

HomeView

FRONT END FRAMEWORKS DAR

Uniting Streaming Services on One Site

March 9, 2022

Unite
Christian Lam
Daniel Monge
Eric Truong (Team Leader)
Erina Lara
Michael Lamera

Introduction

Unite has considered the following three options for HomeView's front-end framework:

- React
- Angular
- Vue.js

	<u>React</u>	<u>Angular</u>	<u>Vue.js</u>
Pros	<ul style="list-style-type: none">● Saving time while re-utilizing components● Virtual DOM enhances experience of users and the work of developers● Open-source library with various tools● Steady code is supplied by one direction data movement● Good for single-page apps● Used by a majority of frontend developers● Good for those who expect high traffic and require a steady platform● User-friendly for new developers	<ul style="list-style-type: none">● Makes coding procedure easier due to its refactoring services and enhanced navigation● Component-based pattern of Angular sanctions forms a UI with single components● Large ecosystem● Angular material reorganizes material design interface production● High performance● Enlarges the execution of browser-based apps by modernizing the contents in less time since it uses two-way data binding	<ul style="list-style-type: none">● Extensive and detailed documentation● Simplicity and clarity● Browser dev tools extensions● Code reusability and simple integration● Lets you scheme whole thing from scratch, effective for big projects● Apt for progressive web apps, dynamic web apps, and complex projects
+0.80	+0.80	+0.60	+0.60

Cons -0.40	<ul style="list-style-type: none"> • Absence of documentation due to significant pace development • Comparatively long learning curve • Developers find it challenging to comprehend the complications of JSX • Need hands-on knowledge with JavaScript -0.40	<ul style="list-style-type: none"> • Angular complication • Relocating legacy schemes from AngularJS to Angular • The CLI documentation is not fairly defined • Learning effort is high -0.40	<ul style="list-style-type: none"> • Reduced developers' community • No support community to respond to the intricacies • Flexibility leads to code irregularities -0.30
Subtotal 0.40	0.40	0.20	0.30
Metrics			
Number of Contributors +0.20	1,342 +0.20	1,035 +0.15	283 +0.05
Number of Downloads +0.20	15,330,334 +0.20	475, 752 +0.05	3,161,261 +0.10
User Satisfaction Score +0.20	88% +0.20	42% +0.05	85% +0.15
Size of Framework +0.20	43K +0.15	143K +0.05	23K +0.20
Total (out of 1.20)	1.15	0.50	0.80

Conclusion

Based on the above metrics and research, the React framework had scored the most points during the review. As compared to the other frameworks, React is the most friendly for new developers. In addition, it is notably easy to utilize for single-page applications, which is ideal for HomeView. React is also the most popular framework when it comes to front-end frameworks, with a large community of contributors and user satisfaction.

For the first calculated subtotal, our team decided to score each individual category's pros and cons. The highest possible number of points to gain from the pros would be +0.80, whereas the highest possible deduction of points from the cons would be -0.40. To calculate each option's subtotal from the recorded pros and cons, the number of cons would be subtracted from the option's total number of pros. In our case, the React option scored the highest (by having +0.80 pros and -0.40 cons, resulting in a subtotal score of +0.40). Following the same procedure, the Vue.js option placed second with a subtotal of +0.30, whereas the Angular option resulted last, with a subtotal of +0.20.

After calculating the subtotal, we analyzed the metrics of each option. For the number of contributors, the React option scored +0.20 due to having a number of 1,342 contributors whereas the Angular framework scored +0.15 for having 1,035 contributors, and Vue.js scored +0.05 for having 283 contributors to the framework. For the number of downloads metric, the React framework had around 15,330,334 downloads, scoring the highest with a score of +0.20. On the other hand, the Angular framework scored +0.05 for having a metric of 475,752 downloads. Vue.js, however, had scored +0.10 for having a metric of 3,161,261 downloads. For the user satisfaction scoring metric, the React option scored +0.20 for having a high user satisfaction rate of 88%. While Vue.js came in close second scoring +0.15 for a satisfaction rate

of 85%, and the Angular framework was last, scoring +0.05 due to a low satisfaction rate of 42%.

For the size of the framework metric, Vue.js scored the highest with +0.20 for being a lightweight framework of only 23K. React scored +0.15 coming in second for having a size of 43K. Angular came in last, scoring +0.05 for having a heavy size of 143K.

After calculating the subtotal and metrics from each framework, the highest possible scoring was a 1.20. The React framework scored the highest, with a score of 1.15, whereas the Angular framework scored 0.50, and the Vue.js framework scored 0.80. As a conclusion, based on its metrics and pros and cons, our team conclusively and unanimously decided to utilize the React framework for the front-end framework of HomeView.

References

<https://www.monocubed.com/blog/best-front-end-frameworks/>

<https://www.freecodecamp.org/news/angular-react-vue/>

<https://2020.stateofjs.com/en-US/technologies/front-end-frameworks/>

<https://www.npmtrends.com/angular-vs-react-vs-vue>

<https://gist.github.com/Restuta/cda69e50a853aa64912d>