r_udokuj: AngularJS [1] vs. Bootstrap [2]

Kunal Kulkarni, Neeraj Deshpande, Sahil Dorwat, Aditya Upadhyay
Department of Computer Science
North Carolina State University
Raleigh, NC, USA
{krkulkar, ndeshpa, ssdorwat, adupadhy}@ncsu.edu

Abstract—**In this project we compared AngularJS (https://github.com/angular/angular) and Bootstrap (https://github.com/twbs/bootstrap). We used various metrics for our comparison, some of which were bus factor, ease of use, start up cost of a sample project, Github metrics etc. Our findings indicate that in general, developers prefer Bootstrap over AngularJS when it comes to ease of use, higher flexibility and lower start up cost. Our research also indicates that when it comes to using a strong, robust platform for front end projects, developers prefer AngularJS over Bootstrap. We also compared bug related metrics like the number of open bugs in each project, the frequency of bug fixes etc. After analyzing our results, we recommend that if developers want to build structured, single page web applications, AngularJS is the framework to choose. AngularJS (launched 2010) has a lot of powerful features, has a larger developer base and since it is a little older than Bootstrap (launched 2011), it has more contributors and releases than Bootstrap. On the other hand, if developers are looking for an easy to use, highly flexible and lightweight framework for front end development, Bootstrap is the way to go.**

I. Overview

Description of project scope (i.e., projects compared, tools applied, metrics considered). We chose AngularJS and Bootstrap as our open source projects because all of the team members wanted to learn about these very popular projects and how developers have contributed to them throughout the years since their inception. We were aware that the fundamental reason why these 2 projects are used by developers is different i.e. AngularJS is used for developing single page web applications, whereas Bootstrap is more CSS and designing oriented. However, we felt that both projects started at around the same time (2010 vs 2011), and were both very popular front end frameworks. So, we decided to conduct our research on these 2 open source projects.

Initially, we had thought of conducting static analysis of the codes of the 2 projects using various linting tools available on the Internet. We had considered using ESLint, which is a pluggable linting utility for Javascript. However, based on the feedback from the Professors and the TAs, we dropped this idea. We also felt that since these projects have been in the market for quite a long time, and have been used so extensively by developers worldwide, linting might not provide us with a lot of useful data, since there is a very low possibility of finding issues with the syntax of the codes. We have described the metrics and factors that we considered in our research in detail in the section below titled ”Approach”.

As for the scope of this project, the project was a semester long project. We set regular deadlines for different milestones all throughout the semester, which ensured that work was being done regularly and at a good pace. By mid October, we had installed both the projects on our machines. We had also installed the tools we used for our analysis, and had configured them properly so that they executed seamlessly on all of our machines. By the start of November, we had conducted our initial analysis of both projects by going on their Github pages, and looking at the various readily available metrics on Github. We had also started looking at each of their documentations to analyze which one was better and clearer to understand. By mid November, we had written a homegrown script to mine data from Github. We extracted various metrics like commit frequency, commit activity, contributors by various developers etc. We made use of the Github API to mine all of these metrics. By end of November, we had conducted an online survey to gain developer’s perspectives on metrics like ease of use, ease of understanding, which one developers would prefer etc.

Also, to get an insight of how the users of these two technologies feel about the them, we conducted a short survey with students of computer science background at NC State University. The survey consisted of questions relating to the ease of use and preference of the developers. Also, the developers commented about the reasons as to why they prefer a particular tool.

The tools that we used were: 1. Github API [3] - to mine all the necessary data from Github. 2. Python [4] - to write our homegrown script for mining of Github data. 3. Postman [5] - to test our script and the JSON requests our script was sending to the Github API. 4. Gittrends [6] - to mine some advanced metrics from Github. 5. CodeClimate [7] - to mine code smells, bug complexities, etc.

GitHub repo containing tool output and homegrown script is present in the footnote: [1]

II. Approach

The first task we did when we got this project was explore a set of candidate projects and shortlisting which projects we wanted to carry out our research on. As mentioned before, we wanted to explore front end technologies and we felt AngularJS and Bootstrap were comparable. Although they

[1] https://github.com/Kunalkul1/Software-Engineering-Project-Fall-2018

are fundamentally used for different purposes, in the sense that AngularJS is mainly used to create single page web applications whereas Bootstrap is mainly used for CSS styling and Javascript utilities, we felt that the tools serve a common purpose and that is to provide a framework for front end development. Their inception was also around the same time, so we thought that the metrics like developer base, commit frequency etc would be comparable.

The next step was to select the tools we would use for our analysis.

Following this, we conducted several meetings to finalize the metrics we would be using for our analysis. We had started out with a few metrics like lines of code, bugs per kloc. But after reading the midterm presentation feedback we received, we added a few more metrics. We agreed with the Professors and TAs when they mentioned that these metrics might have several implications, and it would be wise to use some different metrics for our comparison. We finally came up with a list of 31 metrics we would use. In a broader way, these can be categorized as follows:

- Contribution statistics
- Release cycle/frequency
- Bus factor
- GitHub statistics (issues, pull requests, etc.)
- StackExchange support statistics
- Initial cost analysis
- Support and Documentation
- Bug coverage and severity
- Code analysis (complexity, duplication, etc.)
- Remediation time
- Technical Debt ratio

After finalizing our metrics, we started our analysis on data that was readily available on the Github pages of both the projects. We found some useful data like commit frequency graph, contributors on Github. Next for the Bus Factor metric, we used the GitTrends utility, which readily provided this metric. Furthermore, we mined project metrics from Github like frequency of commits, top contributors, average bug count by severity, bugs per component etc. We developed a script in Python, which extracted data from Github using the Github API by passing JSON data. We tested our script using Postman. In brief, the script was used RestAPI to calculate commit history that is number of commits per week or per day, total number of contributors which was used to calculate the top 5 contributors for the same and extracted the creation and closing date and time stamp which was then used to calculate the bug lifetime. These factors were sorted based on severity of the bug. We also developed a sample application using AngularJS and Bootstrap to test the a few metrics.

As mentioned above, we also conducted a survey among NC State Computer Science students to find out what developers think about the ease of use, ease of learning, documentation clarity etc. We analyzed the responses to the Google form we created, and we have presented our results below. We found that in general, Bootstrap was more popular among developers when it came to easy of use and ease of learning. When it came to preference of projects, a lot of developers chose AngularJS because of its power, robustness and functionality.

Thus, we were able to gather sufficient data for comparison of both the projects. This project was a great learning experience for all of us, and we understood the importance of analyzing the different metrics of projects before using them in projects.

III. Conclusions

The experiments lead to a number of insights. Firstly, Bootstrap holds an upper hand in terms of preference based on the survey. The key factors that result in this result are the ease of use, ease of learning and simplicity of implementation. But the survey is mainly based on insights of student audience and hence does not involve ideas for large scale development. In such cases, AngularJS is highly preferred based on its higher feature set and community support. This is observed by the fact that few people who preferred AngularJS had a prior experience with large scale projects.

Considering bus factor, AngularJS has a bus factor of 6 while Bootstrap has a factor of 2 for their master branch respectively. A higher bus factor indicates that even though a large number of significant contributors exist, but in case of a contributor moving out of the project, the project will not be impacted as heavily as in case of lower bus factor. For example, if one out of the two major contributors of Bootstrap were to leave, it would have a huge impact while if one of the six major contributors were to leave the AngularJS project, the impact would be lesser.

As discussed earlier, AngularJS has a steeper learning curve compared to Bootstrap. This makes it costlier for a developer to learn and implement a particular project in AngularJS. Hence, Bootstrap has a lower starting cost. Considering the bug count by severity, the data for AngularJS is available on Github, however, there was not enough data to be found for Bootstrap and hence this metric could not be used for comparison. Additionally, the documentation for both of them is quite vast.

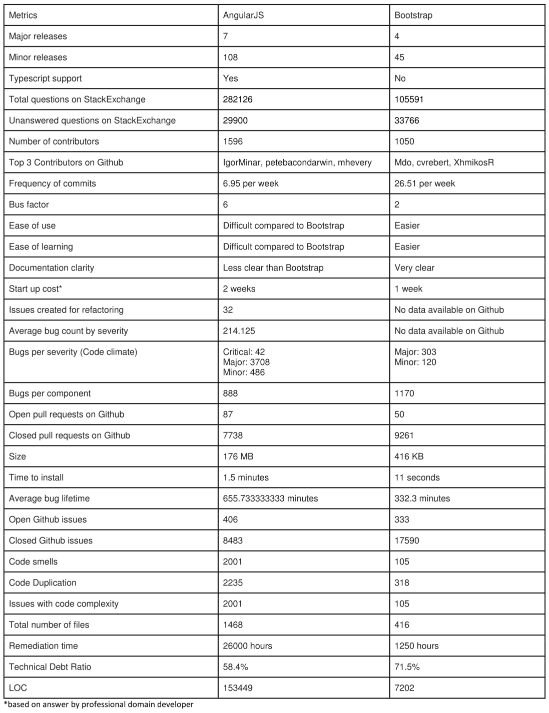
The major and minor releases for AngularJS is quite more as compared to Bootstrap. For AngularJS there is a major release after every six months and there are three minor releases between two major releases. For Bootstrap, major release is every 2 years and minor release being 2 to 3 times between every major release. AngularJS supports typescript while Bootstrap runs on JavaScript. Also, AngularJS has a backward compatibility of one year while Bootstrap has a longer compatibility period.

Usually the developers in the survey were primarily students, people preferred bootstrap over AngularJS in ease of learing and usage. However, there was almost equal result for which tool they are likely to use.

Hence, AngularJS is preferred in large scale modular applications. On the other hand, if developers are looking for an easy to use, highly flexible and lightweight framework for front end development, Bootstrap is should be the preference.

RESULTS

Figure 1 illustrates the extensive study of comparison of Bootstrap and AngularJS.



Metrics	AngularJS	Bootstrap
Major releases	7	4
Minor releases	108	45
Typescript support	Yes	No
Total questions on StackExchange	282126	105591
Unanswered questions on StackExchange	29900	33766
Number of contributors	1596	1050
Top 3 Contributors on Github	IgorMinar, petebacondarwin, mhevery	Mdo, cvrebert, XhmikosR
Frequency of commits	6.95 per week	26.51 per week
Bus factor	6	2
Ease of use	Difficult compared to Bootstrap	Easier
Ease of learning	Difficult compared to Bootstrap	Easier
Documentation clarity	Less clear than Bootstrap	Very clear
Start up cost*	2 weeks	1 week
Issues created for refactoring	32	No data available on Github
Average bug count by severity	214.125	No data available on Github
Bugs per severity (Code climate)	Critical: 42 Major: 3708 Minor: 486	Major: 303 Minor: 120
Bugs per component	888	1170
Open pull requests on Github	87	50
Closed pull requests on Github	7738	9261
Size	176 MB	416 KB
Time to install	1.5 minutes	11 seconds
Average bug lifetime	655.733333333 minutes	332.3 minutes
Open Github issues	406	333
Closed Github issues	8483	17590
Code smells	2001	105
Code Duplication	2235	318
Issues with code complexity	2001	105
Total number of files	1468	416
Remediation time	26000 hours	1250 hours
Technical Debt Ratio	58.4%	71.5%
LOC	153449	7202

*based on answer by professional domain developer

Fig. 1. Results table

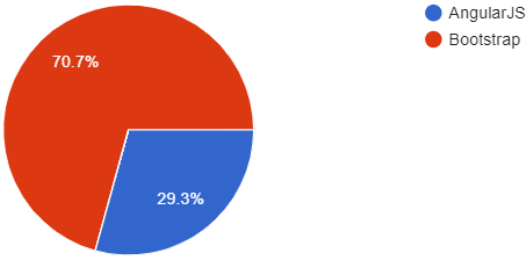


Fig. 2. Survey: Easier to Learn

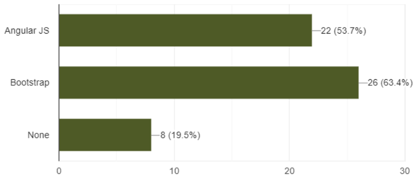


Fig. 3. Survey: Previously Used

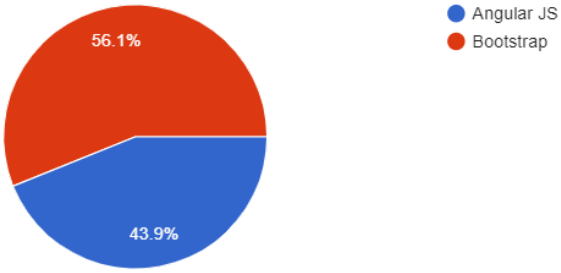


Fig. 4. Survey: Overall Preference

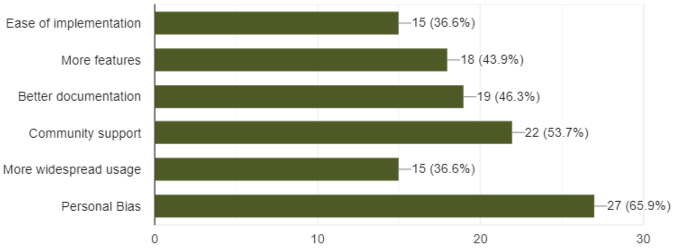


Fig. 5. Survey: Reason of Preference

The survey conducted mostly included Masters students studying at NC state. There were a total of 41 responses. Even though we can see from the survey that Bootstrap is easier of use as compared to AngularJS whereas the overall preference of AngularJS is quite same as that of Bootstrap due to personal bias, better features and community support.

The python script that used the GitHub and stackexchange APIs used to mine the results of github and questions asked on stackexchange, is present on https://github.com/Kunalkul1/Software-Engineering-Project-Fall-2018 link. The output of the script can be obtained in the same repository. Also, the authentication of the results is present at the same location. The sample project which was used to compare the results is also present in the same directory.

Additionally, these results were calculated on the AngularJS and Bootstrap projects forked from the original repositories.

Future work may involve developing individual projects on AngularJS and Bootstrap and comparing and contrasting them. Also, it would be a good idea to compare Bootstrap with some other CSS frameworks like Foundation and AngularJS with other frameworks like ReactJS.

REFERENCES

[1]AngularJS Available at https://github.com/angular/angular.

[2]Bootstrap Available at https://github.com/twbs/bootstrap.

[3]GitHub API: GitHub Inc.

[4]Python Software Foundation. Python Language Reference, version 2.7. Available at http://www.python.org

[5]Postman: PostDot Technologies. Available at https://www.getpostman.com/docs/v6/reference/

[6]GitTrends. Available at http://gittrends.io/ .

[7]CodeClimate Available at https://codeclimate.com/.