

SOFTWARE METAPAPER

Feature Usage Explorer: Usage Monitoring and Visualization Tool in HTML5 Based Applications

Sarunas Marcuska,¹ Pekka Abrahamsson ¹

¹ Free University of Bolzano, Bolzano, Italy

Feature Usage Explorer is a JavaScript library, which automatically detects features in HTML5 based applications and monitors their usage. The collected information can be visualized in a Feature Usage Diagram, which is automatically generated from an input json file.

Currently, the users of Feature Usage Explorer have to design their own tool in order to generate the json file from collected usage information. This option remains viable when using the library in order not to constraint the user's choice of preferred data storage.

Feature Usage Explorer can be reused in any HTML5 based applications where an understanding of how users interact with the system is required (i.e. user experience and usability studies, human computer interaction field, or requirement prioritization area).

Keywords: feature reduction, feature usage diagram, software bloat, user experience studies, usability studies, human computer interaction, html5

Funding statement

Sarunas Marcuska is a PhD student of the Free University of Bolzano. He has received a scholarship from the province of Bolzano for his studies. The tool is a part of his PhD work.

(1) Overview

Introduction

Feature Usage Explorer was developed to support feature reduction initiative [1], which aims to automatically identify features with a low value (i.e. very rarely used) and systematically remove them from the system. The reduction initiative is of commercial interest as between 30 to 50 per cent of most software products consist of unnecessary features [2], known as Software Bloat [3]. This causes computer applications to become slower and require higher hardware capacities, both of which increase the costs of system maintenance.

To address the problem of Software Bloat Feature Usage Explorer, automatically detects features in web applications, monitors their usage, and creates a figure which enables users to visualize the collected information. In this context a feature is: "a realized functional requirement of a system (i.e. is an observable unit of behaviour of a system triggered by the user)"[4], where the user can be human, other system, web-service, etc.

The efficacy of the Feature Usage Explorer was assessed for recall and precision measurements [5]. The results from this case study demonstrated the ability of the tool to determine complete feature sets with high accuracy.

Note that at present this tool works only on HTML5 technology and features using flash technology are not presented in the feature set.

Objective methods, which could help to estimate the value of a feature, were explored in a separate case study [6]. Estimating the value of a feature currently involves detailed questionnaires that are not only costly but are also limited in their value due to non-compliance of key stakeholders.

Finally, the introduction of a modelling notation, Feature Usage Diagram, which enables users to visualize feature usage information, was evaluated in [7]. Validation of the elements of the Feature Usage Diagram was determined in a separate case study. The results showed that users find the notation easy to learn and understand, and that the tool support would make it practical to use. Therefore, a Feature Usage Diagram was added to Feature Usage Explorer. The decision makers could use this notation during the feature reduction process.

There are several existing techniques that allow users to collect usage related information in a software system [8, 9, 10]. The advantage of the Feature Usage Explorer is that it helps to analyse the system strictly on a feature level, while the existing approaches were not created for

A more detailed description of the aforementioned fields is provided below:

- **id**. This field represents a unique identifier of a feature, or a feature group. It can have any integer value.
- **name**. This field represents the name of a feature, or a feature group. It can have any string value.
- **usage**. This field shows how many times a feature was used. It can have any string value.
- **type**. This field can have only two values “feature”, or “feature group”. If a type is “feature group” then “siblings”:[] attribute must be present, which represents features that are in the feature group. The siblings’ field must have the same attributes as feature: id, name, usage, type, linksOut, linksIn.
- **linksOut**. This field represents outgoing links from a feature, or a feature group. It can have a multiple links outgoing links.
- **linksIn**. This field represents incoming links to a feature, or a feature group. It can have a multiple incoming links.

The example of the input json file is available in the code repository: <https://raw.githubusercontent.com/smarciuska/feature-usage-explorer/master/json-examples/json.txt>

A demo json file generator, written in Java programming language, is available in the code repository: <https://github.com/smarciuska/feature-usage-explorer/tree/master/simple-json-generator>

More details on how to configure Feature Usage Explorer library are provided in the user manual: <https://github.com/smarciuska/feature-usage-explorer/blob/master/UserManual.pdf>

The example with Feature Usage Explorer functionalities is provided in demo.html file: <https://github.com/smarciuska/feature-usage-explorer/blob/master/demo.html>

Quality Control

User-acceptance tests were performed during the case studies. The GUI has been tested on Mozilla Firefox, Google Chrome and Apple Safari. We provided several different input json files to test the automatic Feature Usage Diagram generation.

(2) Availability

Operating system

Any operating system, which has one of the following web-browsers: Firefox 2+; Safari 3+; Chrome 1+.

Programming language

JavaScript, HTML, CSS.

Additional system requirements

Feature Usage Explorer can be integrated only on web based applications.

Dependencies

jquery 1.6: <https://ajax.googleapis.com/ajax/libs/jquery/1.6/jquery.js>
 fabric 0.9.35: <https://github.com/kangax/fabric.js/tree/v0.9.35>

List of contributors

- Sarunas Marcuska (marciuska.sarunas@gmail.com)
- Pekka Abrahamsson (pekka.abrahamsson@unibz.it)

Archive

Name

GitHub

Persistent identifier

<https://github.com/smarciuska/feature-usage-explorer/blob/master/FeatureUsageExplorer.zip>

License

GNU General Public License version 3

Publisher

Sarunas Marcuska

Date published

02/05/13 (version 1.0)

Code Repository

Name

GitHub

Identifier

<https://github.com/smarciuska/feature-usage-explorer>

License

GNU General Public License version 3

Date published

02/05/13 (version 1.0)

Emulation environment

Name

Apache httpd 2.4.X

Identifier

<http://httpd.apache.org/>

License

<http://www.apache.org/licenses/>

Date published

25/02/13

Language

English

(3) Reuse potential

Feature Usage Explorer has a great potential to be reused in any situation where companies, or researchers need to understand how users interact with the system: user experience, usability studies, human computer interaction, feature reduction, and requirement engineering fields.

For example, in feature reduction field, Feature Usage Explorer can help to identify a complete feature set. The tool can then monitor the usage of features and send the collected information to a selected data storage. Finally, the usage data can be visualized in Feature Usage Diagram. The Feature Usage Diagram might reveal rarely used features, which are the primary target for the feature reduction. Moreover, the library can be easily extended to

enrich the existing functionality in order to satisfy more specific needs. For example, a company might be interested to understand not only which features are used, but also who the users are. Therefore, the code could be extended to send the ip address or user name to the server if it is stored in a session/cookie variable.

Moreover, the software could be particularly useful in requirement engineering field. Currently, the requirements are prioritized and their value is estimated through questionnaires and surveys of the key stakeholders. The usage information collected by the tool can be used as an automatic feature value indicator. For example, the most frequently used features might indicate high value, while the rare used features might mean that they are not that valuable for the users [6]. Of course, company should take into consideration that some features are intended to be used rarely (i.e. settings).

Finally, the Feature Usage Explorer can be used in user experience studies in order to improve the quality of a system. The tool already provides sufficient information for the analysis of feature patterns, including feature usage data and feature usage visualization. The decision makers can use this information to redesign the system in such a way that it is more attractive for the users. For example, the usage patterns might show that users use login functionality many times in comparison to the other features. In this case company might redesign the system in such a way that it remembers user credentials and uses them to automatically login to the system.

Most of the aforementioned extensions can be easily implemented by sending extra information to a selected server using ajax calls:

```
$.ajax({
  dataType: "json",
  url: url-to-server + "?parameter-name=" +
    EXTRA-INFORMATION,
  data: data,
  success: success
});
```

To modify or to extend the Feature Usage Diagram would require modification of the code of the library. The complexity of the task is directly related to the modification which is required but it likely to prove challenging. For example, it is relatively easy to add additional attributes to a feature or a feature group, but it is more complicated to extend the code that user can manually select the list of features and then group them into a feature group.

Acknowledgements

We would like to thank all participants who provided they time and effort during the case studies, which were driving the feature reduction research. Moreover, we would like to thank all companies that gave us valuable feedback on how to further improve Feature Usage Explorer.

References

1. **Feature reduction initiative**. Available at: <http://www.featurereduction.org>. [Last accessed 30 April 2013].
2. **Ebert, C and Dumke, R** 2007 Software measurement. Heidelberg: Springer. p. 561.
3. **Xu, G, Mitchell, N, Arnold, M, Rountev, A and Sevitsky, G** 2010 Software bloat analysis: finding, removing, and preventing performance problems in modern large-scale object-oriented applications. In: Proceedings of the FSE/SDP workshop on Future of software engineering research. ACM. pp. 421-426. DOI: <http://dx.doi.org/10.1145/1882362.1882448>
4. **Eisenbarth, T, Koschke, R, and Simon, D** 2003 "Locating features in source code". *Software Engineering, IEEE Transactions on* 29(3): 210-224. DOI: <http://dx.doi.org/10.1109/TSE.2003.1183929>
5. **Marciuska, S, Gencel, C, and Abrahamsson, P** 2013 Automatic Feature Identification in Web Applications. Available at: http://www.featurereduction.org/case-study/feature_identification.pdf
6. **Marciuska, S, Gencel, C, and Abrahamsson, P** 2013 Exploring How Feature Usage Relates to Customer Perceived Value: A Case Study in a Startup Company. In: Proceedings of the 4th international conference on software business. Springer.
7. **Marciuska, S, Gencel, C, Xiaofeng, W, and Abrahamsson, P** 2013 Feature Usage Diagram for Feature Reduction. In: Proceedings of the 14th international conference on agile software development. Springer.
8. **Atterer, R, Wnuk, M, and Schmidt, A** 2006 Knowing the user's every move: user activity tracking for website usability evaluation and implicit interaction. In: Proceedings of the 15th international conference on World Wide Web. ACM. pp. 203-212.
9. Microsoft Spy++. Available at: [http://msdn.microsoft.com/en-us/library/aa264396\(v=vs.60\).aspx](http://msdn.microsoft.com/en-us/library/aa264396(v=vs.60).aspx). [Last accessed 30 April 2013].
10. Google Analytics. Available at: <http://www.google.com/analytics>. [Last accessed 30 April 2013].

How to cite this article: Marciuska, S and Abrahamsson, P 2013 Feature Usage Explorer: Usage Monitoring and Visualization Tool in HTML5 Based Applications. *Journal of Open Research Software* 1:e7, DOI: <http://dx.doi.org/10.5334/jors.af>

Published: 18 October 2013

Copyright: © 2013 The Author(s). This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 Unported License (CC-BY 3.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited. See <http://creativecommons.org/licenses/by/3.0/>.

 The *Journal of Open Research Software* is a peer-reviewed open access journal published by Ubiq Press

OPEN ACCESS 