Conceptual design and implementation of an automated metrics and model-based usability evaluation of UI prototypes in Figma.

Lara Bertram

Hochschule Düsseldorf (HSD), lara.bertram@outlook.de

Markus Dahm, Prof. Dr. Ing. M.Sc.

Hochschule Düsseldorf (HSD), markus.dahm@hs-duesseldorf.de

Evaluations should be conducted regularly throughout the development of interactive applications to ensure and improve usability. Automated approaches can enable time- and cost-effective evaluations, as the participation of test and expert personnel is not required. Especially in early phases of development, usability evaluations could thus be made possible without much effort. This paper describes a concept for automated usability evaluation of UI prototypes that supports the design and can complement classical usability evaluation methods. To test this concept, a plugin was implemented that evaluates design drafts in the Figma prototyping tool. The evaluation is based on usability metrics, a GOMS model for modelling interaction sequences that also supports predicting processing time and hint patterns for identifying usability problems in interaction sequences. The tool was tested and evaluated in a number of trials, which provided insights into the suitability of the concept and identified approaches for further development.

CCS CONCEPTS • Human-centered computing ~ Human computer interaction (HCI) ~ HCI design and evaluation methods • Human-centered computing ~ Interaction design • Human-centered computing ~ Human computer interaction (HCI) ~ HCI theory, concepts and models

Additional Keywords and Phrases: Ergonomics, Usability, Evaluation, Usability Evaluation, Metrics, Model, GOMS, KLM, Prototyping, UI-Prototyping, Prototype, UI-Prototype, UI-Design, UX-Design, Figma

1 INTRODUCTION

Throughout the development process, evaluations can find and correct significant usability deficiencies of an interactive application early on [1]. Therefore, it should be possible to integrate evaluations into the development process, especially in early development phases, in a time and cost effective way. It would be ideal if the tools used for designing and testing interactive applications offered the possibility of automated evaluation. Therefore, this paper examines existing approaches of automated usability evaluations and transfers them to the automated examination of UI prototypes. Based on the findings, a prototypical implementation was developed. The usefulness and usability of this implementation is evaluated by conducting and evaluating user tests.

2 RELATED WORK

Approaches to automated usability evaluation have so far mainly focused on the evaluation of existing applications. They can be assigned to three categories [2] as follows. Firstly, interaction-based evaluation methods use recorded user interactions that are either compared with an "optimal" reference (e.g. [3]) or examined for certain patterns, e.g. interaction patterns for usability problems (e.g. [4]). Secondly, metrics-based methods aim at evaluating the user interface on the basis of various usability indicators (e.g. [5]). Thirdly, model-based approaches use analytical modelling or simulation techniques to make predictions about the usability of a user interface (e.g. [6]).

A (partially) automated usability evaluation of UI prototypes is currently provided by tools for performing remote (interaction-based) usability tests (e.g. in [7]). Especially for accessibility testing, some plugins of different prototyping tools can also be used. Such plugins often examine (metrics-based) visual factors such as colour contrast (e.g. in [8]). Other (prototyping) tools for metrics-based evaluation of the usability of UI prototypes also focus on visual, textual and functional, including simple navigational, investigations (e.g. [9]).

The number of tools for model-based evaluations of UI prototypes seems to be still small. The tool StEM aims to make predictions about the performance of a user interface based on the so-called Touch Level Model [10], a model for quantifying touch-based interactions, by placing individual actions on the image of an interface and combining them into a sequence of interactions. [11]

3 A TOOL FOR AUTOMATED USABILITY-EVALUATION

In this paper, we describe an automated usability evaluation of prototypes which is implemented as a tool in the form of a plugin for the web-based prototyping tool Figma [12]. Considering the context of use, the way designers work and different characteristics of UI prototypes, both a metrics-based and a model-based approach seem to be suitable [13]. These methods are combined to support different approaches to design and to explore a wider range of usability aspects.

3.1 Metric-based evaluation

3.1.1 Metrics

For the metrics-based evaluation, a metrics collection is used and slightly extended. The collection is largely taken from [9] and is based on the Quality in Use Integrated Map (QUIM) [14], the Research-Based Web Design & Usability Guidelines [15] and the Web Content Accessibility Guidelines (WCAG) [16]. For the evaluation, 21 metrics [13] are chosen, which relate to different aspects of the usability of user interfaces.

3.1.2 Evaluation

In order to start an evaluation, the user selects relevant metrics and Figma frames (for details on the user interface see [13]). Each frame represents a page of the design to be examined. Since designers develop their own design procedures over time or follow certain steps, support for repeated work steps can increase the efficiency of an automated tool. For this purpose, evaluation profiles can be defined. An evaluation profile groups several metrics, e.g. concerning the same usability criteria according to QUIM [14]. Evaluation profiles can be defined by the user him/herself.

3.1.3 Presentation of the results

After the evaluation, the results are visualised by listing the results of the selected metrics (see [13] for more). A coloured icon indicates whether violations of a metric were found. If a metric has a threshold value (tolerance), this is used to classify the violations (yellow or red). If violations were found, the problematic frames, elements or attributes are listed below the name of the metric. Clicking on these list entries highlights the affected frames or elements in the draft.

Below the current results list, a history of past evaluations consisting of date, time and the results are displayed in list format. Thus, current and former results can be quickly compared to check for improvements.

3.2 Model-based evaluation

Within the course of the model-based evaluation, the tool allows to define interaction sequences representing user tasks. A sequenceconsists of several successive individual actions and can be evaluated based on a GOMS-Model. By comparing two interactions sequences several variants of a design can be compared directly with the support of the tool.

3.2.1 Basis for evaluation

Various models have been established for describing individual processes in the context of human-computer interaction. The GOMS model uses different operators and empirically determined time values to predict the duration of an interaction and is often used to compare several design alternatives [17]. In addition to support individual evaluations of a single interaction sequence, different hint patterns are used for an extended investigation. These hint patterns are based on patterns of interaction-based approaches to usability evaluation (see section 2) and can indicate usability problems. We use a special GOMS Model, the Keystroke Level Model (KLM).

3.2.2 Definition of the interaction steps to be evaluated

In order to define interaction sequences in the context of a prototype evaluation, the user of the plug-in creates tasks (for details on the user interface see [13]). Each task consists of one or more individual actions such as clicking a button or a link or entering content in input fields. In all representations a task is identified by its name and a colour. To define an action, an interaction type is linked to a concrete interaction element of the design. This relationship is visualised by an annotation that indicates the task by its colour and the positioning of the action within the task by a number. When adding individual actions, usability checks are performed concerning the size of the interaction element. If the task already contains one or more actions, it is also checked whether the successive actions of the task are compatible with each other. They are compatible if a previous action can reach a subsequent action. Several tasks can in turn be combined within a scenario. In this case, successive tasks are also checked for compatibility. (for more on compatibility checking, see [13])

3.2.3 Evaluation

Previously defined interaction sequences can be evaluated on a task or scenario basis by choosing one to two tasks or scenarios. In a single task or scenario evaluation, the tasks are evaluated using the KLM and some hint patterns. Since within a scenario there may be hint patterns in the transition areas between the tasks, these areas are considered separately in the scenario evaluation. In an evaluation of two tasks or scenarios, two evaluations are done followed by a comparison of the results.

3.2.4 Presentation of the results

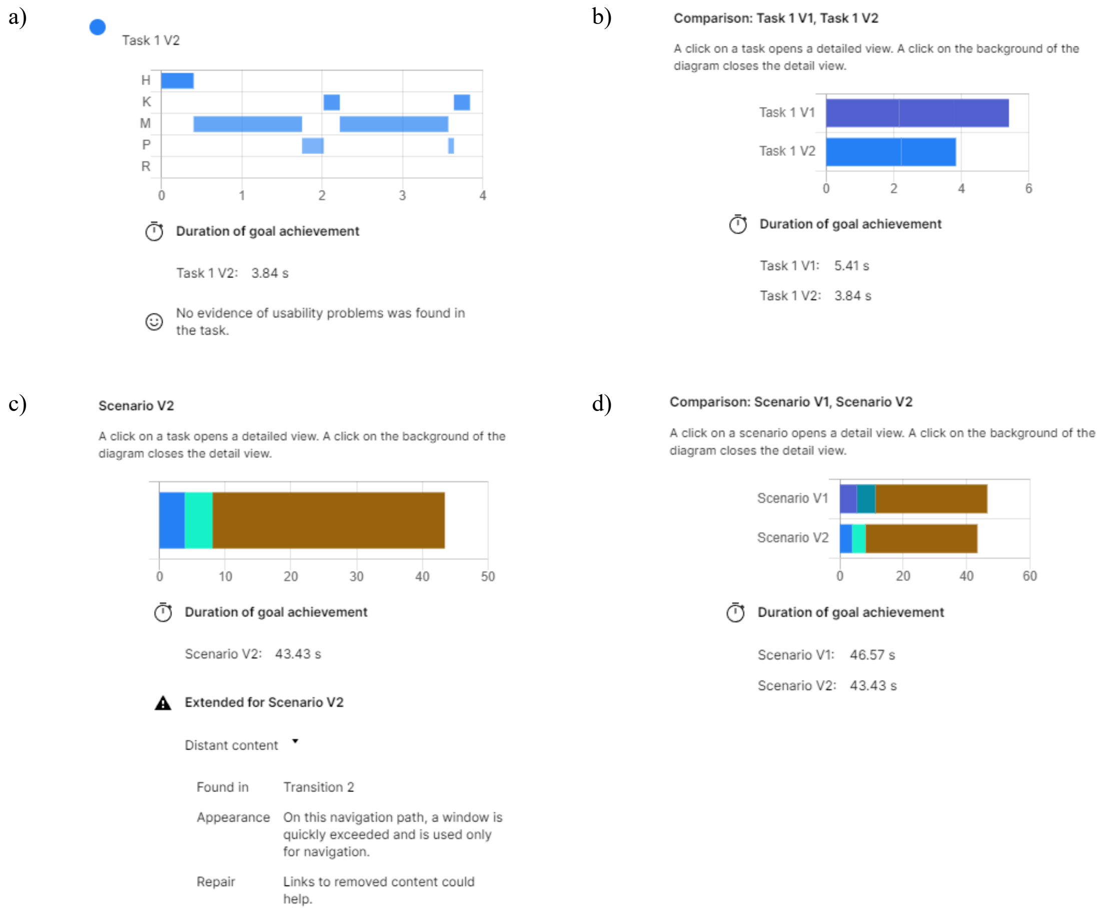
After defining and evaluating interaction sequences on a task or scenario basis, the tool visualizes the results. The visualization uses bar charts to give the designer a direct overview of the duration of a task or scenario. Thus, in case of a comparison, the designer can directly see with which design a certain goal can be reached faster.

The result of a task evaluation (see figure 1a) contains a bar chart showing the operators of the KLM distributed over the interaction sequence. The frequency of the respective operators, their duration and the total duration of the task processing can be seen in the diagram. Below the bar chart, the time it takes to complete the task according to the model is shown, as well as some information about the hint patterns found.

A scenario evaluation (see figure 1c) leads to a clickable bar chart that separates the individual tasks within the scenario by means of colour distinctions. Similar to the evaluation of a task, the total processing time of the scenario and some information about the hint patterns found are displayed below the bar chart. By clicking on the bar area of a task, the task evaluation (described above) is displayed below the scenario bar chart.

The comparative evaluation of tasks (see figure 1b) or scenarios (see figure 1d) leads to a similar display of results. Within a bar chart, the two tasks/scenarios are compared in time. Below this are the amounts of the processing times. Clicking on a task/scenario opens the respective view of the evaluation results (for this task/scenario).

For each task and each scenario (identified by name), past evaluations are saved. If past evaluations are available, a list of past values with date and time can be opened below the "Duration of goal achievement". Thus, past and current evaluations can easily be compared to review the progress being made.



a) Task 1 V2

Duration of goal achievement

Task 1 V2: 3.84 s

No evidence of usability problems was found in the task.

b) Comparison: Task 1 V1, Task 1 V2

A click on a task opens a detailed view. A click on the background of the diagram closes the detail view.

Duration of goal achievement

Task 1 V1: 5.41 s

Task 1 V2: 3.84 s

c) Scenario V2

A click on a task opens a detailed view. A click on the background of the diagram closes the detail view.

Duration of goal achievement

Scenario V2: 43.43 s

Extended for Scenario V2

Distant content ▾

Found in	Transition 2
Appearance	On this navigation path, a window is quickly exceeded and is used only for navigation.
Repair	Links to removed content could help.

d) Comparison: Scenario V1, Scenario V2

A click on a scenario opens a detail view. A click on the background of the diagram closes the detail view.

Duration of goal achievement

Scenario V1: 46.57 s

Scenario V2: 43.43 s

Figure 1: Presentation of the results of the model-based evaluation

4 TESTS AND RESULTS

Tests were conducted to investigate the usability, supportiveness and actual efficiency increase of the plugin. Six people took part in these tests and completed two tasks, each of which addressed one of the approaches to usability evaluation. They then completed a questionnaire that was used to investigate the acceptance and usefulness of the plug-in.

As a result, the metrics-based evaluation seemed to be more understandable due to its lower abstraction and the experience of the test persons with such evaluation methods. The evaluation results could support the improvement of the design draft. However, the concept of evaluation profiles was not understood by all test persons. Some of the participants mistakenly assumed that the creation of profiles was compulsory and would be used for evaluation in the next step.

The use of a model-based evaluation led to a greater number of challenges. Despite initial problems in understanding the definition of tasks and scenarios, all participants were able to solve the tasks given to them. When reading the evaluation results, some participants had problems identifying the clickability of the bar charts, despite a textual hint directly above. Furthermore, the results of an individual evaluation seemed to be rather incomprehensible in most cases, especially with regard to the evaluation of the predicted processing time. The results of the hint patterns found were perceived by the test persons, but did not lead to any noticeable changes or improvements in the interface. Only two test persons suggested improvements.

The evaluation of the questionnaires (see [13]) showed that the majority of the test persons felt supported in the evaluation as well as in the design. The evaluation results were largely seen as helpful. The likelihood of reuse and recommendation also seemed very high from the point of view of the test persons themselves, which suggests high confidence in the test results.

5 CONCLUSION & OUTLOOK

In the context of this study, it has been shown that both a metrics-based and a model-based approach are suitable for the automated usability evaluation of UI prototypes. However, the evaluation of the actual applicability, acceptance and effectiveness of the concept requires further investigation.

The direct integration of evaluation into a popular tool for creating UI prototypes offers several advantages over standalone solutions, especially for increasing efficiency. Above all, the possibility of directly defining and evaluating interaction sequences, as well as comparing multiple sequences within a prototyping tool represents an innovative approach that offers new possibilities for evaluation.

Especially when investigating several design alternatives, a prediction of the interaction duration seemed to offer a quick decision support. However, it seemed to be less useful for the evaluation of individual interaction processes. The use of hint patterns to find usability problems also seemed to provide only limited help, which may be due to the selection of the patterns or the wording of the recommendations for improvement. Since the patterns used originally referred to ready-to-use applications, the identification and inclusion of further patterns of investigable usability aspects of prototypes could lead to an improvement.

The use of functional or navigational usability metrics can allow a broader investigation of different aspects in early phases of development, but it is questionable how much added value these actually have in the prototyping domain. Some aspects depend on the actual implementation and should be investigated there anyway. The implementation framework of the tool included only a small selection of metrics. In the context of the implementation of further metrics, it should be investigated to what extent a larger selection influences the effectiveness and efficiency of the designers and which metrics are actually used on a regular basis.

Further information: https://medien.hs-duesseldorf.de/personen/dahm/Projekte/Seiten/FigmaUsabilityCheck.aspx

REFERENZEN

[1] F. Sarodnik und H. Brau, Methoden der Usability Evaluation: wissenschaftliche Grundlagen und praktische Anwendung, Bern: Hogrefe Verlag, 2016.

[2] M. Bakaev, T. Mamysheva und M. Gaedke, *Current Trends in Automating Usability Evaluation of Websites,* 2016, pp. 510-514.

[3] A. Vargas, H. Weffers und H. Vieira da Rocha, „A Method for Remote and Semi-Automatic Usability Evaluation of Web-based Applications Through Users Behavior Analysis,“ in *MB '10: Proceedings of the 7th International Conference on Methods and Techniques in Behavioral Research*, 2010.

[4] P. Harms und J. Grabowski, *Usage-Based Automatic Detection of Usability Smells,* S. B. Heidelberg, Hrsg., 2014, pp. 217-234.

[5] J. Marenkov, T. Robal und A. Kalja, „Guideliner - A Tool to Improve Web UI Development for Better Usability,“ in *Proceedings of the 8th International Conference on Web Intelligence, Mining and Semantics (WIMS '18)*, New York, 2018.

[6] S. Feuerstack, M. Blumendorf, M. Kern, M. Kruppa, M. Quade, M. Runge und S. Albayrak, *Automated Usability Evaluation during Model-Based Interactive System Development,* S. B. Heidelberg, Hrsg., 2008, pp. 134-141.

[7] UsabilityHub Pty. Ltd., „UsabilityHub,“ [Online]. Available: https://usabilityhub.com/. [Zugriff am 18 Mai 2022].

[8] Stark Lab Inc., „Introducing Stark,“ 2022. [Online]. Available: https://www.getstark.co/. [Zugriff am 9 Februar 2022].

[9] F. Trojahn, *Improving the Usability of UI Prototypes by Incorporating an Automated Evaluation in the Design Process,* 2020.

[10] A. D. Rice und J. W. Lartigue, *Touch-Level Model (TLM): Evolving KLM-GOMS for Touchscreen and Mobile Devices,* 2014, pp. 1-6.

[11] A. Goguey, G. Casiez, A. Cockburn und C. Gutwin, *Storyboard-Based Empirical Modelling of Touch Interface Performance,* 2018, pp. 1-12.

[12] Figma Inc., „Figma,“ [Online]. Available: https://www.figma.com/. [Zugriff am 18 Mai 2022].

[13] L. Bertram, *Automatisierte Analyse und Bewertung der Usability von Graphical User Interfaces,* 2022, HS Düsseldorf, https://medien.hs-duesseldorf.de/personen/dahm/Projekte/Seiten/FigmaUsabilityCheck.aspx.

[14] H. K. Padda, *QUIM map: a repository for usability/quality in use measurement,* 2003.

[15] U.S. Department of Health & Human Services, *Research-Based Web Design & Usability Guidelines,* 2006.

[16] W3C, „Web Content Accessibility Guidelines (WCAG) 2.1,“ 05 Juni 2018. [Online]. Available: https://www.w3.org/TR/WCAG21/. [Zugriff am 27 Januar 2022].

[17] B. E. John und D. E. Kieras, *The GOMS Family of Analysis Techniques: Tools for Design and Evaluation,* 1994.