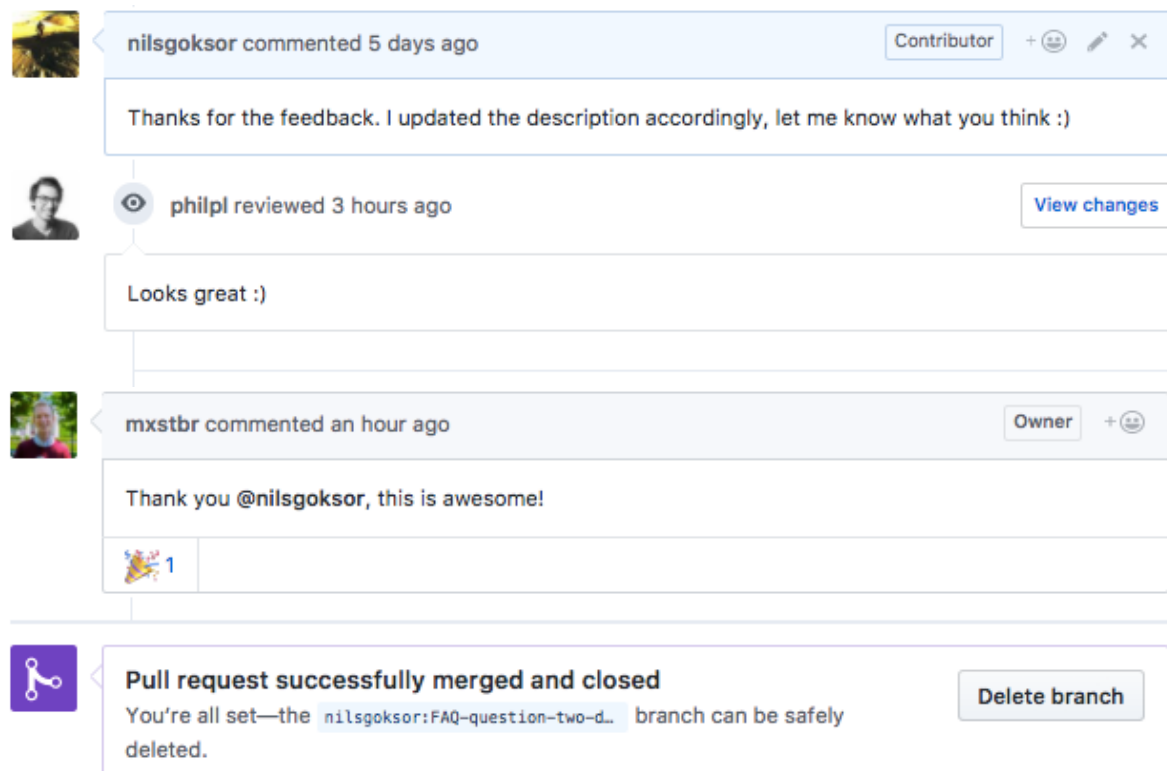
## Feedback

After making the PR I received some feedback. It was mostly minor suggestions like changing the title but one suggestion was quite big, it was suggested that I should not use a live example (which is frequently used on the website) since it doesn't really add much explanation to this question. Making an concrete example with a code snippet was enough for this matter.

While I understand and agree with then motivation given it was still a bit frustrating, since this was a feature I had spent quite some time on. For the next time, I'll make sure I understood exactly what's asked for before I do unnecessary work.

## Pull request two

After working to apply the requested changes for a few days I was ready to submit my 2nd PR, and the next day I received the good news, the PR was accepted! It felt great to have made a contribution to the project and to see the updated website with code that was actually written by me...!



*Figure 4: The second pull request was successfully merged*



## What I've learned

The most important thing I've learned is without a doubt to not be afraid to contribute to a project in the future! To make a contribution doesn't have to be difficult at all, with a friendly community there to help and watch over you.

I learned a lot more, for instance how to get acquainted to the code of big projects and find the right lines to modify, and to really make sure I understand what's asked of me before I get to work.

I'll for sure continue contributing to this and other open source projects in the future, it might even become my new years resolution...