

Technologies for Super Micro Websites: Itty.bitty vs. Docs365 Itty Bitty

Yuta Nemoto*, Vitaly Klyuev**

*Department of Computer Science and Engineering, The University of Aizu, Japan

**Graduate Department of Computer and Information Systems, The University of Aizu

s1240234@u-aizu.ac.jp, vklyuev@u-aizu.ac.jp

Abstract— In this paper, we discuss the “Itty.bitty” technology. It is a new way to share HTML/text information. We pay attention to the effectiveness of its tools by comparing it with the Docs365 Itty Bitty from the point of view of variety and richness of expression. To make a comparison, we redesigned the existing academic laboratory Website and encoded it into the Itty.bitty URL and Docs365 Itty Bitty document. The following features were selected as criteria for this comparison: the implementable components such as textual, graphical, audio, and visual items; tools to adjust fonts, screens, and graphical elements; instruments to design the Website presented in these technologies, methods to deploy Websites, size Websites in KB, speed to appear in the browsers, type of browsers for viewing Websites, opinions of the users on the design, etc. Technological steps to create Websites are as follows: For the Itty.bitty URL, we can convert the HTML directly on the Itty.bitty Website. Once we put the target string (In this case, HTML), it automatically converts the string into the encoded URL. On the other hand, for the Docs365 Itty Bitty, we can do it by redesigning the content of a particular web page into a Google Docs document and translating it into Itty.bitty URL utilizing the Docs 365 Itty Bitty add-on. Our general conclusion on the results of this comparison are as follows: These technologies have their own niches. Applications implemented in Itty.bitty are suitable for super small Websites with very limited features. An approach to utilize Docs365 Itty Bitty gives more freedom, but there is a number of restrictions, and the developer needs to check them in the case by case manner.

Keywords— Itty.bitty, Website, Development, Deployment

I. INTRODUCTION

Itty bitty [1] is a cutting-edge technology in the field of the Web, which makes it possible to run a very small website without any server to host the pages. This technology encodes the content text or HTML into an URL by compressing the data using the Lempel-Ziv-Markov chain algorithm [2], and encoded URL can be shared directly with any users. When users view this encoded web page, their browsers access itty.bitty.site to decode the data and gets the main text or HTML. The advantages of this technology are in its simplicity, independence, and privacy. Particularly when users share a small text or web page such as their own profile, message or poetry on a page, it provides a superior convenience. Also, as a point differs from the general web contents, only people who have the URL can see the published contents. It means the

content is not available for unspecified viewers. It is also efficient to limit the usual risk on the websites.

The general approach to obtain the Itty.bitty URL is just to visit Itty.bitty.site and use the function provided on this page. By pasting the content text or HMTL to the form, this function gives, it automatically compresses the data, encodes the information into an URL, and gives the result. Because of the outstanding usability of this technology, there is a product called Docs365 Itty Bitty. It is the add-on working on the Google Docs and providing a function to encode the Docs document into Itty Bitty URL. Actually, this add-on works as follows: transforming the Docs document into the HTML document and encoding the transformed HTML into the Itty Bitty URL, but users can obtain the Itty Bitty URL of the particular document only by clicking a button on the add-on menu.

In this study, we make it clear which of these tools is more useful as an expression method, according to the criteria we state in section 2. Finally, our goal is to reveal the features in each method and clarify what kind of condition or content each approach stands out its characteristics.

II. METHODS

To compare the features and effectiveness of the products generated by Itty.bitty.site and Docs365 Itty Bitty, we redesign existing website [3] and express it in the micro website using these Itty.bitty tools.

The website for redesign especially has the features: bilingual expression without reloading, access counter function, and responsive design to support the view from multi-devices. In order to implement a bilingual feature, it has a switch-button on the header navigation bar. The website contains a part which called an external CGI and CSS program to realize access counter and responsive design.

In this development, we selected the following criteria to compare these features:

- 1) components that can be implemented by the particular tool;
- 2) instruments to design page contents;
- 3) tools to adjust the page components;
- 4) methods to deploy the website;
- 5) the data size of the converted website;
- 6) speed to appear in the browser;

- 7) browsers which are able to view the website;
- 8) user opinions for the whole document design.

Item 1 and 2 specify whether the particular component can be expressed in the webpage or not and how rich designs can be realized in each method. Item 3 specifies whether there is any general method to adjust the component's size, placing, etc. or not. Here we investigate the flexibility of the product converted by each method. Item 4 is about the way to deploy, we can discuss its complexity in each method. We characterize the way to deploy, below in this section. Items 5 and 6 are about the converted product in each method, we can recognize the method efficiency aspect. Item 7 is about what the available browsers to see the web page with each method. The final item is about the whole design, we can discuss which is better as a document.

For the generation of websites, we prepare the product according to the following processes.

A. Method to generate the micro website using itty.bitty.site

To make a website with the itty.bitty.site tool, we prepare an HTML file for converting. We redesign the existing website to a single page website. We should check out what part of the original website can be implemented on this single page and what part cannot be converted. Especially, if the original website consists of multiple web pages, we need to reorganize the links carefully. Also, if the webpage is using any external resources such as images, CSS file or JavaScript file, we need to change the accessing path from relative to absolute.

After we prepare the HTML file, we visit itty.bitty.site [1], put the HTML code and convert it into an URL. Then this automatically generated URL including the information of the whole HTML page.

B. Method to generate the micro website using Docs365 Itty Bitty

To make a website with tool Docs365 Itty Bitty [4], we prepare a Google Docs document. For this case, we create a similar document including images and placing layouts to the original web page content. In this reprinting process, we get the awareness about the available expression in the Google Docs functions.

After creating the Google Docs document, we use the add-on, Docs365 Itty Bitty to convert the document into Itty.bitty URL. This add-on actually generates the URL in the following way: It converts the Docs document into HTML and then converts the generated HTML into itty.bitty URL. The point is that once it generates HTML, it is not optimized for the particular device/screen size. Because of that, it may occur some unexpected misplacing.

III. RESULTS

In this section, we present the Itty.bitty products generated according to the procedures described in section 2 and the summary of the comparison.

First, we show the screenshots of each produced website (The original website is available at: <http://web-ext.u-aizu.ac.jp/labs/is-se/>). Fig. 1 shows the Itty.bitty product

generated by itty.bitty.site [1] and Fig. 2 shows the one generated by Docs365 Itty Bitty [4]. Details on comparison are presented in the following subsections.

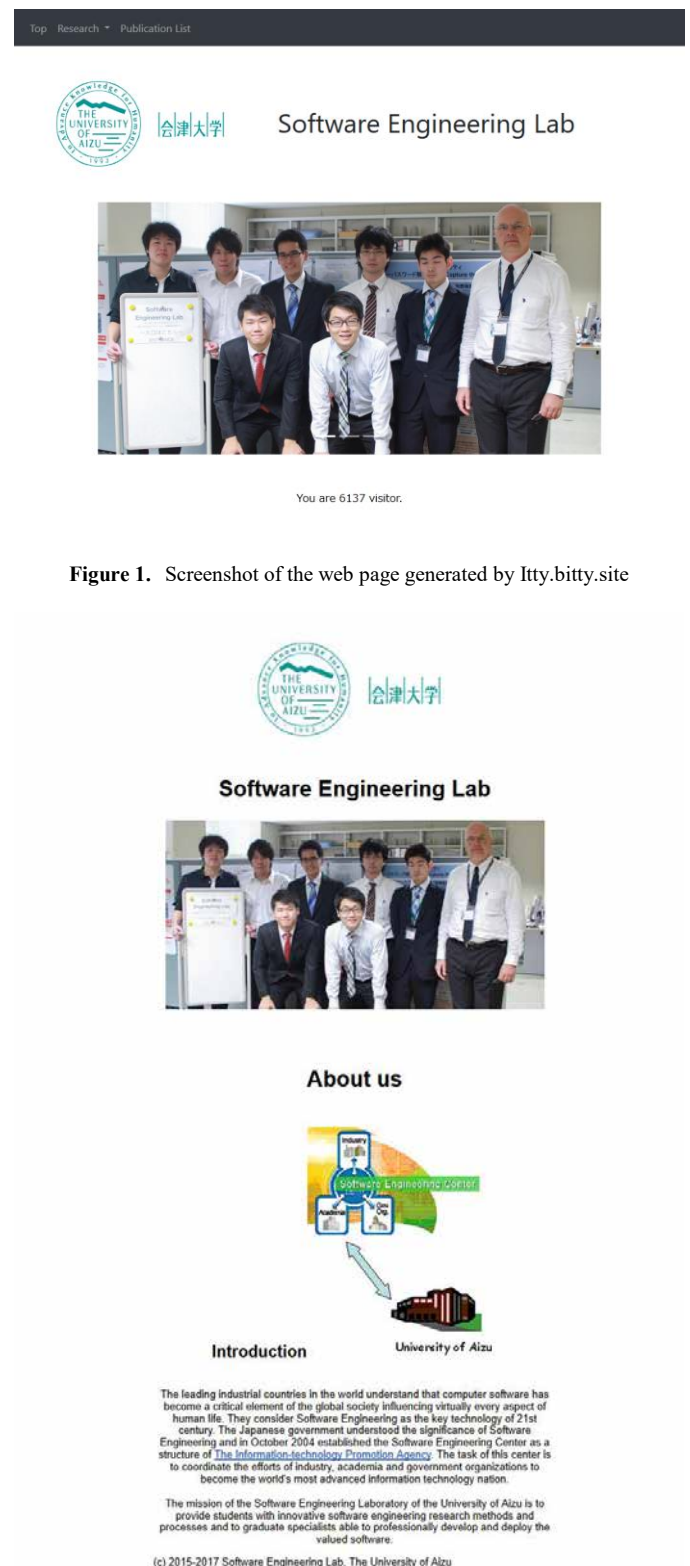


Figure 2. Screenshot of the web page generated by Docs365 Itty Bitty

A. Comparison of the Implementable Components

We analyze 5 kinds of components: Textual, Graphical, Audio, Visual and Dynamic. Details are below.

- 1) **Textual Items:** In the Itty.bitty.site product, the texts are originally prepared as an HTML document. Because this method makes Itty.bitty URL by converting the given HTML directly, we need to prepare the original texts in the HTML file, and the elements are classified by the header, paragraph, etc. To express a plain text on the HTML, we need to write it by surrounding the text with a couple of marking tag. For example, just for expressing the text "Hello World", it is needed to be written in the form: "<p>Hello World</p>". Actually, this small tag makes it possible to define the text structurally. On the other hand, for the Docs365 document, the content texts are originally prepared as a docs file. It means that texts are not classified, but written in a free format by the Google Docs' GUI.
- 2) **Graphical Items:** In the Itty.bitty.site product, this feature is applied by bringing the resource from the external web by using the tag: . Conversely, in the case of the Docs 365 product, we can insert images, simple figures or graphs directly to the Google Docs document.
- 3) **Audio Items:** For the Itty.bitty.site web page, it is implemented as an audio element in the HTML file by referencing external audio resources in the tag:


```
<audio controls>
  <source src=[resource URL] type=[source type]>
</audio>
```

 or by the operation from JavaScript program. On the contrary, for the Docs365 web page, it is impossible to embed audio file in a general way.
- 4) **Visual Items:** For the Itty.bitty.site page, visual items can be implemented by CSS. Actually, as the general way, we can manage the items such as the background color of the page, header/footer design, placing layout, etc. On the other hand, for the Docs365 page, visual features such as text effects or other simple designs including background color are given as the fundamental functions of Google Docs.
- 5) **Dynamic Items:** In the original web page [3], there is an item which needs dynamic action: Access counter. In the original one, it is implemented by calling the content which accesses the other CGI program running in the web server. For the Itty.bitty.site page, there are no difficulties to implement this because we can embed the same function in the HTML with JavaScript code. Contrary, for the Docs365 Itty Bitty page, it is impossible in a general way, because such dynamic components cannot be implemented in the normal Google Docs tool.

B. Comparison of the instruments to design the page

For the Itty.bitty.site web page, generally the main tool to design the HTML components is CSS. Normally this CSS program is separated from the main HTML program and these design features are applied by calling the particular CSS in the HTML. The JavaScript components are also available for making a dynamic web page. On the other hand, for the Docs365 web page, normally we can use the basic Google Docs tools.

C. Comparison of the tools to adjust components

We analyze tools to adjust font, screen, and graphical elements. Details are below.

- 1) **Font Adjustment:** For the Itty.bitty.site web page, font adjustments including font type and effects can be managed by CSS. For example, when we change the font family and size of a paragraph, we can write "font-family: '[font family name]'; font-size: [relative or absolute value];" to the style part of the particular <p> tag. Contrary, for the Docs365 web page, font type or effects in the document can be changed by standard function in Google Docs.
- 2) **Screen Adjustment:** The difference between the Itty.bitty.site page and the Docs365 page for the point of adjusting screen is whether it can provide responsive web design in the original website [3] or not. In the itty.bitty.site product, it can get the screen resolution when it is shown in the browser and adjust sizing and placing of the page components. By this feature, it can provide an appropriate web page for the particular screen size. On the other hand, there are no functions to adjust the screen size or components placing on the Docs365 tool. It means Docs365 page always has the same format in every screen sizes.
- 3) **Adjustment of Graphical Elements:** For the Itty.bitty.site web page, the tool used for adjusting graphical elements' size or placing is CSS normally. By referencing external CSS file on the web or writing CSS elements in the HTML document, we can manage the parameters of the components on the browser directly. In contrast, for the Docs365 web page, we use the Google Docs basic functions to adjust the graphical elements. Here, the important thing is that we cannot operate the product's actual component size on the browser, with the Docs365 tool, because the conversion from docs file to the HTML file is done automatically by this add-on. It means that places for images, resizing the component, etc. are up to the add-on program.

D. Comparison of the method to deploy the website

As we mentioned earlier, the Itty.bitty.site web page is generated by the following operation:

- 1) Write an HTML document for the particular web page.
- 2) Put the code in the form of Itty.bitty.site [1].

- 3) The site automatically converts the HTML to the Itty.bitty URL.

The Docs365 web page is generated by the following operation:

- 1) Create a docs document in Google Docs.
- 2) Open the add-on Docs365.
- 3) Click “Create itty bitty site...” button, then it automatically prepares an HTML from the document and generates the Itty.bitty URL automatically.

The structural difference is that the source HTML program before it is converted to Itty.bitty URL is written by a human or program.

E. Comparison of the data size of the itty.bitty products

We present the actual data size of the Itty.bitty URL and the HTML file before it converted to the URL in Table 1.

Here, HTML refers to the page before it is converted to Itty.bitty URL, and Encoded URL literally means the URL encoded by each method. As a result, obviously, both of the URL and HTML file with Docs365 has a quite large data size.

TABLE 1. DATA SIZE OF THE PRODUCTS

Product	Itty.bitty.site	Docs365 Itty Bitty
Encoded URL	3.46KB (3,541 characters)	328KB (336,005 characters)
HTML	8KB	344KB

F. Comparison of the speed to appear in the browser

In this subsection, we present how fast we can see each web page in the browser. As a result, there is an obvious difference in the appearance speed between the Itty.bitty.site page and Docs365 page. Just for the web page texts and design contents, Itty.bitty.site page appears definitely faster than the Docs365 page. But we observed that there is no significant difference between them to appear the whole web page. Although Itty.bitty.site page appears faster than another one, there is a latency until the picture component is shown in the browser. On the other hand, even the Docs365 page appears slower, the whole web page including text, design contents, images, etc. come at the same time.

G. Comparison of the available browsers to view pages

We check whether page viewing works on major browsers on PC: Chrome ver. 70.0.3538.110, Firefox ver. 63.0.3, Microsoft Edge ver. 17.17134, Internet Explorer ver. 11.0.95, and Safari 11, and browsers on Smartphone: Chrome ver. 70.0.3538.110, Firefox ver. 63.0.2, and Safari 10.

For the Itty.bitty.site, we can see it on all browsers except for two Microsoft browsers: Edge and Explorer. We tested on both PC and Smartphone. For Docs365 web page, it is available only on two PC browsers: Chrome and Firefox. On the other browsers on PC and Smartphone browsers, the content of the Docs365 web page cannot be seen.

H. Comparison of the whole document design

The Itty.bitty.site page looks much better than the one of the Docs365 page, because the document converted by

Itty.bitty.site is manually written in HTML, and can be operated more flexible with CSS and other functions. On the other hand, there is no malleable adjustment for the Docs365 page. For example, the size of the page components is not adjusted even when the screen size has been changed. It may occur some unexpected place-changing of components because the Docs format is changed into the totally different format: HTML, by add-on's automatically adjusting. But from the point of view of the difficulty when we create the design, Docs365 Itty Bitty is easier for people who are not familiar with HTML/CSS document writing. This method makes it possible to create the contents with GUI for general people also. Data in Table 2 summarise presented results.

IV. DISCUSSION

In the previous section, we summarize the results of comparison according to criteria. We can see 2 issues which make the obvious differences between them: the main tool to implement each document design and the document structure in each product.

For the point of the main tool, the difference between the Itty.bitty.site and Docs365 Itty Bitty is that Itty.bitty.site page is mainly designed by CSS written by a programmer, but Docs365 page design is the converted one from Google Docs document by add-on program. Actually, that is why there is the obvious design-level difference between 2 products. We can apply various kinds of design characteristics including Header/Footer designs, fonts or backgrounds, in the implementation with CSS, and dynamic components including Access Counter or Picture Slide are also available by utilizing JavaScript. On the other hand, when we use the general Google Docs tools, the variety of functions or available components is limited. The strength of this way is in the simplicity for making the document. We can describe the document in the easy GUI and do not need to write complicated program code. For example, when we want to make a combination between multiple pictures on the document, it is quite difficult to realize it by coding. In addition, Google Docs provides simple ways to express figures, graphs, etc. These functions make it easy to organize the document and we can prepare the page quickly. One more important point is in the availability of the responsive web design. In the original website [3] for this conversion, it equips the ability to prepare appropriate view for each screen size/browser/device. If the HTML is written by the programmer, we can implement this responsive web design, because we can manage that by using CSS/JavaScript. If the HTML is prepared by Docs365 add-on, it always gives the same view of the page regardless of the screen size, and it is impossible to implement responsive web design. Actually, it is a very important feature for better user experience [5] and one of the biggest design problem for the generated websites. Therefore, the features in each way at the point of design tool are the following: the web page generated by using Itty.bitty.site has a rich design, and the web page generated by using Docs365 Itty Bitty is good at the preparation process of the document.

TABLE 2. SUMMARY OF THE RESULTS

Criteria	Itty.bitty.site	Docs365 Itty Bitty
Implementable components		
Textual Items	As a HTML document, classified by header, paragraph, etc.	As a Docs file, not classified, but written freely.
Graphical Items	Images from web or any web server.	Images, simple figures, graphs, etc. designed in Docs.
Audio Items	Reference external audio resources.	Impossible in general ways.
Visual Items	Designs by CSS, such as background, header/footer, etc.	Designs including background or text effects in basic functions.
Dynamic Items	Available as general web pages.	Impossible in general ways.
Implementable components		
Adjust Fonts	Using CSS.	Using tools in Docs.
Adjust Screens	Changeable with the screen size. Responsive web design is available.	No functions to adjust the screen size or components placing.
Adjust Graphical Elements	Using CSS.	Using tools in Docs.
Others		
Instrument s to Design Web Page	Using CSS or JavaScript.	Using tools in Docs.
Methods to deploy Website	Change HTML to URL on itty.bitty.site	Change Docs file to HTML, and HTML to URL on Docs365 add-on.
Size of the Website	Small.	Big.
Speed to Appear	Fast, but latency to show images.	Slow, but images and document appears at the same time.
Available Browsers	All browsers except Microsoft browsers.	Unavailable on Smartphone and Microsoft browsers.
Opinion for Whole Document Design	Rich designs including responsive design can be implemented. It is better as the usability of the website.	No flexible adjustments can be applied and designs are limited. However, it is easy for creating the document.

For the point of the structure in each product, there are big differences at the data size of the HTML file, URL, and the speed to appear on the browsers. As a result, the page generated by using Docs365 Itty Bitty has significantly bigger data size than the one of Itty.bitty.site, and appears slower than another one. However, obviously the point is whether the image file comes from any external resources or the Itty.bitty URL itself. As we wrote in the Results section, we need to read the external image resources to express the images in the

Itty.bitty.site web page, but contrary, it includes the image resource data with HTML when the HTML file is created by Docs365 add-on. That is why the size of the file data and URL with Docs365 is apparently bigger than the ones with Itty.bitty.site. Actually, in the Itty.bitty.site page, images come out within 2 seconds after the other components such as texts, background or header designs appeared. On the other hand, in the Docs365 page, images and the other components including texts etc. pop up at the same time. The length of URL affects the number of available browsers to view this web page. According to RFC7230 [6], it states “Various ad hoc limitations on request-line length are found in practice. It is RECOMMENDED that all HTTP senders and recipients support, at a minimum, request-line length of 8000 octets.” and each browser has own limit of characters. There is no accurate information for the max URL length in most of the browsers. Article [7] reports the practical numbers in major browsers. For example, it states the number formally, in the Internet Explorer, it allows the length of 2,083 characters [8], and that is why both of the products in Itty.bitty.site way and Docs365 way with the URL length of more than 3,000 characters are not displayed on the Internet Explorer. Although we cannot conclude that the number of URL characters is absolute factor of this result, the number of Docs365 page URL, more than 300,000 characters, is too long for general browsers.

V. CONCLUSIONS

As we discovered, both the Itty.bitty.site and Docs365 Itty Bitty ways have own characteristics. The Itty.bitty technology has a valuable possibility as a tool to make micro websites, developers can choose the more appropriate way to make the product. Especially, the way with the Docs365 tool has a superiority on its simplicity and versatility, but there are some elements which cannot be implemented in this way, such as responsive web design or inclusion of external resources. The processing way with Itty.bitty.site doesn't have a particular limitation as a single page website, but developers are needed to write in similar way when designing a general website. In addition, because of the structure of each product, contents in the web page may affect the length of the URL, particularly in the Docs365 way. Developers are needed to consider carefully the characteristics of their product and use the proper tool for the development.

REFERENCES

- [1] itty.bitty. [Online]. Available: <https://itty.bitty.site>, Nov, 2018.
- [2] Jacob Ziv, Fellow, IEEE, and Abraham Lempel, “A Universal Algorithm for Sequential Data Compression”, IEEE Transactions on Information Theory, Vol. IT-23, No. 3, May, 1977.
- [3] Shohei Otsuka, Keisuke Kato, and Vitaly Klyuev, “Academic Laboratory Website Development”, In Proc. the 2015 17th IEEE International Conference on Advanced Communications Technology (ICACT2015), July 1-3, 2015, Phoenix Park, Korea, pp. 768-771, DOI: 10.1109/ICACT.2015.7224903.
- [4] Docs 365 Itty Bitty. [Online]. Available: <https://apps.docs365.io/AboutIttyBittyAddon.aspx>, Nov, 2018.
- [5] Devita Mira Lestari, Dadan Hardianto and Achmad Nizar Hidayanto, “Analysis of User Experience Quality on Responsive Web Design from

- its Informative Perspective” International Journal of Software Engineering and Its Applications Vol. 8, No. 5, 2014, pp. 53-62.
- [6] R. Fielding, Ed., Adobe, J. Reschke, Ed. and greenbytes, “Hypertext Transfer Protocol (HTTP/1.1): Message Syntax and Routing” Internet Engineering Task Force (IETF), June 2014, p. 20.
- [7] WWW FAQs: What is the maximum length of a URL?. [Online]. Available: <https://boutell.com/newfaq/misc/urllength.html>, Nov, 2018.
- [8] Microsoft Support. [Online]. Available: <https://support.microsoft.com/en-gb/help/208427/maximum-url-length-is-2-083-characters-in-internet-explorer>, Nov, 2018.



Yuta Nemoto is a bachelor student in the Department of Computer Science and Engineering at the University of Aizu, Japan. His research area is web application development.



Vitaly Klyuev is a professor at the University of Aizu, Japan. He received a Ph.D. degree in Physics and Mathematics from St. Petersburg State University, Russia in 1983. His research domain includes information retrieval, software engineering and analysis of computer algorithms. He has more than 100 publications in referred journals and conference proceedings, three co-authored and eight co-edited books. Dr. Klyuev is a member of editorial board of several academic journals and a program committee member of more than 20 conferences sponsored by ACM, FTRA, IEEE, ISCA and IARIA.