angular / **components** Public

Code **Issues** 1.7k Pull requests 163 Actions Projects 8 Wiki Security Insigh

New issue

Jump to bottom

docs: clarify support for ViewEncapsulation.ShadowDom #13837

Open **Splaktar** opened this issue on 27 Oct 2018 · 8 comments

Assignees

Labels area: many docs elements help wanted P2

Splaktar commented on 27 Oct 2018

Proposal

What is the expected behavior?

If `ViewEncapsulation.ShadowDom` is not supported, this should be called out in the docs to save developers time and frustration.

What is the current behavior?

`ViewEncapsulation.ShadowDom` is not supported and is not documented.

What are the steps to reproduce?

1. Open https://stackblitz.com/edit/angular-i5jw3k?file=app/table-selection-example.ts
2. Notice how the checkboxes aren't themed
3. Open https://stackblitz.com/edit/angular-sbrzfb?file=app/input-overview-example.ts
4. Notice how the inputs are not themed or styled properly

What is the use-case or motivation for changing an existing behavior?

To save developers time and frustration.

Which versions of Angular, Material, OS, TypeScript, browsers are affected?

Angular: 7.0.0
Angular Material: 7.0.2
TypeScript: ~3.1.1
Browsers: Tested on Chrome, Firefox, and Safari latest stable versions

Is there anything else we should know?

Nope

👍 23

Splaktar added the **docs** label on 27 Oct 2018

Splaktar added the **P3** label on 15 Nov 2018

zlepper commented on 13 Jan 2019

Is there any plans to support ShadowDom at some point?

Splaktar mentioned this issue on 4 Apr 2019

docs: clarify that ViewEncapsulation.Native is not supported #3386

Closed

tmburnell mentioned this issue on 2 Jun 2019

Angular Elements with outlined input label #16168

Closed

josephperrott added **P2** and removed **P3** labels on 5 Jun 2019

josephperrott assigned **jelbourn** on 5 Jun 2019

jelbourn commented on 5 Jun 2019

We have to define that "support" means, as well. We'll probably never change the components to *use* Shadow DOM, but when we switch to CSS variables the theming system won't prevent other people from using it.

👍 1

xerofun commented on 21 Jun 2019

@jelbourn where is this on the road map (days, weeks, months)? We're starting to plow headlong into a webcomponent architecture but are heavily invested (and wish to remain so) with our material components.

Are you aware of a way to work around this. Since we're just starting off with componentizing some of our resources, it wouldn't be much effort to make the components happy with angular.

jelbourn commented on 21 Jun 2019

Switching the theming system to use CSS variables is probably going to be early 2020, but for other things (like overlays) we don't have concrete plans yet.

Splaktar added the **help wanted** label on 1 Aug 2019

Splaktar commented on 1 Aug 2019 • edited ▾

Note that when working with Angular Elements, `ViewEncapsulation.None` and using Sass (SCSS) to make sure that your styles are scoped to components is recommended at this time.

Splaktar added the **elements** label on 2 Aug 2019

Splaktar mentioned this issue on 2 Aug 2019

docs(Getting Started): Updated material Getting Started guide to use install schematics and simplify the overall process #16654

Merged

PapaNappa commented on 14 Apr 2020

We are now approaching mid 2020. What is the status and roadmap of this?

Being able to use material in components with ShadowDom opens a lot of opportunities and cleaner components/code/CSS.

☺ 👍 18

andrewseguin added the area: many label on 12 Jun 2020

Devstone11 commented on 25 Jun 2021

Hi folks, just bumping this issue again. Is there a recommended way to use ShadowDom without losing all the styling on Material components?

jelbourn commented on 1 Jul 2021

Angular is dropping IE11 support in v13, which opens the door to us switching to CSS variables, which would make it *possible* to support Shadow DOM.

Sometime in the next year we do plan on revamping our theming system. This would involve switching to CSS variables and aligning more closely with the current Material Design spec, and would probably coincide with moving the MDC-based components to stable. However, we don't yet have the detailed design of what that would look like.

☺ 🚀 2

Assignees

jelbourn

Labels

area: many docs elements help wanted P2

Projects

None yet

Milestone

No milestone

Development

No branches or pull requests

8 participants