TEXT EDITORS: IMPORTANCE, PREFERENCES, AND IMPROVEMENTS IN I.T. INDUSTRY

Benya Chongolnee
Purdue University
ENGL 420
Spring 2017

Memo of Transmittal

To: Devon Cook
From: Benya Chongolnee
Subject: Overview of white paper
Date: March 31st, 2017

I am writing this memo of transmittal to describe what this white paper contains as well as why it is important for everyone to read and understand. This white paper contains facts from an interview with an Information Technology professional in 2017, a survey that was passed completed by seventeen Purdue University undergraduate students in 2017 and a few online resources.

The 21st century is filled with technology and fun gadgets for people to use to make their lives better and easier. None of that technology would work without programming; however, programming would not be feasible without a programming text editor. This is similar to a book since a book can not be a book without Microsoft Word since it is a place for a book to be written in. A programming text editor is a place for programmers to write their code and it offers a variety of functionalities to make programming more efficient and faster.

Throughout this white paper, I will mention the following:

- Why this matter
- Functionalities of each of the text editors
- Problems of current text editors
- Ways to improve text editors

The current text editors have a lot of great functionalities; however, there is a lot of room for improvement. I have a proposal to fix this issue in the Information Technology world. Text editors still have a long way to go in order to make programming faster and more efficient. With my proposal, I will be able to make programming more efficient as well as attracts more people to want to learn how to code.

This white paper will be able to convey the readers the importance of text editors and how it can change the way technology is created. On top of this, this paper will be able to teach some readers about some of the existing text editors or attract them to want to learn about programming. For any questions or concerns, please feel free to reach out to me at bchongol@purdue.edu.

Table of Contents

Executive Summary

Imagine the world without technology. That is the world without text editors. Text editors are very important in today's society because without a text editor, there would be no place to write a computer program. And without a computer program, there will be no technology to make lives easier and better. There are various types of text editors that exist out there and each one of them has a lot of great functionalities that convey a certain person to use that text editor.

Why does this white paper about text editors matter for anyone? First, programmers should know which text editors are the best to use for projects. They would want to know which text editors have certain functionalities that will make programming easier and faster. However, for non-programmers, they should read this white paper to be more knowledgeable about programming and current text editors.

There are a lot of text editors being made in order to compete with other text editors or to create more functionalities. Each one of the text editors has limitations; however, they also have their own unique functionalities. This can range from live previews to bracket pairing. With that being said, there is a lot of room for text editors to be improved since there are still limitations.

To improve the efficiency of programming, I am proposing a new text editor that will combine all of the positive functionalities of the current text editors. The new text editor will include functionalities such as a variety of languages, free and usable on every operating system, customizable, live previews, auto indentation, bracket pairing, auto filling, color coded, ease of use, and the ability to share code in real time. The new editor will be able to contain all of the great functionalities of current text editors while no longer have the limitations that current text editors may have. All of the great functionalities combined will be able to not only make programming more productive, it will also attract more people to want to learn how to code. The more programmers there are, the more technology that will be created to make everyone's lives easier.

1. INTRODUCTION

Almost everything in this world uses technology. This varies from cars, telephones, and microwaves, but people do not always think about what makes technology work the way it should. One or several people come together to write several lines of code to tell technologies how to behave. In order to write the code, a text editor must be used. Text Editor, according to Napier, Rivers, and Wagner (2006), means a "software that allows you to create text documents but lacks the special document creating and formatting features found in a word processor". Microsoft Word, Google Docs, and Notepad are all considered a text editor. All of these are very important in order to convey a message or create something meaningful.

For the purpose of this white paper, I will be referring to a text editor as software that allows programmers to edit their programming code, and it may offer various functions such as a compiler or color coding. I, an Information Technology major at Purdue University, have used many different text editors and personally understand the advantages and the disadvantages between all of them. On top of this, for this research I conducted an interview with an I.T. professional and conducted a survey through Qualtrics in 2017 that was passed to seventeen different I.T. majors whose interests varies from software developing to designing.

2. WHY DOES THIS MATTER?

The knowledge about text editors is very important since most of the technology requires code. A programmer, for either several years or just starting, would want to know which text editors are best and why. Non programmers should still read this white paper to be more knowledgeable about programming. This white paper will include some of the current text editors and some of their functionalities, its problems, and suggestions that could improve current text editors.

As mentioned before, text editors are incredibly important in order to create computer programs just like how Microsoft Word and Google Docs are important when writing an essay. Text editors is a place to write code and have the computer understand it. Without text editors, there will not be a place for computer code to be written in at all.

Furthermore, the person that I interviewed for this research is an Information Technology (I.T.) professional for over 30 years. He has many experiences with programming and knows many people in the I.T. industry. My interviewee emphasized that he uses his prefered text editor every day to either write some type of script code or modify source code. His co-workers also use roughly the same amount or more (2017, personal phone interview). This shows the importance of text editors in I.T. and how much it is being used each day.

3. TEXT EDITORS

3.1 Functionalities

When I asked my interviewee how his preferred text editor, TextWrangler, is different from the others, he responded “I don’t think [TextWrangler] is very different [than other text editors]. It depends on [how] fast or [if it can] check your code [in] real time” (2017, personal phone interview). This means that for him, current text editors are very similar. A lot of text editors have similar functionalities and capability; therefore, there are no such thing as the best text editor, it solely depends on the individual’s preferences and what the text editor is used for. From this, I wanted to break down the data to show what some of the text editor’s functionalities are. The following chart are the most popular text editors and more text editors can be found online.

Text Editor	**Built-in?**	**Price**	**Color Coded**	**Languages**
Atom	No	Free	Yes	PHP, Javascript, HTML, CSS, Sass, Less, Python, C/C++, etc
Brackets	No	Free	Yes	HTML, JS, CSS, etc
Dreamweaver	No	~$19.99/mo	Yes	C#, HTML, CSS, ASP, Javascript, PHP, Visual Basic,etc
Geany	No	Free	Yes	C/C++, C#, Java, JavaScript, PHP, HTML, LaTeX, CSS, Python, Perl, Ruby, Pascal, Haskell, Erlang, Vala, etc
gedit	No	Free	Yes	C, C++, Java, HTML, XML, Python, Perl, etc
GNU EMACS	No	Free	Yes	C
GNU NANO	No	Free	No	C
IntelliJ IDEA	No	Free - JVM & android development ---------- ~$399.00 - Web & enterprise development	Yes	Java, Python, CSS, HTML, Groovy, SQL, Ruby, Perl
NetBeans	No	Free	Yes	Java, HTML, Javascript, C/C++, PHP
Notepad	Yes -	Free	No	HTML, CSS, Javascript,

	Windows Only			Python, C++
Notepad++	No	Free	Yes	HTML, CSS, Javascript, Python, C/C++, C#, Batch, PHP, Perl, etc
Sublime Text	No	Free ---------- $70.00- for license	Yes	ASP, HTML, CSS, Javascript, Python, C/C++, C#, Batch, PHP, Perl, Java
TextWrangler	Yes - Mac	Free	Yes	HTML, CSS, Javascript, Python, Perl, Shell, etc.
VI	No	Free	No	C
Visual Studio	No	Free - for students, open source, and individual developers ---------- ~$499-$1199 - other	Yes	C, C++, VB.NET, C#, Python, Ruby, Node.js, M

Functionalities Chart (continued)

3.2 Specifics

For this part of the white paper, I will go into more details for some of the text editors mentioned in the chart. In order to do this, I did some secondary research on top of my interview and survey in order to back up my reasonings.

Atom

Atom is a more popular text editor among the advanced programmers. According to Adarsh Verma, Fossbytes' co-founder, stated that Atom is "the text editor of 21st century, it's a modern text editor that's hackable to the core". (2016). Atom is customizable and offers thousands of open source packages. This text editor is hackable according to one of the person that took my survey in 2017. The person stated that it is very flexible. This text editor is perfect for those that are more advanced in programming and would like more flexibility in their text editor.

Brackets

Brackets is similar to a lot of other web developing text editors out there. This text editor was launched by Adobe back in 2014 for free, according to Verma in his article in 2016. Just like a lot of other web developer text editor, it offers live preview and inline editing. Ryan Stewart mentioned in his personal blog that Brackets offers "quick edit, jump to definition, and live highlight…[quick edit] means [that] you can jump directly to your preprocessor code where the styles are actually defined, edit those, recompile them, and see the changes" (2014). This makes Brackets more efficient since the user does not have to waste time in order to switch between different files.

Dreamweaver

Dreamweaver is one of the few text editor that has a monthly charge despite the fact that their features are very similar to other web developer text editors such as Brackets. However, since the people I surveyed attend Purdue University, they are able to use Dreamweaver for free. They stated that they like using dreamweaver because of the live code previews, autocomplete, and bracket highlighting when in fact, other many web developer text editors also have the same features. One of the participants mentioned that Dreamweaver "takes awhile to load [and] live preview sometimes doesn't work" (2017, survey). Because of this, there are a lot of different web developer text editors out there that are much more efficient than Dreamweaver.

Notepad

Notepad is familiar to many because it is installed automatically on every Windows computer. Its main purpose is for users to take notes. Despite this, it can be use for programming as well. One of my survey participants said that they like using notepad for quick programming because it is built into the computer (2017, survey). On top of this, my interviewee mentioned that Notepad "is fast because [it is a] small program [therefore it] loads faster" (2017, personal phone interview). Notepad is great for small programs that do not require many functionalities since it is readily accessible to use especially when time is limited.

Notepad++

Four out of seventeen people that participated in my survey mentioned that their most used text editor is Notepad++. According to one of my survey participants, Notepad++ "uses low resources, is feature rich, and looks nicer [than other text editors]" (2017, survey). Notepad++ offers variety of features such as auto indentation to help coding be easier. Auto indentation means that certain portions of the code, such as in between brackets, are indented to make the code more organized and easier to read. It is also color coded to help the users organize their code better and for the code to stand out more. However, one interviewee stated that one bad thing about Notepad++ is that it does not have a compiler. Most languages require a compiler in order to run the code. Without a compiler, code is just a few words and numbers.

Sublime Text

Sublime Text is one of the most popular text editors out there. A survey participant stated that they enjoyed using Sublime Text because it is very simple, reliable, and has add-ons to increase programming productivity (2017, survey). Verma stated in his article that Sublime text uses plugins "that are maintained under free-software licenses" (2016) . Sublime text is free to download; however, according to one of my survey participants, it often asks the programmer to buy a license (2017, survey).

TextWrangler

A survey participant stated that they use TextWrangler regularly because it is very basic and easy to use (2017, survey). On top of this, my interviewee stated that "90% [of the time] TextWrangler can do" a command (2017, personal phone interview). He has been using TextWrangler for the past 10 years and have never tried any other text editors. He stated that he likes the fact that it has code verification meaning that it shows him what part of the code is giving him errors. This help him figure out what is wrong with his program easier and faster. Having a code verification greatly benefit programmers since it reduces the time for programmers to try to figure why their code do not work.

4. PROBLEMS

There are a lot of problems with the text editors that currently exists, despite that fact that many people in this world use it. Gomez, Egan, and Bowers (1986) stated in their collaborative Human-Computer Interaction journal, "the fact that people in so many different occupations must learn to use text editors also suggests that individual differences in learning will be sizable". What they meant by this is that the ability for different people to learn text editors will be different per person. Text editors can be hard to learn, and if one gives up on learning how to use a text editor, they will not want to learn how to code at all.

From doing this research, some text editors such as Dreamweaver can be very expensive, especially for students that are just getting started on coding. If something requires a lot of money, a lot of people will turn away from it; therefore, a lot of people might not want to learn how to code for this reason. On the other hand, advanced programmers would prefer more flexibility in their text editors than new programmers. Some programmers may need extra functionalities that could be confusing for others. On top of this, some text editors only offer one computer language such as C. This gives limitations for the programmers since they can only use the text editor for only one language.

Lastly, collaborating in the I.T. industry is very difficult because the only way that codes can be shared is for the file to be put on a website such as Github or sent through email. This is a big problem in the I.T. industry because if a feedback is made, the programmer need to fix the program on their own computer, then put the file on the website or send it through email again.

5. PROPOSED SOLUTION

Since there are no text editors currently that are perfect, a perfect text editor should be made in order to make programming easier and more efficient. By combining all of the positive functionalities of the text editors that exist today as well as adding more functionalities, the text editor will be a perfect solution for programmers to utilize. There are a few requirements for this perfect text editor.

First, the text editor must offer a wide variety of languages. The top languages that it should provide are Java, C#, C/C++, HTML, CSS, Javascript, Python, PHP, and Perl. The more languages it offers, the better the text editor will be. This avoids programmers needing to install a different text editor just because one does not provide the desired language. The text editor should also be free so that programmers do not have to worry about the cost of the text editor. Doing so will be able to attracts more people to want to learn how to code. On top of this, it should be customizable for the users meaning that the user can modify the text editor to their preferences to provide flexibility. Extensions can be provided as an open source to make the text editor be more customizable. To make the text editor the most effective, having live previews

will greatly benefit the programmer. Live previews will show in real time what the program will look like when the program is compiled. This will help the programming be faster since the programmer can see the changes right when they write the program. Just like most existing text editors, it should contain auto indentation, brackets pairing, auto fills, and color coded. Doing so will help the program be more organized, easier to read, and reduce common mistakes. Furthermore, the text editor should be easy to use and provides tutorial for users if they need it. This will help new programmers focus more about learning how to code than learning how to use the text editor. Most importantly, the text editor should offer the ability to share live code with various people. This will make collaborating and receiving feedback easier in the I.T. industry.

Having an effective text editor that includes all of the positive functionalities of existing text editors will be able to make programmers create programs faster as well as easier for those that are new to programming. The text editor will be able to improve productivity in the programming world and attract younger crowds to learn how to code. The more people that are interested in programming, the more technology that will be made. All of the technology that will be made will have the potential to make everyone's lives easier and better.

6. CONCLUSION

Text editors are very important in the 21st century because without text editors, code will simply not exist. There are many text editors out there that exist today and each one of them has great functionalities such as brackets pairing and flexibility. With that being said, current text editors have a lot of problems that limit programmers to create the best code.

To improve the problems that many text editors face, I am proposing a new text editor that will combine all of the positive functionalities of the current text editors. The new text editor will include functionalities such as a variety of languages, free for every operating systems, customizable, live previews, auto indentation, bracket pairing, auto filling, color coded, ease of use, and the ability to share code in real time. All of the great functionalities combined will be able to make programming more efficient and faster while no longer have the problems that current text editors may have. The new text editor will also be able to attract more people to want to code since the text editor are very easy to use.

REFERENCES

Gomez, L. M., Egan, D. E., & Bowers, C. (1986). Learning to use a text editor: some learner characteristics that predict success. *Human-Computer Interaction*, 2(1), 1-23. Retrieved from http://dl.acm.org/citation.cfm?id=1453769

Napier, H., Rivers, O., & Wagner, S. (2006). *Creating a Winning E-Business* (2nd ed.). Retrieved from https://books.google.com/books?id=IZQFAAAAQBAJ&pg=PA330&dq=%22notepad+software%22+%22text+editor%22&hl=en&sa=X&ved=0ahUKEwiYx4OV6PfMAhVDPT4KHc2xAi8Q6AEINDAA#v=onepage&q=%22notepad%20software%22%20%22text%20editor%22&f=false

Stewart, R. (2014, September 11). Preprocessor Support in Brackets. *Ryan Stewart.* Retrieved from http://ryanstewart.net/brackets/adobe/preprocessors/2014/09/11/preprocessor-support-in-brackets/

Verma, A. (2016, December 20). 9 Best Text Editors For Linux And Programming | 2017. *Fossbyte.* Retrived from https://fossbytes.com/9-best-text-editors-linux-programming-2017/