

Automated Heuristic Evaluation for Usability of Multiple Web Interface

Mihirtab Zewdu

Kolfe Keranyo Sub City Land Information Centre,
City Government of Addis Ababa, Ethiopia
mihirtz@gmail.com

Dagmawi Lemma

Department of Computer Science, Addis Ababa
University, Ethiopia
dagmawil@yahoo.com

Abstract

Heuristic evaluation is the most common usability evaluation method of web interface, that involves usability specialists' judgment on agreed upon usability principles (heuristics). The current specification method is limited on independent expert judgments. Though, there are works to evaluate web interfaces from the usability perspective, these works underlying developer heuristics and lacks consistency in using the agreed represented standard heuristics.

This research addressed the above problems by representing agreed heuristic specification of principles via ergonomics guideline and automating evaluation that would made possible by AHEMW algorithm. As a result, this automation is one step towards a procedural syntax based heuristic evaluation of multiple web interfaces that brings consistent evaluation by maintaining users' preferences.

Keywords: heuristic Evaluation; Usability Evaluation; Web Interface

1. Introduction

The unaddressed issues/problems that need to be solved over this study are; How to represent the heuristics for automation to usability evaluation of web pages? Could it be possible to find evaluation techniques that consistently perform automated evaluation across multiple web pages? Usability in the ISO standard for software qualities (ISO 1991b) is defined as “a set of attributes of software which bear on the effort needed for use and on the individual assessment of such use” [2]. Also Eason defines usability as “the degree to which users are able to use the system with the skills, knowledge, stereotypes and experience they can bring to bear”.

Heuristic evaluation is defined as a usability evaluation technique in which the most appropriate solution is chosen using rules and principles [3, 4].

Recently used evaluation techniques are used to obtain information from limited usability attributes, which plan accessibility guidelines to allow interface evaluation only through pre-defined and limited guideline evaluation definition method. The interface evaluation syntax reduces dependency on implementation specifics and makes more reusable [5]. The ultimate goal is to design and implement automated heuristic evaluation techniques for

consistent evaluation of publicly accessible hypertext structures.

The study is worthy because of user preference based web software evaluation method developed, simple and natural expression of web users creating satisfaction and creating consistency in evaluation of multiple pages. However, the study is limited on context awareness and prediction and types of guidelines (i.e., guidelines have to be representable as per this work).

2. Related Work

2.1 Automation of Heuristic Evaluation

The literature has showed different usability evaluation methods such as analytical modeling and information simulation model. Most of these usability evaluation tools were recording action methods a user makes while exercising in a given web interface. This produces voluminous log report that makes it difficult to map the recorded log to usability actions, which has a need for further analysis on the huge amount of data that reduces evaluators' efficiency.

Literature [7, 9] shows that most quantitative methods of evaluating web sites focus on statistical analysis of usage patterns. The analysis is mainly based on server logs, traffic-based analysis, and time-

based analysis that provide data the evaluator needs to interpret. However, it has been delimited on the lack of agreement over guidelines and suggests as there is no one path to good design. The studies open a way toward the expression of guideline definition in platform independent definition by creating empirically justified and reproducible interface design recommendations.

A study in [7] tried to demonstrate knowledge-based web automated evaluation tool with reconfigurable guidelines optimization (KWASMI). It covers guidelines based evaluation for a single web page with Guideline Definition Language (GDL) representation [13]. The guideline definition is done using GDL. It lacks to have a heuristic basis for the criteria's and platform independence in the definition.

2.2 Design of Automated Evaluation

It is important to have the ability to facilitate the evaluation of web pages by creating well-developed (platform independent) guideline among institutions that wish to implement them. Guideline based evaluation of a content specific web based evaluation from the contents of HTML source sent to the web browsers has decisive impact for usability judgment.

2.3 Guideline Representation

Design advisor [10, 11] explained by Faraday contains an ergonomic algorithm that predicts the visual path between elements on the web page depending on their type, size, and color. It has been used to assist the visual analysis of web pages. Following eye tracking analysis results of design advisor provides log report results of evaluated web pages. Which evaluators biased towards aesthetically pleasing interfaces and structurally feasible designs regardless of efficiency and effectiveness.

Web Static Analyzing Tool (Web SAT) processes HTML tag grouping in an individual page and does not suggest possible mitigation and improvement strategies [12]. This tool does not support other usability guidelines such as IEEE Standard except the HTML composition of web pages.

2.4 Multiple Web page Evaluation

The problem with multiple pages is unaddressed to automated evaluation of metadata description, due to the nature of web navigation and crawling web page is nondeterministic and technologically it was unreachable.

Even if single page has made great improvements to the multiple views of user preferences it has limitation of handling logged and graphed information [8].

3. The Proposed Solution

Lack of usability experts and increased demand of experts as a possible solution consists of retaining the knowledge and experience of these experts and expressing it in syntax form to be reviewed and applied. So, the manual heuristic makes an overhead to the evaluation because of the need for organization, limited domain expertise, and need to have automation.

An evaluation is needed to perform evaluation of the product selection process and produce a usable user interface at stage n+1 by making conformity of ergonomic and functional requirements to the adherence of heuristics.

In the manual heuristic evaluation, results of the assessment and experts' judgment is recorded as written reports from each independent evaluator or by having the evaluators verbalized comments from an observation as they go through the interfaces. The currently often used approach to heuristic interface evaluation is manual evaluation that provides independent interface usability judgment, which has bias more to the aesthetics of the web interfaces. In general, the representation helps to obtain direct access to relevant evaluation points in the usability evaluation, as a result proceed through the different steps of the evaluation.

3.1 Guideline Representation

A knowledge representation is needed for guidelines of a web that could be used as if they explicitly state the attributes of the usability guidelines of web content. The representation can be new or modified guideline criteria enabling evaluators to express the usability body of

knowledge done in a separate definition to come across the heuristic definition.

The evaluator need to configure according to the web problem of the system evaluation presented. The overall guideline structure grouped into five problem domain design evaluation decision structure are, high level design, conceptual content, content objects, navigation & search, and content presentation process.

Identifying the guidance on structuring WWW user interface problem attributes: The syntax followed to represent heuristics is:

Table 1: Syntax definition for the AHEMW representation

Syntax element	Use interpretation	Syntactic nature that entails to the evaluators`
[HTML Tag/ Evaluated content substring] alternative name: Evaluated value	Specific notation to which the evaluation acts. It perform layout or functional evaluation	The subject of the evaluation,
[Existence or Pattern]	The item the functions i.e. possible values: either existence or pattern,	Act as a verb of required evaluated value,
[Needed or Not needed or optional] for Existence or [pattern {tag_substring}] for pattern	Is a value where the specified web rendering content is either ‘needed’ or ‘not needed’ is specified. ‘sub-sting’ for the pattern needed.	Act as a value object, configured to the evaluator context.
[Success and failure messages to the problem]	Show message for success and problem message for corrective notification.	
[Scope of criteria in the evaluation]	Scoped the evaluation guideline is either for only home page or for any of the child pages.	

The categorization into five problem domain of user interface design in Table 1 of heuristics helps to follow through

G: Place title information consistently, where G is set of criterion of ‘C1 ‘and ‘C2’

By applying the grammatical syntax the principle criterions defined to the evaluation pattern.

C1: Title substring (Tag) should exist and C2: Title should be presented consistently

Definition using syntax for C1: [title], [existence], [needed], [Title Exists], [Title Doesn’t exist], [any]

[HTML Tag/Evaluated content substring][Existence/Pattern] [Needed/Not needed/Optional] [Messages to the process] [Scope of criteria]

The syntax defines standards to establish to a grammatical practices, identify changes to improve web services incorporating opinions and needs of users [6]. Once defined in representation, heuristic knowledge can be evaluated by the distinguishable parts of web content.

Definition using syntax for C2: [title], [pattern {title_substring*}], [needed], [Title of a page is consistent], [Title of a page is inconsistent], [any]

As defined in the above the represented guideline first, title is checked if it exists in the title tag, then the ‘title_substring’ pattern of checked in the <title> tag, whether the string pattern is followed or not. The criteria used is set for a heuristic evaluator guideline specification.

* The title substring to be evaluated is stated. For example: Yahoo has a pattern Yahoo mail, Yahoo news.. following “Yahoo- service name” pattern.

3.2 Automation of Heuristic Evaluation

When a heuristic evaluation of the web was conducted based on usability standard, the objective was to assess the web pages compliance with usability standards and design guidelines. The usability issues were classified according to the usability problem domain design evaluation decision structure.

Design of Automated Evaluation

Several groups are attempting to create computer interpretable guideline representations that can provide user specific decisions [15]. It is important to have the ability to facilitate the evaluation of web pages by creating well-developed (platform independent) guideline among institutions that wish to implement them. Guideline based evaluation of a content specific web based evaluation from the contents HTML source sent to the web browsers has decisive impact for usability judgment.

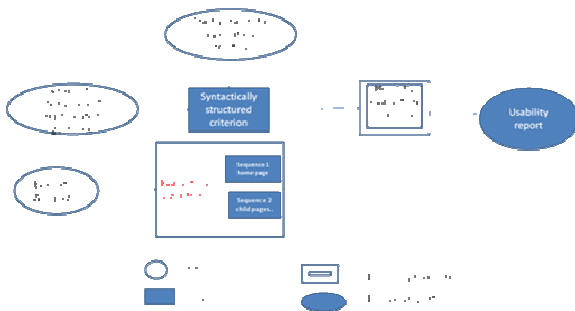


Figure 1: Generic MWIE model of heuristic evaluation.

As illustrated in Figure 1, the MWIE model usability guideline representation is specified in

language independent specification, where the evaluators come across the different views.

The knowledge of notation is promoted to user interface evaluation patterns by providing a specified evaluation interaction design notation and documenting recurring solutions [14]. After that, evaluation report is developed that expose multiple pages usability violation consistently.

3.3 Algorithm of Automated Evaluation

The representation in a separate evaluation configuration focuses to work and execute the heuristic definition that makes separation of guideline easily modifiable and web programming language independent web interface content

Web pages consist of a set of display tags, links and additional data associated child links. All web servers mainly dispatch HTML tags to browsers. Home page is the main page through which users typically enter a website and whose URL is typically published or linked as the main web address. The Uniform Resource Locator (URL) substring is retrieved from the home page and child pages content. The URL is queued for web pages content that uses URL of home page stored in temporary variable stack for recursive analysis.

Algorithms pick the right criteria for the respective situation, in a sequential approach for usability problems [16].

<-- = takes method URL= address in <a href ""> tag

Test result = result of syntactic string comparison DOM to HTML

Start

```
HttpRequest(URL of homepage) // provided by the evaluator in the AHEMW graphic interface
```

```
http request (Test stub [i++]) <-- Array[page[i]]
```

```
Evaluate Homepage <-- TestStub(page[1]) //Homepage
```

```
HttpRequest Success (Home page) {
```

```
  [HTML Tag/ Evaluated content substring][Existence/Pattern] [Needed/Not needed/Optional] [Messages to the process] [Scope of criteria]
```

```
  Test result[11] <-- string compare(Hompage)
```

```
  [HTML Tag/ Evaluated content substring][Existence/Pattern] [Needed/Not needed/Optional] [Messages to the process] [Scope of criteria]
```

```
  Test result[12] <-- string compare(Hompage)
```

```

Usability evaluation Result <--Test result(page [i])
Array(i =0 i< numberTestStub i++)
{
  http request (Test stub [i++]) <-- Array[page[i]]
  Evaluate page [i]<-- TestStub (page[i])
  For (j=0 guideline[j] <-- guideline +1) Httprequest Failure (URL) {
    Do {
      http request (Test stub [i, URL) attempt <-- maxAttempts
      Attempts Success : JUMP X
      Failure : Report as “broken link”
    }while (Attempt<3)
    TestStub[i+1] <-- For each (<a href = “descendent child URL address” >)
      : JUMP to Point X
  }
}
Usability evaluation Result <--Test result(page [i])
End

```

Algorithm 1: AHEMW Algorithms

The two main activities done in the algorithm are: firstly, the evaluating algorithm queue home page by defining set of TestStub make set of evaluation with a needed syntax parameters.

Start

Httprequest(URL of homepage) // provided by the evaluator in the AHEMW graphic interface

Evaluate Homepage <-- TestStub[1]
(Homepage)

The usability algorithms are deterministic as they build the contents inspection of a web interface by applying usability rules defined in the representation. The action is made by syntactic string comparison to guideline represented and retrieved home page DOM of HTML to a guideline defined pattern and existence, which is stripped from the structural HTML markup. The algorithm generates report for each test case as long as there are evaluations covered as to the required loop.

4. Prototype

The AHEMW algorithm is simulated to show how the representation and heuristics contribute for the automation of usability evaluation. In addition, the representation concepts have been enabled to

show in a natural human understandable, pre-defined and web content based usability corrective recommendation to be made. The prototype is developed in JavaScript JavaScript with ExtJS library was used for reading (retrieval) file, and retrieving DOM structure of web.

The AHEMW is divided into three sub systems (Figure 2): graphical interface, guideline representation, and evaluation decision subsystem, because it helps to embrace new approaches and perspectives in order to cope with the complexity of modern evaluation.

Graphical interface subsystem: The evaluation needs URL information for evaluation service request. The evaluation graphical interface is provided by URL.

Guideline representation subsystem: Make ISO 9241-151 heuristics of human-system interaction part 151 guidance on WWW standard into easy specification in a separate text file so that any evaluator can be behind an interface without impacting the rest of the subsystem.

Evaluation decision subsystem: Provides analysis to the landing and any of child pages. The attributes result is presented to the corresponding identified problem.

Automated Heuristic Evaluation of Multiple Web Interface

Enter Web Address:

Consistency	Broken Links	Recommendations
Analysis Result for http://localhost/abc/ Title Exists Head is present Body Exists Landing page present Analysis Result for http://localhost/abc/a.html Title Doesn't exist Head is present Body Exists Analysis Result for http://localhost/abc/c.html Title Exists Head is present Body Exists	http://localhost/abc/ is Active link. (In http://localhost/abc/) http://localhost/abc/a.html is Active link. (In http://localhost/abc/) http://localhost/abc/c.html is Active link. (In http://localhost/abc/) http://localhost/abc/dfa.html is Broken link. (In http://localhost/abc/a.html) http://localhost/abc/ababa.html is Broken link. (In http://localhost/abc/c.html)	Guideline: Title: Basic Page Items Recommendation: Every page ... (for http://localhost/abc/a.html) Page should have title (for http://localhost/abc/a.html)
Analysis 2 broken links		
Evaluation Success Result: 80%. Cases: 15 & Errors: 3		

Figure 2: Prototype for running usability evaluation

5. Conclusion

The analysis report is made by an empirical validation and taking a case of in HiLCoE School of Computer Science and Technology site. The representation facilitates communication amongst evaluators practicing across the globe. Also it helps evaluation usability specialists to understand and share best practices.

The automation with an algorithm enables evaluators better understand reactions to their web evaluation and help them defend against guideline critics facilitating different solution assumptions for the problem of web pages. The AHMW enabled to meet upcoming evaluation challenges by creating platform for evaluation language independence and studying representation that address evaluation of multiple pages consistently.

References

- [1] A. Holzinger, "Usability Engineering Methods for Software Developers", *Communications of the ACM*, Vol. 48, No. 1, pp. 71–74, Jan. 2005.
- [2] G. Gaffney, "What is Usability Testing?," in *Information & Design*, 1999.
- [3] B. A. Myers, R. Wolf, K. Potosnak, F. A. South, and C. Graham, "Heuristics in Real User Interfaces", *INTERCHI'93 Panel*, pp. 1–4, 1993.
- [4] A. Seffah, M. Donyaee, R. B. Kline, and H. K. Padda, "Usability Measurement and Metrics: A Consolidated Model", *Software Quality Journal*, Vol. 14, No. 2, pp. 159–178, Jun. 2006.
- [5] B. Venners, "Design Principles from Design Patterns: A Conversation with Erich Gamma, Part III," *Scribd*, <http://www.scribd.com/doc/79668962/Design-Principles-From-Design-Patterns>.
- [6] M. Y. Ivory, "An Empirical Foundation for Automated Web Interface Evaluation", *University of California at Berkeley*, 2001.
- [7] R. Grandgagna and U. De Louvain, "KWARESMI – Knowledge-based Web Automated Evaluation Tool with Reconfigurable Guidelines Optimization," *Proceeding of 2nd Int. Conf. on Universal Access in Human-Computer Interaction UAHCT*, No. 4, pp. 1504 – 1508, 2003.
- [8] M. Y. Ivory and M. A. Hearst, "The State of the Art in Automated Usability Evaluation of User Interfaces", *ACM Computing Surveys*, Vol. 33, No. 4, pp. 1–26, 2000.
- [9] M. Y. Ivory and U. C. Berkeley, "Empirically Validated Web Page Design Metrics," *CHI 2001*, Vol. 3, No. 1, pp. 53–60, 2001.

- [10] K.-P. Robert W. Proctor, "Web Guidelines vs GUI Guidelines", In *Handbook of Human Factors in Web Design*, pp. 691–692, 2005.
- [11] C. Mariage, "State of the art of Web Usability Guidelines," ... of *Human Factors in Web ...*, pp. 1–25, 2005.
- [12] J. Scholtz, S. Laskowski, and L. Downey, "Developing Usability Tools and Techniques for Designing and Testing Web Sites," in ... on *Human Factors & the Web*, Vol. I, pp. 1–10, 1993.
- [13] F. B. Tan and L. L. Tung, "Exploring Website Evaluation Criteria using the Repertory Grid Technique: A Web Designers' Perspective", in *SIGHCI 2003 Proceedings*, pp. 65–69, 2003.
- [14] D. Nuno and J. Nunes, "Object Modeling for User-Centered Development and User Interface Design: The Wisdom Approach," No. April, 2001.
- [15] P. Chawan and S. Ninoriya, "Towards the Quality Measurement of Web Content Management System," *International Journal of Advanced Engineering & ...*, No. 148, pp. 148–151, 2011.
- [16] E. K. Burke, B. L. Maccarthy, S. Petrovic, and R. Qu, "Knowledge Discovery in a Hyper-Heuristic for Course Timetabling Using Case-Based Reasoning," 2002.