Received ; | Revised ; | Accepted
DOI: xxx/xxxx

ARTICLE

Evaluation of User Interface Design Metrics using Generator of Realistic-Looking Dashboard Samples

Olena Pastushenko* | **Jiří Hynek** | **Tomáš Hruška**

Brno University of Technology,
Faculty of Information Technology,
IT4Innovations Centre of Excellence,
Božetěchova 1/2,
612 66 Brno,
Czech Republic

Correspondence
*Olena Pastushenko.
Email: ipastushenko@fit.vutbr.cz

Conflicts of interest: none

Summary

The analysis of user interfaces using quantitative metrics is a straightforward way to quickly measure interface usability and other various design aspects (such as the suitability of page layout or selected colors). Development and evaluation of objective metrics corresponding with user perception, however, usually requires a sufficiently large training set of user interface samples. Finding real user interface samples might not be easy. Therefore, we rather use generated samples. In such case, we need to provide a realistic-looking appearance of samples. This paper describes a workflow of the preparation of such samples. It presents a configurable generator based on the composition of simple widgets according to a predefined model. It also describes a reusable library for simple creation of widgets using capabilities of the JavaScript framework Vue.js. Finally, we demonstrate the applicability of the generator on a generation of dashboard samples which are used to evaluate existing metrics of interface aesthetics and show the possibility of their improvement.

KEYWORDS:
aesthetics, dashboard, generator, usability guidelines, user testing

1 | INTRODUCTION

A user interface (UI) is the first (and usually the only) part of the system which is seen by users. Evaluation of the user interface quality is an essential part of the design process (Johnson 2013; Nielsen 1994a; Preece, Rogers, & Sharp 2015). Factors like aesthetics and even the first impression of users play an essential role in the usability and acceptability of the whole system (Lindgaard, Fernandes, Dudek, & Brown 2006; Tractinsky, Katz, & Ikar 2000). A usual way how to evaluate the usability is to let a representative subset of the users to use the interface and analyze their satisfaction and ability to perform selected tasks. The results of a user testing can provide sufficient initial feedback for the UI designers. However, the user testing can take time and effort (communication with users and understanding their needs which are often based on subjective feelings). It can significantly increase the expenses of the system. Thus—in the age of rapid evolution of artificial intelligence—it seems to be reasonable to automatize the evaluation process or at least some of its parts.

The *metric-based* evaluation of UI is one of the approaches which help to automatize the evaluation process (Charfi, Trabelsi, Ezzedine, & Kolski 2014). UI designers use metrics to measure various UI aspects (e. g., layout, or colors). Then, the measured values are used for the detection of usability problems according to the quantitative design guidelines or for the calculation of the overall level of the UI quality (Hollingsed & Novick 2007; Nielsen 1994b). Quantitative measuring of UIs became significant with the evolution of graphical user interfaces. In the 1980s, UI designers used metrics to evaluate textual user interfaces (Smith & Mosier 1986; Tullis 1984). In the 1990s, they applied metrics in tools for the automatic design and evaluation of graphical user interfaces (Ivory & Hearst 2001). Examples of the tools are AIDE (evaluating layout efficiency, alignment and balance) (Sears 1995), SHERLOCK (evaluating interface consistency) (Mahajan & Shneiderman 1997), or SYNOP (an expert system for static ergonomic evaluation of graphic industrial control displays) (Kolski & Millot 1991). In the 2000s, researchers were improving guideline models and designed new metrics (Seffah, Donyaee, Kline, & Padda 2006; Vanderdonckt 1999). They put a higher emphasis on the soft design aspects like the first impression of users and aesthetics of UIs (Ngo, Teo, & Byrne 2003). Readers can also find books dealing with application of design guidelines and metrics—e. g., (Albert & Tullis 2013; Vanderdonckt, Farenc, & on Tools for Working with Guidelines 2000).

Design guidelines should respect the analyzed kind of visualization tool. For example, maps should respect the design principles used in cartography (e. g. Bertin's visual variables (Deeb, Ooms, & De Maeyer 2011)). There are different kinds of UIs (e. g., dashboards) with a different purpose and end users. Every kind of UI should be designed according to the specific design conventions and knowledge of the users (Johnson 2013). Moreover, every user is a unique one with a subjective way of perception and thinking. Hence, the design guidelines might not be applicable in a general way. They should be adjusted according to the design requirements of a chosen kind of UI and requirements of users. New UI metrics and design guidelines should be compared with reviews of real users using a sufficiently large number of UI samples. It can be elaborate and time-consuming to create the training set manually. Hence, the automatic generation of UI samples seems to be a reasonable approach. Several studies generated black & white screens composed of simple shapes (like squares or triangles) to evaluate the interface aesthetics (Altaboli & Lin 2011; Salimun, Purchase, Simmons, & Brewster 2010). There was also an attempt to replace simple shapes with random images to simulate a real UIs (Bauerly & Liu 2008).

The problem of the synthetic samples composed of simple objects is that they usually ignore many characteristics of the real samples which might influence the user perception. It might cause the overfitting of metrics or design guidelines. On the other hand, generation of the realistic-looking samples increases the number of possibilities how the samples can vary from each other. For instance, usability of the metrics measuring screen layout (e. g., Balance or Symmetry) should not be evaluated only with the samples varying in positions and sizes. Ngo et al. (2003) suggested that the metrics should also consider other characteristics of the screen (e. g., the type or color of widgets) which might have an impact on the perceived layout as well. It is not always easy to find the characteristics since the researchers need to count with a limited number of users who can provide a limited feedback. The researchers should have a basic conception of the characteristics which they expect to be influential in quality of a specific kind of user interface. Then, they can define a hypothesis about a UI characteristic, generate appropriate UI samples and perform user testing to prove or disprove the hypothesis. Finally, the user experience could be used for design and improvement of metrics measuring the significant characteristics. The problem is that the initial conception of researchers requires knowledge based on user experience which is, however, the aim of the research. This makes a cyclic dependency shown in Figure 1.

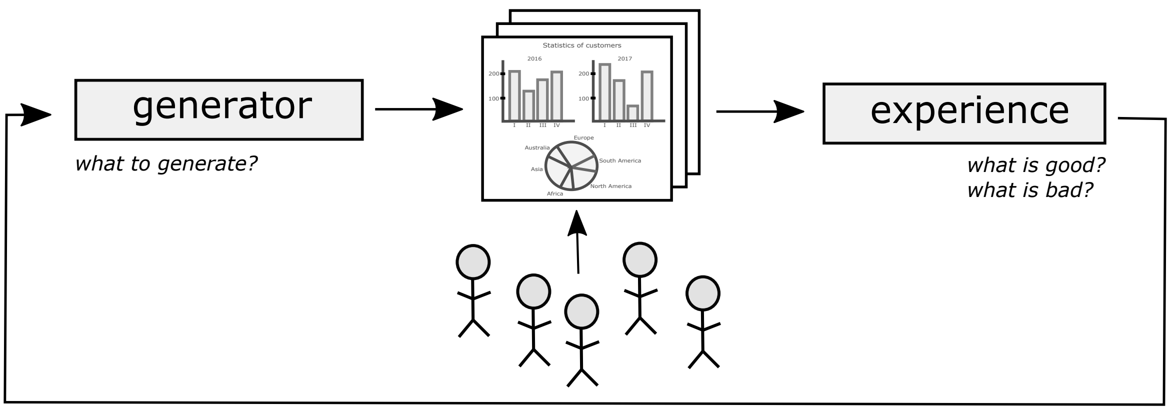


FIGURE 1 The cyclic dependency. We need to have a user experience to generate an appropriate set of interface samples which can be used to gain the user experience.

The goal of this research is to provide a framework which would deal with the problem of cyclic dependency and define a process of design and improvement of metrics and design guidelines. The theoretical part of the paper provides a summary of the dashboard tool in Section 2, followed by the brief overview of the guidelines for dashboard development in Section 3, and existing dashboard generators in Section 4. Section 5 introduces the framework. It defines the process of generation of interface samples and gathering a user experience about the UI samples. Section 6 presents a generator which generates samples based on values of a UI model. Then, Section 7 demonstrates the applicability of the generator on the generation of the dashboard samples which are used for evaluation of the Balance and Symmetry metrics designed by Ngo et al. (2003). We analyze the impact of UI widgets' color, type and displayed dataset on the usability of these metrics. Finally, Section 8, 9, and 10 present, evaluate and summarize the results.

This paper extends the findings published by Pastushenko, Hynek, and Hruška (2018). It provides a better description of dashboard, design guidelines and existing solutions as well as our solution focusing on the generator of UI samples. We also added the second independent study analyzing the perception of layout symmetry.

2 | DASHBOARD

Because of the continuous progress in the areas of Information and Communication Technologies and fast-paced business environments, the amount of generated and consumed data increases significantly. This phenomenon is known as *information overload* (Yigitbasioglu & Velcu 2012). It is the state, in which users are provided with too much information, which leads to negative consequences. It may not influence only the task-related actions but also a health condition of the user. For example, a lower sense of task accomplishment might increase the stress level (Levy 2008). The problem evolves even more if we design the tools for using and representing the data poorly. This can lead to a situation when these tools distract user attention instead of helping the user to make the decision. We can consider dashboards as one of the possible solutions to this problem. The well-designed dashboards should improve decision-making by supporting cognition and perception. Therefore, they should consist of appropriately chosen and arranged visual display media which can present a big amount of data in a meaningful way. It is crucial for the quick perception and processing of the data by the user (Schwendimann et al. 2016).

Few (2006) stated that a dashboard is 'a visual display of the most important information needed to achieve one or more objectives; consolidated and arranged on a single screen so the information can be monitored at a glance.' According to Malik (2005), a dashboard is 'a rich computer interface with charts, reports, visual indicators, and alert mechanisms that are consolidated into a dynamic and relevant information platform.' The main idea of dashboards is not only to display or visualize some specific data but also to be a useful tool in the decision-making process. Dashboards can be used for visual analytics (Zhang et al. 2012). Wexler, Shaffer, and Cotgreave (2017) present examples of very different scenarios of dashboard application. They can help users monitor the status of key performance indicators or notify them when some measures are out of predefined bounds. A *performance dashboard* is 'a multi-layered application built on business intelligence and data integration infrastructure that enables organizations to measure, monitor, and manage business performance more effectively' (Eckerson 2010). A dashboard represents the presentation layer. It is a type of display and form of a presentation designed for communication. Dashboards are composed of different data visualization tools, such as maps, charts, grids, and gauges. Graphical elements of a dashboard, presenting a specific type of information, are called *widgets*.

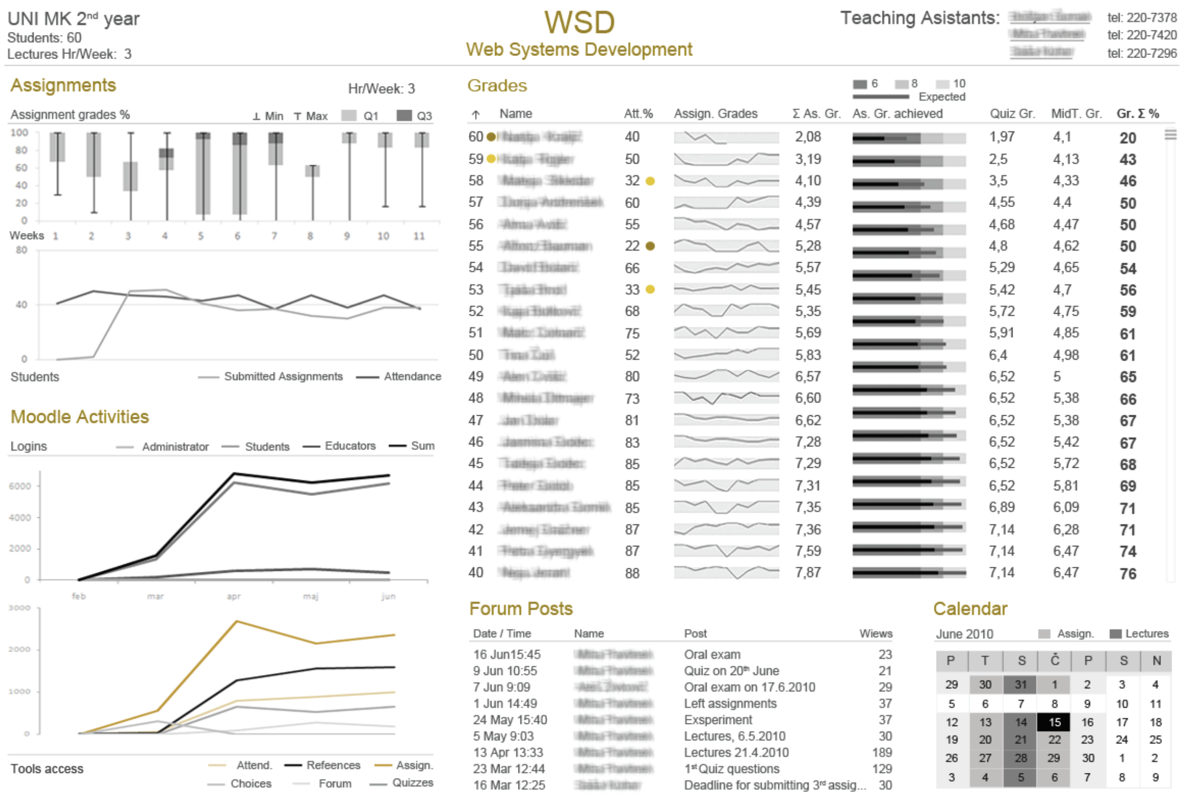


FIGURE 2 An example of the learning dashboard designed for lecturers. It contains the general data about the course, detailed overview of the grades, attendances, assignments and other information regarding education processes. Source of the dashboard: (Podgorelec & Kuhar 2011).

Categorizing the dashboard plays the essential role in its development. It helps to distinguish and select the main features and concepts of the design. According to Few (2006), we can distinguish dashboards using several taxonomies, such as:

1. **the role of the dashboard:** analytical, operational, strategical
2. **the data domain:** executive, health-care, marketing, learning, performance
3. **the type of data:** quantitative, non-quantitative
4. **interactivity:** static display, interactive display

Data visualization analysis is widely used to predict customers' behavior and understand their motivation. However, the areas which could benefit from dashboards are not limited to the business environment only. Since students might be seen as customers within higher education institutions, the analysis and appropriate visualization of educational data can help to increase the efficiency of the studies and improve the learning process (Podgorelec & Kuhar 2011; Ruipérez-Valiente, Muñoz-Merino, & Delgado Kloos 2018). In such case, we talk about the *learning dashboards*.

Education is a representative example of the area which deals with rich datasets. In the e-learning platforms, the information about the users might be gathered from various sources of context, such as forums and chats activity, assessments timings and results, comments with rating (Santos & Boticario 2015). Combined with data analysis, the learning dashboards can lead teachers to discover students' performance patterns, predict problems and find motivational elements (Podgorelec & Kuhar 2011). The information about the factors which affect learning is one of the most investigated questions in education (Ruipérez-Valiente et al. 2018). The learning dashboard concept is based on such areas as the educational data mining, learning analytics, and information visualization. It is 'a single display that aggregates multiple visualizations of different indicators about learner(s), learning process(es) and learning context(s)' (Schwendimann et al. 2016). An example of the learning dashboard is shown in Figure 2.

We should design and develop dashboards appropriately to achieve their maximal usage and benefits. Researchers have shown that teachers hardly use raw statistical data of learning dashboards (Zorrilla, García, & Álvarez 2010). It is often more complicated to follow data presented textually than graphically. Thus, most of the researches regarding learning dashboards focus on identifying what data is meaningful to various stakeholders in the educational process, and how they should visualize these data (Schwendimann et al. 2016). Improving the data visualization quality and their logical organization is an important task not only for learning dashboards but all dashboards in general. Few (2006) highlights that poorly designed dashboards would cause inefficient and ineffective communication. The science regarding the visual perception and sense-making is complicated, and those who develop dashboards do not always master it. Quantitative design guidelines should be developed to help the developers to design and implement appropriate dashboards without diving into this science.

3 | GUIDELINES FOR DASHBOARD DEVELOPMENT

Few's definition of dashboard establishes the design requirement: present the information consolidated and arranged on a single screen so the information can be monitored at a glance. A user should be able to get quickly familiarized with the content of the dashboard and find something that deserves attention. Few have provided a framework based on a knowledge of famous books regarding design and graphics e. g., (Tufte 1983; Ware 2012). This framework contains guidelines for the dashboard design, including examples of well-designed dashboards. Examples of such guidelines are:

1. Eliminate the non-data pixels (decorations) to decrease distraction of users (based on Tufte (1983)).
2. Consider Gestalt principles to help a user recognize coherent groups of UI better (based on Ware (2012)).
3. Select appropriate charts and colors for emphasizing the relationship between data and highlighting the critical information (Figure 3).

Even more than ten years after the release of Few's publication, we can still observe that the majority of dashboards ignore Few's design guidelines or express them in their own way. We assume that the reason might be the complexity and vague definition of the framework and the lack of other sources which would provide formal and quantitative knowledge in the area of dashboard design. Guidelines are usually described qualitatively, and it is difficult to convert them into a set of strict rules. For instance, the selection of appropriate charts and colors usually depends on an actual context, and it cannot be completely generalized. Also, it is difficult to describe Gestalt principles formally as summarized by Jäkel, Singh, Wichmann, and Herzog (2016). Hence, design guidelines are usually simplified into basic rules with limited usage. We described an example of simple quantitative dashboard design metrics and guidelines (Hynek & Hruška 2016). We have shown that we can distinguish the well-designed dashboards by the measuring their overall colorfulness or analysis of their histograms. However, the metrics consider UI screen as a matrix of pixels which does not correspond with user perception.

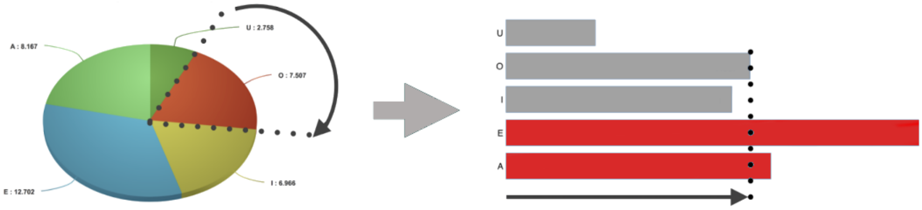


FIGURE 3 An example of the design guideline. It is easier to compare the values using the bar chart. It is also better to use one subtle color than several vivid colors which should be used to emphasize important information (e. g., the values higher that a limit).

One possible step in improving the metric-based evaluation is to analyze the objects (*widgets*) in the screen and their properties (e. g., size or position) as described by Charfi et al. (2014). The objects are usually represented by their boundaries. The analysis of objects in the screen is more similar to the real perception of a screen by a human than the analysis of particular pixels (objects within a scene as described by Baker, Jones, and Burkman (2009)). Ngo et al. (2003) have published an example of advanced object-based metrics. They provide 13 metrics measuring aesthetic aspects of the screen. We chose the two metrics—Balance and Symmetry—to test the applicability of the generator of UI samples. Hence, we focus only on these three metrics in the following text.

Ngo (2001) defines the chosen metrics as:

1. Balance is 'the difference between total weighting of components on each side of horizontal and vertical axis' (Figure 4).

2. Symmetry is 'the extent to which the screen is symmetric in three directions: vertical, horizontal, and diagonal' (Figure 5).

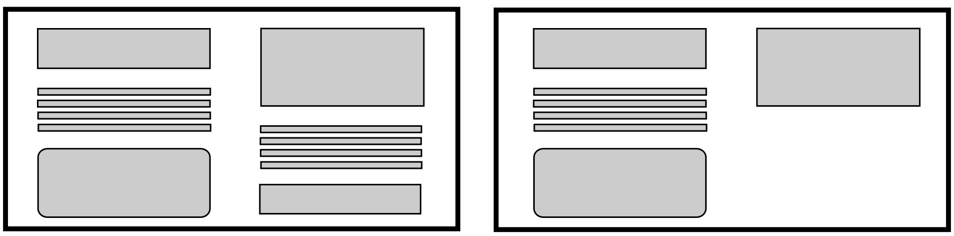


FIGURE 4 A balanced (left) and unbalanced (right) screen.

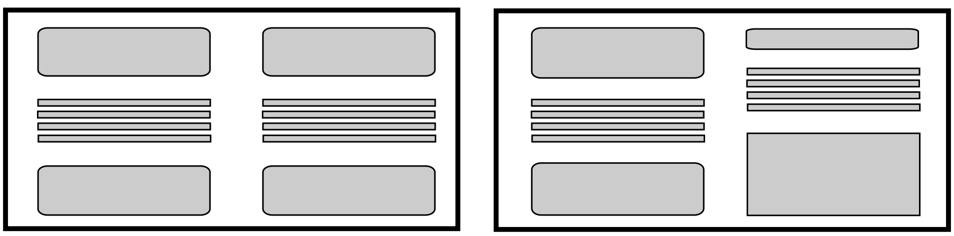


FIGURE 5 A symmetric (left) and asymmetric (right) screen.

The metrics provide an advanced approach for user interface analysis. The designer specifies the rectangular boundaries of the UI objects. Then, the boundaries are used as inputs for the formulas of the metrics.[1] Results of the metrics are the values in the range $\langle 0, 1 \rangle$ and they are used as inputs for the regression function measuring overall level of aesthetics (Ngo & Byrne 2001). An example of the tool using the metrics of aesthetics of UI has been designed by Zen and Vanderdonckt (2014) or Hynek and Hruška (2019). The aesthetics has an impact on the overall satisfaction of users with the product (Liu 2003), and it might influence the usability of user interfaces (Tractinsky et al. 2000). Hence, we could create design guidelines based on the metrics of the aesthetics.

[1] Readers can find the formulas of the metrics in Ngo et al. (2003)

3.1 | Specification of the Problem

The metrics of aesthetics have, however, one significant limitation. They consider only the position and size of the widget boundaries. The question is: *Are there any other UI characteristics which should be considered by the metrics?* Ngo et al. (2003) have suggested that color and type of widgets should be involved in the metrics' formulas. Figure 6 explains the hypothesis. Several researchers evaluated the applicability of the metrics. (Altaboli & Lin 2011; Bauerly & Liu 2008; Salimun et al. 2010) used the synthetic layouts. (Mazumdar, Petrelli, Elbedweihy, Lanfranchi, & Ciravegna 2015; Purchase, Hamer, Jamieson, & Ryan 2011; Reinecke et al. 2013; Zain, Tey, & Goh 2011; Zen & Vanderdonckt 2014; Zheng, Chakraborty, Lin, & Rauschenberger 2009) worked with real interfaces, usually web pages. None of them considered evaluating the impact of color and type of widgets on the usability of the metrics. We believe that these characteristics are significant. Dashboard usually contain various kinds of charts using various colors. Hence, the metrics should consider color and type of dashboard UI objects.

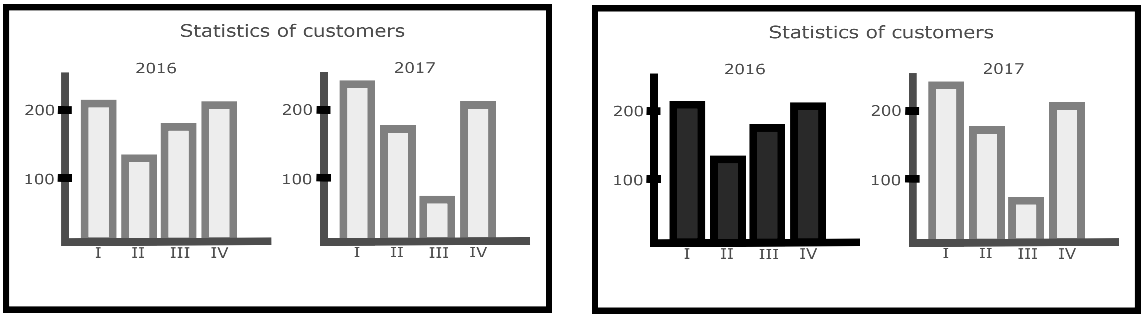


FIGURE 6 Impact of color on the metrics of aesthetics. Both screens would be rated by the same value of Balance and Symmetry according to formulas of Ngo et al. (2003) since they consider only the dimensions of widgets. We suggest that a user would distinguish these two screens since they consist of a different color. For instance, we expect that user would rate the left screen as more balanced and symmetric than the right one because of unbalanced and asymmetric color distribution in the right screen.

One possible way to find the prove or disprove the hypothesis is to let users rate the UI samples containing the same layout of widgets, but varying in the color and type of the widgets. Then, we can compare the results of the metrics with the reviews of the users. One of our tasks was to find the appropriate UI samples for such evaluation.

4 | EXISTING SOLUTIONS

It is not usually easy to find a sufficient number of samples which would meet specific requirements (e. g., the samples varying in one visual attribute). Hence, researchers often create synthetic samples comprised of simple shapes representing simplified widgets. Altaboli and Lin (2011) generate screens containing four black squares with different dimensions to test extreme values of chosen Ngo's metrics. Salimun et al. (2010) generate layouts comprised of triangles. Bauerly and Liu (2008) replace black squares with random images to make the displays look realistic. The advantage of this approach is the possibility to generate samples automatically. It allows the researchers to create a large number of samples quickly and precisely according to their requirements. The researchers can analyze specific interface characteristics thoroughly without the distraction of the users caused by any other visual aspects of the interface. The disadvantage is a high simplicity and low variability of the samples because they are usually generated according to strict and simple rules. The samples might miss many visual aspects, which also influence the user perception. The results of evaluations might not reflect the reality.

Another approach is based on programming languages for building UIs and data visualizations—such as R language or JavaScript with a combination of frameworks such as React.js or Vue.js, and libraries for the data visualization: Chart.js, Highcharts or D3.js (Jacobs & Rudis 2014). This approach can significantly improve the look of generated UI samples since the researchers can replace simple shapes with real UI widgets. It helps to make the results of evaluations closer to reality. The disadvantage of the approach is, however, its elaborateness. It usually requires to implement many lines of code to generate a set of complex interface samples. The researchers should not place the widgets arbitrarily. Design experience and knowledge of programming languages of the researchers is required.

Existing tools for building UIs (e. g., the dashboard builders described in Table 1) make the compromise between the previous two approaches. The researchers can quickly prototype real user interfaces connected with real data. The tools usually provide a palette of prearranged widgets which can be interactively placed into a canvas and adjusted according to requirements. The tools also implement connectors for connecting the designed UI with datasets, such as:

- connections with external services (Google Analytics or Facebook)
- uploading files from the computer in various formats (JSON, CSV or XML)
- connections to custom databases
- uploading by email
- external APIs

Users do not need to have experience with programming languages and frameworks. The tools, however, provide lower flexibility and usually miss the possibility of modification (e. g., export samples in the specific format for further analyses). Also, it is not usually possible to generate samples according to predefined rules (e. g., predefined layout, set of colours and widgets). Such rules can be specified declaratively (e. g., set of available colors) or imperatively (e. g., heuristics handling distribution of colors).

There exists a range of commercial on-line tools for dashboards development. We analyzed and compared popular available applications. Table 1 compares these applications according to the current research criteria based on the requirements described in the section 5. The main problem of the analyzed tools is the inability to modify and customize them according to our requirements (e. g., the possibility to specify own model).

TABLE 1 Comparison of the chosen popular on-line web applications for the development of dashboards (August, 2018).

	License	Customization of widgets	Extensibility of tool	Import/Export
Klipfolio.com	paid / trial	available	limited	JSON
Datapine.com	paid / trial	available	not available	not available
Sisense.com	paid / trial	available	not available	JSON
Plot.ly	paid / 25 free samples	limited	not available	JSON for single graphs
theDash.com	paid / publicly saved results	limited	limited	not available

5 | FRAMEWORK FOR IMPROVEMENT OF METRICS AND DESIGN GUIDELINES

Firstly, we established our requirements for the tool for the creation of UI samples:

1. the possibility to specify the structure and appearance of the UI samples effortlessly, quickly, in a declarative way and without the knowledge of implementation details of the tool. We preferred the known serialization formats (e. g., XML or JSON). For the purpose of this document, we call this requirement as *ease of use*.
2. the possibility to specify only a subset of the significant UI characteristics and let the rest of the characteristics to be set implicitly. For instance, we wanted to visualize charts without the need to specify a dataset. We call the requirement *simplicity*.
3. the possibility to simply change the values of specified attributes, so that we can create multiple instances of the same UI varying in a specific UI characteristic. We needed to prepare samples suitable for analyses of the dependency between the UI characteristic on the user perception. We call the requirement *flexibility*.
4. the possibility to extend the tool, add new widgets, update the model or modify the tool according to actual purposes. We call the requirement *extensibility*.

Based on the requirements, we designed a framework for the construction and evaluation of metrics and design guidelines. It defines the process of iterative improvement of UI model and incremental generation of improved sets of UI samples which can be used for an extension of the knowledge about user experience and improvement of metrics and design guidelines. Figure 7 describes the process.

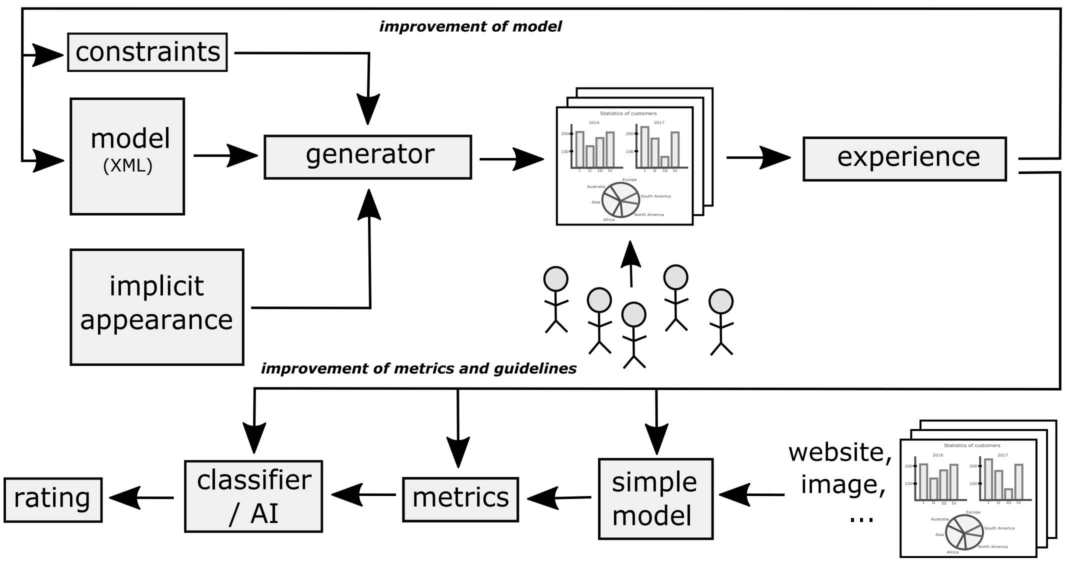


FIGURE 7 The scheme of the construction and evaluation of design guidelines. The initial implicit appearance of UI solves the problem of cyclic dependency described in Figure 1. We can use the user experience for the improvement of the model (for the generation of better samples) and the improvement of metrics and design guidelines.

Listing 1: An example of the UI internal representation in XML language. It describes a dashboard with the resolution: 1280x1024 px. It stores graphical elements. One of the elements represents a bar chart which contains attributes as location, width or height:

```
<dashboard>
  <width>1280</width>
  <height>1024</height>
  <graphicalElement>
    <type>BarChart<type>
    <x>80</x>
    <y>60</y>
    <width>200</width>
    <height>100</height>
    ...
  </graphicalElement>
  <graphicalElement>
    ...
  </graphicalElement>
  ...
</dashboard>
```

The process defined by the framework consists of 2 parts:

1. **Improvement of UI model:**

   (a) At the beginning, the generator of UI samples works with a simple model of UI internal representation which specifies the structure and appearance of a UI. It allows to specify only dimensions and types of interface objects. An example of a serialized UI internal representation is described in Listing 1. The model meets the first requirement—ease of use. It provides a fundamental variability of generated samples. The rest of the information is derived from the implicit appearance which meets the second requirement—simplicity. The implicit appearance defines the basic style of graphical elements and their data sets. It is specified in style templates used by the generator.

   (b) Then, the researchers use the model to generate an initial set of UI samples which are used for making an initial conception of the UI characteristics which are important for the perception of UI quality.

(c) The model is extended iteratively according to the user experience which is obtained from the analysis of the user reviews reviewing the generated samples. The generator should be flexible and extensible (the third and fourth requirements), so it can work with new instances of the model. The more comprehensive the model is, the higher level of the samples variability the generator provides. Researchers can use the constraints to filter unimportant types of samples when they want to analyze the particular design aspect. **In this research, we use specific layouts (Figure 15) which represent the constraints.**

2. **Improvement of metrics and design guidelines:** The lower part of Figure 7 describes the applicability of the user experience in the construction of the quantitative design guidelines which evaluates UI in terms of the ratios of UI characteristics measured by metrics. Depending on a situation, the guidelines can represent a simple threshold or an advanced classification algorithm. Researchers should deal with the following problems:

   (a) A real UI can be represented in various formats (a raster image or a structured description—e. g. a web page). The researchers should find a way how to convert the original UI into internal representation compatible with the UI model which is recognized by the metrics. The model needs to be simplified which might limit the metrics. It is not the matter of this research to discuss possible approaches of interface processing. For instance, the readers can find inspiration in the research of Reinecke et al. (2013).

   (b) Researchers should use the user experience to improve metrics so the metrics reflect the user perception.

   (c) Researchers should use the user experience to improve design guidelines so the guidelines reflect user perception and are able to use results of metrics for analysis of design problems, classification or rating the user interface.

Since the existing tools mentioned in Section 4 do not satisfy the requirements of the framework, we decided to develop a custom generator of realistic-looking dashboard interfaces samples. We made a compromise between two approaches: developing a dashboard with the help of programming languages, and using one of the available on-line services.

6 | GENERATOR OF DASHBOARDS

We developed the generator as Single-Page Application. It loads a single HTML page consisted of a canvas and palette of dashboard components which can be dynamically placed into the canvas by a user. The architecture of the developed generator is shown in Figure 8. The back-end supports RESTful API, which allows easier scalability of the application and independent changes of the server and client sides. It also allows extending the generator to support the construction of other interfaces types (e. g., mobile applications UI). The back-end is based on Node.js environment and MongoDB database, which allows to store the model directly in JSON format (Holmes 2015). The front-end is built with the Vue.js framework the using UXgraph library, which we primarily developed for this generator. Another advantage of the developed application is that it may be extended to a hybrid mobile application, using Cordova wrapper.

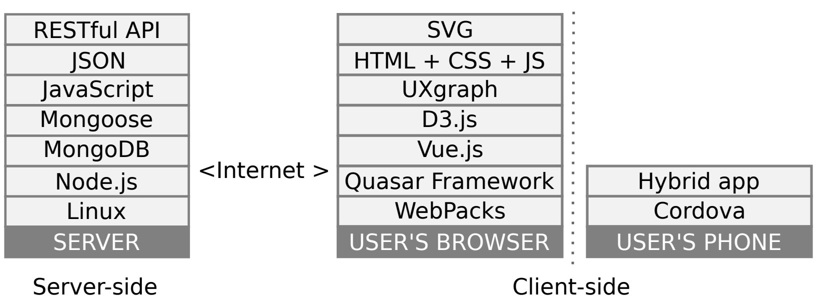


FIGURE 8 The architecture of the generator. It illustrates technologies used for the back-end and front-end as independent applications. The front-end may be wrapped to a hybrid mobile application.

6.1 | Visualization of Data

The crucial part of the generator design was to provide a tool for data visualization. For this purpose, we have developed the UXgraph library combining the advantage of the Vue.js framework for building user interfaces with the D3.js library for data visualization. The reason behind creating a unique library was to have a predefined set of reusable and easily scalable widgets. They would use the same model but with the possibility of applying the different styles. UXgraph provides every widget as a separate Vue.js component, and it contains all code which might be repeated. Then, these components extend basic HTML tags with new custom ones. Every widget type component is located in own single file template, which contains HTML, scripts for the declaration of properties and behavior, and styling (Figure 9). Beta version of the library is available for all Node.js-based projects under MIT License as open-source npm extension (Node Project Manager).[2]

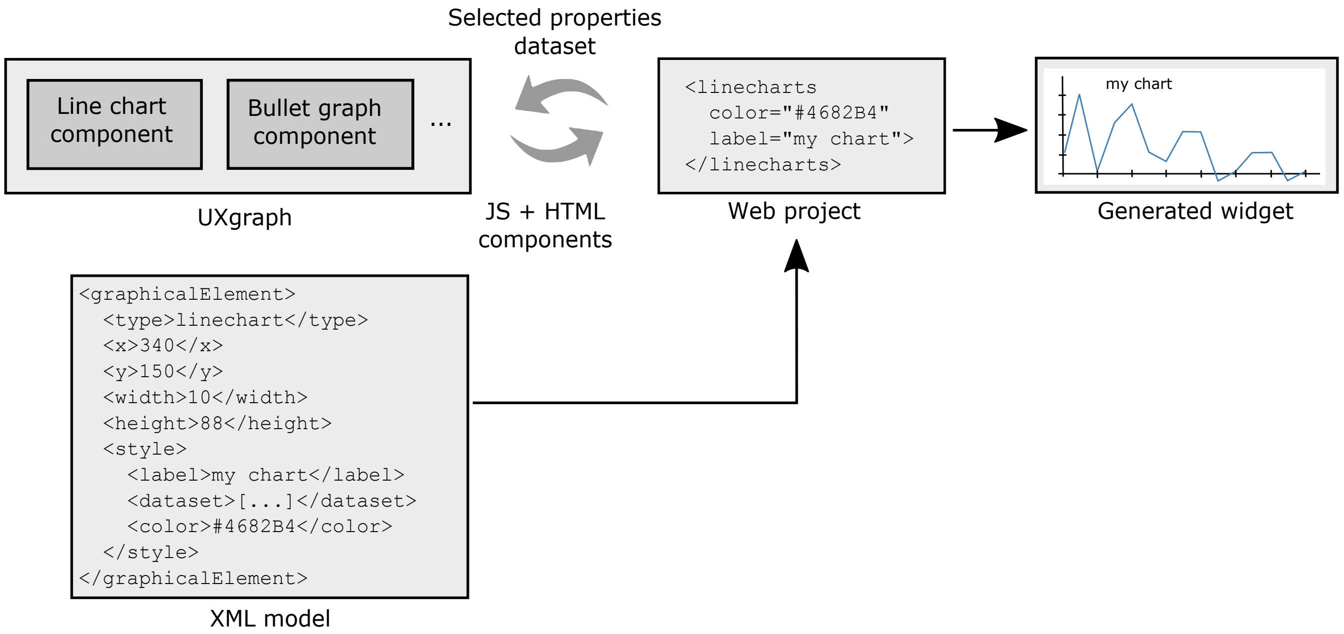


FIGURE 9 Custom HTML tags. New HTML tag is defined for every widget type with the help of UXgraph library.

We based all widgets of this extension on available qualitative design guidelines (Few 2006; Tufte 1983), providing the implicit appearance of widgets. The current version of UXgraph contains the following chart types:

1. bullet graphs, developed by Stephen Few specifically for dashboards, aiming to replace gauges and meters (Few 2006);
2. bar charts (horizontal and vertical), used to display multiple instances of one or more key measures;
3. stacked bar charts (horizontal and vertical), which should be used when it is needed to decompose and compare several instances of a whole;
4. line chart, which should be used to show the shape of data movement;
5. dynamic sparklines, invented by Edward R. Tufte (Tufte 1983) to provide compressed forms of data display.

We have constructed the widgets with the help of the D3.js library which allows rendering of graphical elements in the SVG format. Vue.js library supports the reactivity of the widgets and the whole interface. In particular, UXgraph uses Vue.js *mounted* event to call a method for creating the widget after a component instance is rendered.

When calling the method to create a component instance, we also need to specify its parameters, which represent customizable settings of the component. For this purpose, we use another Vue.js feature which provides the possibility to pass data to the component and set its properties. Since components are reusable and can be inserted basically in any place of the web page, it is essential to keep them in their own isolated scope. The styling parameters from the front-end are passed to the widget components with the help of Vue.js properties (*props*). As a result, we can use the properties anywhere during the SVG construction.

In the beginning, we specify properties for all child components (which contain widgets' templates) using the *props* option:

```
Vue.component('sparkline', {
```

[2]UXgraph NPM package: https://www.npmjs.com/package/ux-graph

```
        // props declaration
        props: ['data', 'color', 'circle', 'label']
})
```

Afterwards, every property can be referred as *this.propertyName* within a component. Then it can be passed from the parent template like following:

```
<sparkline
        color="#4682B4"
        label="Daily defects"
        circle=true>
</sparkline>
```

Finally, we can see the result of the code in Figure 10. We specified default properties for all UXgraph components to make including of a new component even easier. Users need to specify a property only if they want to use custom values.

FIGURE 10 Example of the generated sparkline component, with custom parameters (left) and bar chart component included without any parameters (right).

Vue.js also allows updating the widgets dynamically, when their settings are changed by the user. Vue.js function *v-bind* allows to bind *props* to a certain data on parents. For example, when a user selects a new color for a chart, this parameter dynamically flows down to the child, and the chart is updated without page reload. The example of usage is:

```
<input v-model="parentColor">
<sparkline
        data="[3,2,1,4]"
        v-bind:color="parentColor">
</sparkline>
```

We also developed custom watch functions to redraw widgets when a property is changed without reloading of a page. Every watcher is triggered when a specific property is changed, and then it calls a method which is responsible for drawing the SVG.

6.2 | Generation of Samples

Figure 11 shows the algorithm of the generation of a set of the dashboards interfaces. A user needs to select several parameters, such as the number of samples, template, color scheme and layout density. Afterward, the generator requests the database to get the current version of the needed template XML model, as well as the model of its XML components. Default properties are specified for all UXgraph components to make the inclusion of a new component easy. The users need to specify a property only if they want to use custom values. Based on the retrieved data and the defined parameters, a set of the interfaces is generated. The generator has several built-in templates which are used for the first iteration of samples creation. After a feedback from the reviewers is received, the model of these templates can be changed accordingly.

Along with automated algorithm of dashboards generation, the application allows creating new templates and single dashboards manually. In this case, a user can add any number of widgets to the dashboards, select different color schemes for each of them, change their dimensions and position on the grid. The only requirement is to respect predefined constraints which help to quickly create realistic-looking samples (e. g., positions of widget need to respect grid layout). Every created configuration can be saved as an interface example and exported to the PNG or XML formats. Figure 12 describes the main possibilities of user's interaction with the application and Figure 13 displays the main UI for the dashboard development. All widgets are draggable and resizable. These features allow quick manipulation with their sizes and positions directly on the canvas.

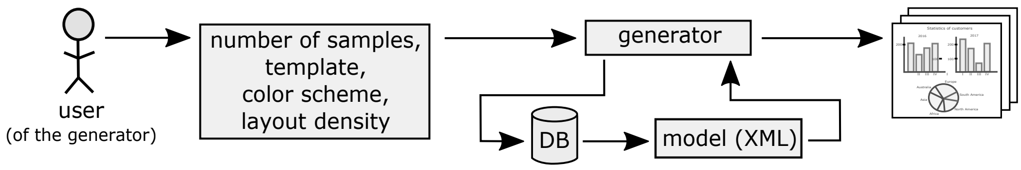


FIGURE 11 Example of the generator workflow.

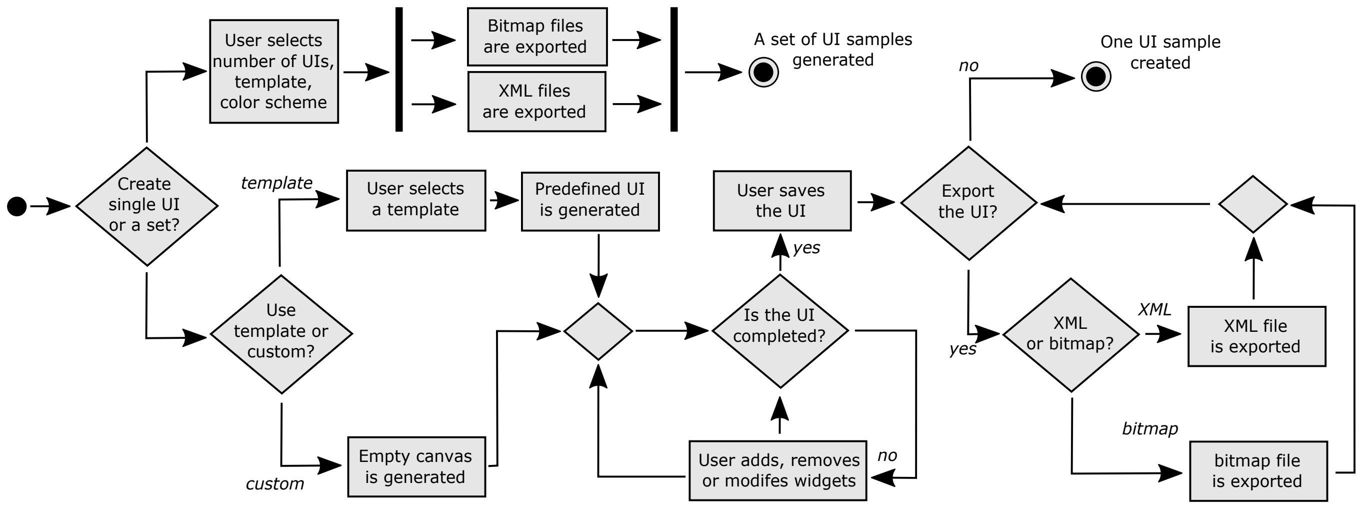


FIGURE 12 The activity diagram
describing the workflow of preparation of UI samples by a user. The user can generate either a set of UI samples or create one sample.

The described generator's structure solves the problems listed in Section 5. It is not needed to know the implementation details of the application to use it. A user can interact with the UI only. It allows to save or export the results effortlessly. It is possible to define only the needed properties quickly. Sometimes, it is not required to specify the properties at all (if random or default values are suitable in this case). It is also possible to quickly change selected characteristics of selected widgets to create UIs varying in a specific attribute only, which makes developed generator very flexible. Finally, extending the widgets list with a new type is as comfortable as adding new Vue.js components to the UXgraph library.

7 | EVALUATION

We performed a small-scale study to evaluate the functionality of the generator. The study analyzed the impact of the color, type of widgets and displayed dataset on the dashboard balance and symmetry as described in Section 3. Firstly, we created four layouts: d_1, d_2 for the evaluation of the Balance metric and d_3, d_4 for the Symmetry metric (Figure 15). The layouts can be considered as highly balanced and symmetric with respect to the formulas of Balance and Symmetry presented by Ngo et al. (2003). Then, we generated a few realistic-looking dashboards for every layout. The dashboards varied in the color (color intensity and hue), chart types (bar charts, line charts, and bullet graphs) and dataset which changes the look of widgets as well (e. g., a higher value is represented by a larger bar which makes the area which is occupied by a particular color value).

Then, we let users rate the UI samples. The evaluation of the Balance metric was performed with 12 users; the evaluation of the Symmetry metric with a different group of 13 users. Both evaluations were independent. The demographic and professional characteristics of participants varied in both cases. The age of participants was 22-50 years, experience with information technology varied: some of them were students in these areas, some of them were employed, and the rest of them had only minimal experience with information technology.

The first group of users was asked to rate the horizontal and vertical balance (using the 5-point scale: $\langle -2, 2\rangle$; -2: the left/bottom side is heavier; 0: balanced; 2: the right/upper side is heavier). The second group was asked to rate overall symmetry (using the 5-point scale: $\langle 0, 5\rangle$; 0: asymmetric; 5: symmetric). Then, the participants of both groups rated the first impression (using the 5-point scale: $\langle 0, 5\rangle$; 0: very low, 5: very high) of the generated screens. They used an interactive form which let them quickly select locations of perceived equilibrium or the rank of the dashboards (Figure 14). This form is not part of the discussed generator, but an illustration of the usage of the generated samples.

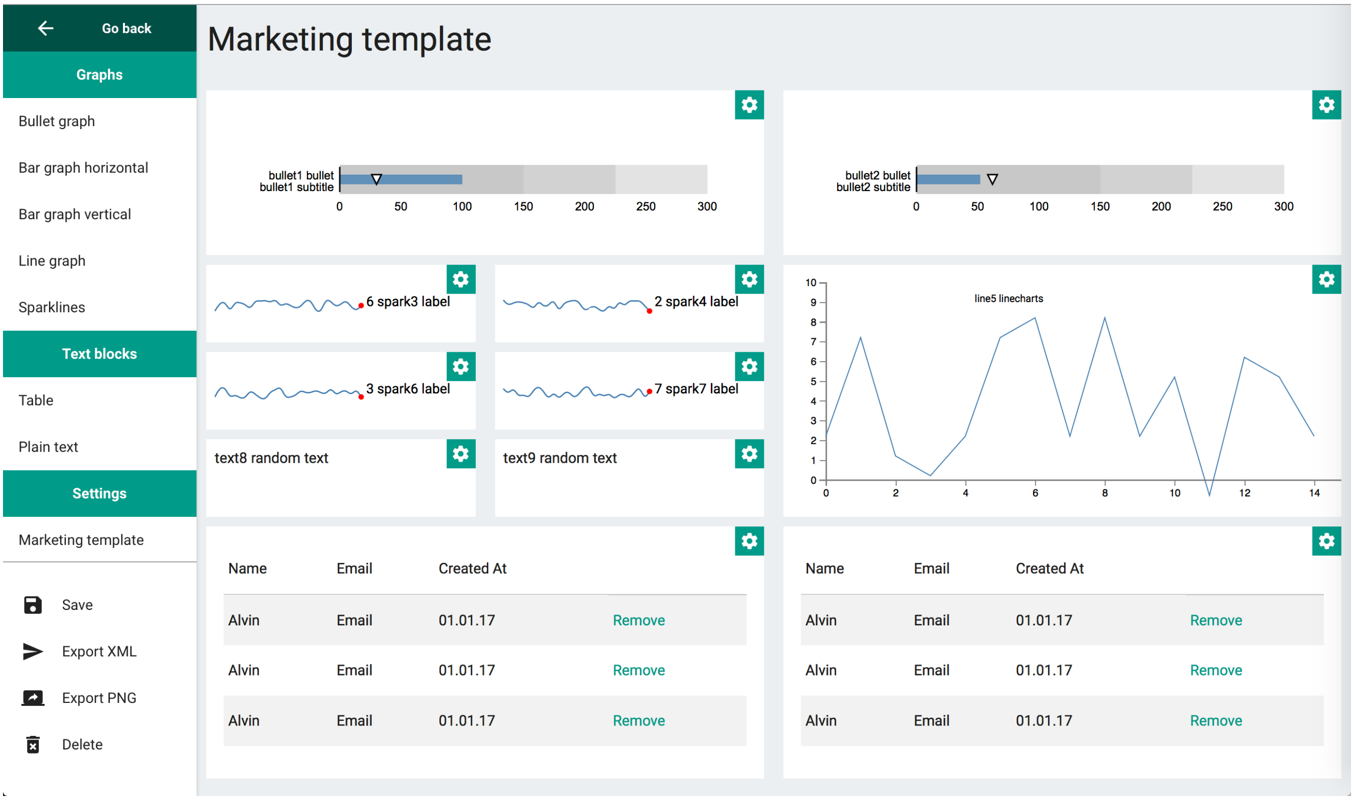
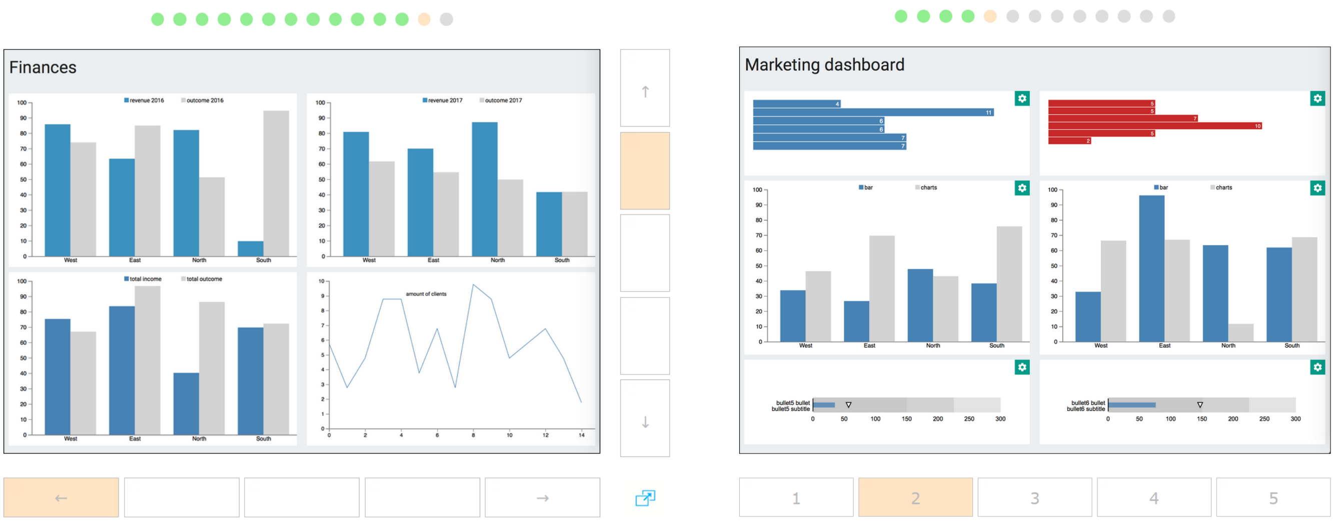


FIGURE 13 Graphical editor for manual creation of dashboard samples. The left sidebar displays a list of all available widgets, which can be added to the grid using drag-n-drop. The body of dashboard is based on a predefined grid. Designers export the dashboard in XML and PNG format for further analysis.



(a) vertical and horizontal balance

(b) overall symmetry

FIGURE 14 An example of the form pages for rating balance (a) and symmetry (b). The weight of the dashboard (a) is perceived on the left and upper sides of the dashboard. The symmetry of the dashboard (b) is rated with low rate of symmetry. The round buttons on the top represents the form pagination. The users went through a set of samples and selected values of UI characteristics according to their subjective perception.

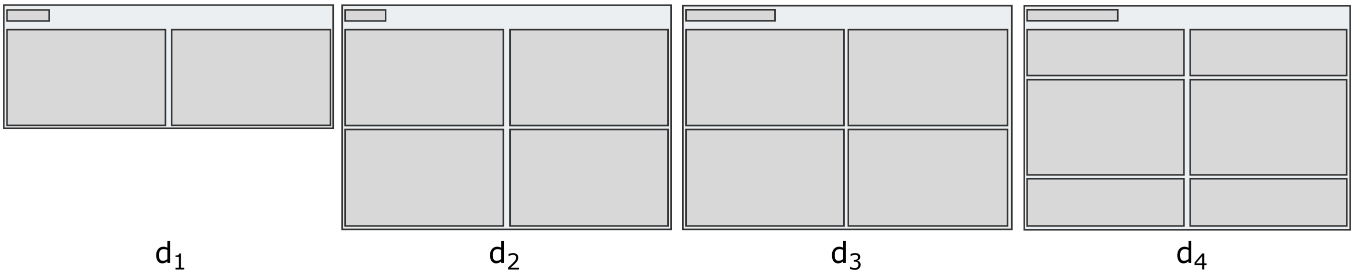


FIGURE 15 The four layouts used for the analysis of user perception and comparison of the users' reviews with results of metrics. They are highly (but not absolutely) balanced and symmetric. Then, we replace the layout regions with different charts, colors and dataset and ask the users to rate Balance and Symmetry of the screen. The layouts d_2 and d_3 are basically the same. They differ in margins and they are used for different metrics.

8 | RESULTS

The results of the user testing have confirmed the impact of the color, type, and dataset of widgets on both dashboard characteristics. Readers can find the results including the surveys and all dashboard samples on-line[3].

8.1 | Balance

Figure 16 presents the average values of horizontal and vertical balance. The layout d_1 tested the change in the perception of the horizontal balance. The layout d_2 tested both axes—horizontal and vertical. Since we composed the reference layouts d_1, d_2 only from gray rectangles, users perceived them as relatively balanced. Replacing the rectangles by real widgets did not changed the level of perceived balance of $d_1^{(1)}$ much. However, even a low change of color intensity of the chart on one side caused a high deviation from the equilibrium ($d_1^{(2)}$). Increasing the color intensity on one side made the layout even more unbalanced. On the contrary, color hue had a low impact on perceived balance ($d_1^{(5)}$).

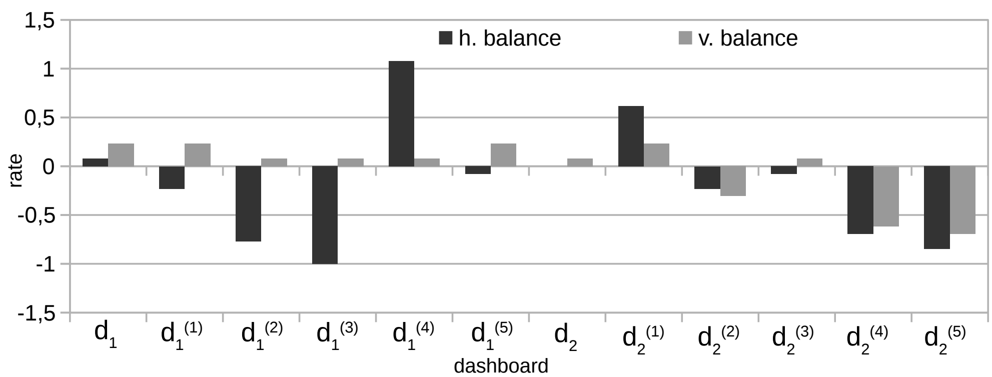


FIGURE 16 The average values of horizontal and vertical balance perceived by the users for the two layouts d_1 and d_2. The reference layouts are highly balanced (close to zero) on the contrary to other dashboards using the same layout but different colors and widget types.

We observed similar results in the layout d_2. In contrast to the layout d_1, readers can notice the change in vertical and horizontal value of balance. Layout d_2 was also used to analyze the impact of widget types. We used a bar chart and line chart. The bar chart was rated as visually heavier than the line chart because rectangles of bar charts usually occupy a larger area of the screen than lines of line charts.

Finally, we did not detect any correlation between the perceived balance and overall first impression of the users. In the future, this evaluation can be repeated with a higher number of samples and users.

[3]Available at: `https://www.fit.vutbr.cz/~ipastushenko/evaluation/form.php`

8.2 | Symmetry

In contrast to the evaluation of the Balance metric, we did not let the users rate symmetry of the screens for every axis separately (for instance,Ngo et al. (2003) consider vertical, horizontal and radial axes). Users directly rated the overall symmetry. Figure 17 presents the results. The reference layouts d_3 and d_4, which were composed of gray rectangles, were rated as highly symmetric. Then, we replaced the rectangles with real widgets of the same type and color. The average values of symmetry of these screens were even slightly higher ($d_3^{(1)}$, $d_4^{(1)}$). Then, we modified widgets of one side o UI, particularly: dataset ($d_3^{(2)}$, $d_4^{(2)}$), color ($d_3^{(3)}$, $d_3^{(4)}$, $d_4^{(3)}$, $d_4^{(4)}$) and type ($d_3^{(5)}$). All modifications caused a decrease in the perceived value of symmetry. The results of the evaluation showed a similar tendency as the results of the evaluation of UI balance. Color, type of widgets and the displayed dataset have impact on perceived weight of objects.

Finally, we did not detect any correlation between the perceived symmetry and overall first impression.

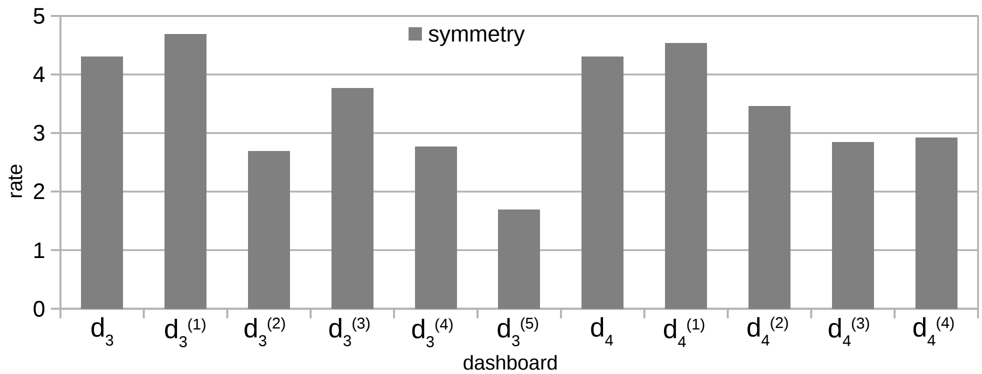


FIGURE 17 The average values of overall symmetry perceived by the users for the two layouts d_3 and d_4. The reference layouts are highly symmetric (close to 5) on the contrary to other dashboards using the same layout but different colors, widget types, and dataset.

9 | DISCUSSION

The small number of samples and users represents the main limitation of the presented evaluation. On the other hand, the primary purpose of the evaluation was not to provide a large-scale study of object-based metrics. The main purpose was to evaluate and demonstrate the applicability of the designed workflow and generator. We performed two studies with two groups of users analyzing different visual characteristics, as discussed in Section 7. Specialization and age of the users varied. We successfully gained the experience of the users. We did not find any correlation between the results and characteristics of users. However, it would be useful to perform a further study with more users analyzing the impact of users characteristics (e. g., age, specialization) on the UI perception. For instance, the perception of people having skills in the art might be different from other people.

The results of the users' reviews confirmed the hypothesis which expected the color, type, and dataset of widgets to be relevant during the perception of layout balance and symmetry. The users tended to perceive the charts using highly intense colors as more weighty comparing the charts using less intense colors. Similarly, the charts containing large graphical elements (e. g., bar charts) were perceived as more weighty than the charts composed from thin lines (e. g., line charts). Finally, the displayed datasets affected graphical elements of charts (e. g., size of bars) which also affected the perception of the weight of charts. The findings of the study are important for the application of object-based metrics which evaluate the weight of UI objects and work only with dimensions of objects. We can use the results for the improvement of metrics—e. g., the Balance and Symmetry metrics designed by Ngo et al. (2003). One possible suggestion is to consider the average color intensity of widgets in the formulas (Hynek & Hruška 2018).

On the other, we have not proven the direct correlation between perception of the balance or symmetry with overall first impression. One reason might be an insufficient number of users and samples. There is also a possibility that users do not prefer maximal values of Balance and Symmetry as suggested by Salimun et al. (2010). It would be useful to make further investigations of the optimal distributions of object weights which are preferred by users. We expect that a large-scale study with a larger set of samples might provide us with a better experience and we would be able to create a design guideline recommending the optimal value of Balance and Symmetry.

There are several ways how to extend the described generator. We could extend the UXgraph library—add new widgets and extend the possibilities to change their appearance by adding more attributes. Then, we could extend the XML model to increase the possible variability of generated samples. We could also support other types of user interfaces, not only dashboards. A good example would be the support of mobile interfaces since they have similar design requirements as dashboards: (1.) the users should perceive the most important information at a glance, and (2.) it is better to present information on a single screen (without scrolling). The generator already supports these requirements. Moreover, the generator prevents other design problems, like the overlapping of the widgets or making them too small. Finally, we could improve the algorithm representing the rules and constraints generating test samples. We could also design a language for the description of the generation rules in a declarative way.

10 | CONCLUSION

This paper described the problem of generating the realistic-looking interfaces used for the design and evaluation of quantitative UI metrics and guidelines. We created the framework for design and evaluation of new metrics using a generator of UI samples. We established four requirements of the generator and named them—ease of use, simplicity, flexibility, and extensibility. We analyzed existing UI builders of dashboards and verified the requirements. The tools were usually suitable and user-friendly for a thorough design of one dashboard, connected with various services providing data. On the other hand, we were unable to extend or modify the tool to quickly generate more samples which would differ in specific characteristics. Hence, we designed and implemented own generator which: (1.) works with XML model describing appearance of dashboard; (2.) uses implicit values of unspecified characteristics; (3.) provides ability to generate more samples of dashboards; (4.) can be modified according to actual requirements (extension of model). We also created a prototype of the UXgraph library which provides reusable UI widgets. We used the library in the generator.

Then, we performed a small-scale study to demonstrate the usability of the generator. We analyzed the impact of color, type of widgets and displayed dataset on the perception of the layout balance and symmetry. Two independent user testings confirmed the impact of these factors. The user experience can be used for the improvement of the widget-based metrics Balance and Symmetry designed by Ngo et al. (2003). One possible suggestion is to consider the average color intensity of widgets in the formulas. It also seems to be worth evaluating more samples with more users. Such evaluation might tell us more about the relationship between the perception of balance and symmetry and overall first impression of the users.

ACKNOWLEDGMENTS

This work was supported by The Ministry of Education, Youth and Sports from the National Programme of Sustainability (NPU II) project 'IT4Innovations excellence in science – LQ1602'.

References

Albert, W., & Tullis, T. (2013). *Measuring the user experience: Collecting, analyzing, and presenting usability metrics.* Elsevier Science.

Altaboli, A., & Lin, Y. (2011). Investigating effects of screen layout elements on interface and screen design aesthetics. *Advances in Human-Computer Interaction*, *2011*, 1–10. doi: 10.1155/2011/659758

Baker, J., Jones, D., & Burkman, J. (2009). Using visual representations of data to enhance sensemaking in data exploration tasks. *Journal of the Association for Information Systems*, *10*(7), 533-559. doi: 10.17705/1jais.00204

Bauerly, M., & Liu, Y. (2008). Effects of symmetry and number of compositional elements on interface and design aesthetics. *Intl. Journal of Human–Computer Interaction*, *24*(3), 275–287. doi: 10.1080/10447310801920508

Charfi, S., Trabelsi, A., Ezzedine, H., & Kolski, C. (2014). Widgets dedicated to user interface evaluation. *International Journal of Human-Computer Interaction*, *30*(5), 408–421. doi: 10.1080/10447318.2013.873280

Deeb, R., Ooms, K., & De Maeyer, P. (2011). The influence of jacques bertin. In *25th international cartographic conference.*

Eckerson, W. W. (2010). *Performance dashboards: measuring, monitoring, and managing your business.* John Wiley & Sons. doi: 10.1002/9781119199984

Few, S. (2006). *Information dashboard design*. O'Reilly Sebastopol, CA.

Hollingsed, T., & Novick, D. G. (2007). Usability inspection methods after 15 years of research and practice. In *Proceedings of the 25th annual acm international conference on design of communication* (pp. 249–255). doi: 10.1145/1297144.1297200

Holmes, S. (2015). *Getting mean with mongo, express, angular, and node.* Manning Publications Co.

Hynek, J., & Hruška, T. (2016). Pixel-based analysis of information dashboard attributes. In *East european conference on advances in databases and information systems* (pp. 29–36). doi: 10.1007/978-3-319-44066-8_3

Hynek, J., & Hruška, T. (2018). Application of object-based metrics for recognition of well-designed dashboards. *International Journal of Human–Computer Interaction*, 1–13. doi: 10.1080/10447318.2018.1518004

Hynek, J., & Hruška, T. (2019). Segmentation of dashboard screen images: Preparation of inputs for object-based metrics of ui quality. In *Proceedings of the 14th international joint conference on computer vision, imaging and computer graphics theory and applications* (pp. 199–207). SciTePress - Science and Technology Publications. doi: 10.5220/0007312301990207

Ivory, M. Y., & Hearst, M. A. (2001). The state of the art in automating usability evaluation of user interfaces. *ACM Computing Surveys (CSUR)*, *33*(4), 470–516. doi: 10.1145/503112.503114

Jacobs, J., & Rudis, B. (2014). *Data-driven security: Analysis, visualization and dashboards*. Wiley.

Jäkel, F., Singh, M., Wichmann, F. A., & Herzog, M. H. (2016). An overview of quantitative approaches in gestalt perception. *Vision research*, *126*, 3–8. doi: 10.1016/j.visres.2016.06.004

Johnson, J. (2013). *Designing with the mind in mind: simple guide to understanding user interface design guidelines*. Elsevier.

Kolski, C., & Millot, P. (1991). A rule-based approach to the ergonomic "static" evaluation of man-machine graphic interface in industrial processes. *International Journal of Man-Machine Studies*, *35*(5), 657–674. doi: 10.1016/s0020-7373(05)80182-8

Levy, D. M. (2008). Information overload. In E. K. Himma & T. H. Tavani (Eds.), *The handbook of information and computer ethics* (pp. 497–516). Hoboken: John Wiley & Sons.

Lindgaard, G., Fernandes, G., Dudek, C., & Brown, J. (2006). Attention web designers: You have 50 milliseconds to make a good first impression! *Behaviour & information technology*, *25*(2), 115–126. doi: 10.1080/01449290500330448

Liu, Y. (2003). Engineering aesthetics and aesthetic ergonomics: theoretical foundations and a dual-process research methodology. *Ergonomics*, *46*(13-14), 1273–1292. doi: 10.1080/00140130310001610829

Mahajan, R., & Shneiderman, B. (1997). Visual and textual consistency checking tools for graphical user interfaces. *IEEE Transactions on Software Engineering*, *23*(11), 722–735. doi: 10.1109/32.637386

Malik, S. (2005). *Enterprise dashboards: design and best practices for it*. John Wiley & Sons.

Mazumdar, S., Petrelli, D., Elbedweihy, K., Lanfranchi, V., & Ciravegna, F. (2015). Affective graphs: The visual appeal of linked data. *Semantic Web*, *6*(3), 277–312. doi: 10.3233/SW-140162

Ngo, D. C. L. (2001). Measuring the aesthetic elements of screen designs. *Displays*, *22*(3), 73–78. doi: 10.1016/S0141-9382(01)00053-1

Ngo, D. C. L., & Byrne, J. G. (2001). Application of an aesthetic evaluation model to data entry screens. *Computers in Human Behavior*, *17*(2), 149–185. doi: 10.1016/S0747-5632(00)00042-X

Ngo, D. C. L., Teo, L. S., & Byrne, J. G. (2003). Modelling interface aesthetics. *Information Sciences*, *152*, 25–46. doi: 10.1016/S0020-0255(02)00404-8

Nielsen, J. (1994a). *Usability engineering*. Elsevier.

Nielsen, J. (1994b). Usability inspection methods. In *Conference companion on human factors in computing systems* (pp. 413–414). doi: 10.1145/259963.260531

Pastushenko, O., Hynek, J., & Hruška, T. (2018). Generation of test samples for construction of dashboard design guidelines: Impact of color on layout balance. In *World conference on information systems and technologies* (pp. 980–990). doi: 10.1007/978-3-319-77712-2_93

Podgorelec, V., & Kuhar, S. (2011). Taking advantage of education data: Advanced data analysis and reporting in virtual learning environments. *Elektronika ir Elektrotechnika*, *114*(8), 111–116. doi: 10.5755/j01.eee.114.8.708

Preece, J., Rogers, Y., & Sharp, H. (2015). *Interaction design: beyond human-computer interaction*. John Wiley & Sons.

Purchase, H. C., Hamer, J., Jamieson, A., & Ryan, O. (2011). Investigating objective measures of web page aesthetics and usability. In *Proceedings of the twelfth australasian user interface conference-volume 117* (pp. 19–28).

Reinecke, K., Yeh, T., Miratrix, L., Mardiko, R., Zhao, Y., Liu, J., & Gajos, K. Z. (2013). Predicting users' first impressions of website aesthetics with a quantification of perceived visual complexity and colorfulness. In *Proceedings of the sigchi conference on human factors in computing systems* (pp. 2049–2058). doi: 10.1145/2470654.2481281

Ruipérez-Valiente, J. A., Muñoz-Merino, P. J., & Delgado Kloos, C. (2018). Improving the prediction of learning outcomes in educational platforms including higher level interaction indicators. *Expert Systems*, e12298. doi: 10.1111/exsy.12298

Salimun, C., Purchase, H. C., Simmons, D. R., & Brewster, S. (2010). Preference ranking of screen layout principles. In *Proceedings of the 24th bcs interaction specialist group conference* (pp. 81–87).

Santos, O. C., & Boticario, J. G. (2015). User-centred design and educational data mining support during the recommendations elicitation process in social online learning environments. *Expert Systems*, *32*(2), 293–311. doi: 10.1111/exsy.12041

Schwendimann, B. A., Rodríguez-Triana, M. J., Vozniuk, A., Prieto, L. P., Boroujeni, M. S., Holzer, A., … Dillenbourg, P. (2016). Understanding learning at a glance: An overview of learning dashboard studies. In *Proceedings of the sixth international conference on learning analytics & knowledge* (pp. 532–533). doi: 10.1111/exsy.12298

Sears, A. (1995). Aide: A step toward metric-based interface development tools. In *Proceedings of the 8th annual acm symposium on user interface and software technology* (pp. 101–110). doi: 10.1145/215585.215704

Seffah, A., Donyaee, M., Kline, R. B., & Padda, H. K. (2006). Usability measurement and metrics: A consolidated model. *Software Quality Journal*, *14*(2), 159–178. doi: 10.1007/s11219-006-7600-8

Smith, S. L., & Mosier, J. N. (1986). *Guidelines for designing user interface software* (Tech. Rep.). Citeseer. doi: 10.21236/ada177198

Tractinsky, N., Katz, A. S., & Ikar, D. (2000). What is beautiful is usable. *Interacting with computers*, *13*(2), 127–145. doi: 10.1016/S0953-5438(00)00031-X

Tufte, E. R. (1983). *The visual display of quantitative information.* Graphic Press.

Tullis, T. S. (1984). *Predicting the usability of alphanumeric displays* (Unpublished doctoral dissertation). Rice University.

Vanderdonckt, J. (1999). Development milestones towards a tool for working with guidelines. *Interacting with computers*, *12*(2), 81–118. doi: 10.1016/s0953-5438(99)00019-3

Vanderdonckt, J., Farenc, C., & on Tools for Working with Guidelines, I. S. I. G. (2000). *Tools for working with guidelines: Annual meeting of the special interest group.* Springer London.

Ware, C. (2012). *Information visualization: perception for design.* Elsevier.

Wexler, S., Shaffer, J., & Cotgreave, A. (2017). *The big book of dashboards: Visualizing your data using real-world business scenarios.* John Wiley & Sons.

Yigitbasioglu, O. M., & Velcu, O. (2012). A review of dashboards in performance management: Implications for design and research. *International Journal of Accounting Information Systems*, *13*(1), 41–59. doi: 10.1016/j.accinf.2011.08.002

Zain, J. M., Tey, M., & Goh, Y. (2011). Probing a self-developed aesthetics measurement application (sda) in measuring aesthetics of mandarin learning web page interfaces. *arXiv preprint arXiv:1101.1606*.

Zen, M., & Vanderdonckt, J. (2014). Towards an evaluation of graphical user interfaces aesthetics based on metrics. In *Research challenges in information science (rcis), 2014 ieee eighth international conference on* (pp. 1–12). doi: 10.1109/RCIS.2014.6861050

Zhang, L., Stoffel, A., Behrisch, M., Mittelstadt, S., Schreck, T., Pompl, R., … Keim, D. (2012). Visual analytics for the big data era - a comparative review of state-of-the-art commercial systems. In *Visual analytics science and technology (vast), 2012 ieee conference on* (pp. 173–182). doi: 10.1109/vast.2012.6400554

Zheng, X. S., Chakraborty, I., Lin, J. J.-W., & Rauschenberger, R. (2009). Correlating low-level image statistics with users-rapid aesthetic and affective judgments of web pages. In *Proceedings of the sigchi conference on human factors in computing systems* (pp. 1–10). doi: 10.1145/1518701.1518703

Zorrilla, M., García, D., & Álvarez, E. (2010). A decision support system to improve e-learning environments. In *Proceedings of the 2010 edbt/icdt workshops* (p. 11). doi: 10.1145/1754239.1754252

AUTHOR BIOGRAPHY

Olena Pastushenko. received her first Masters degree in Automation and Computer-Integrated Technologies at National Aviation University in Kyiv, Ukraine. Afterwards, she received her Engineering degree at the Brno University of Technology, Faculty of Information Technology, where she is currently enrolled as a Ph.D. student. Her research interests are centered around educational technologies, gamification, data visualization and evaluation of user interface usability.

Jiří Hynek is a student in a Ph.D. degree programme at the Brno University of Technology, Faculty of Information Technology. He focuses on issues connected with the evaluation of user interface quality and design of quantitative metrics and guidelines for UI analysis.

Tomáš Hruška is a graduate of the Brno University of Technology, Czech Republic. Prof. Hruška received his CSc. (Ph.D.) in Computer Science and Engineering from the Brno University of Technology, Czech Republic. He has worked in the field of Information Systems and Visualization.

How to cite this article: O. Pastushenko, J. Hynek, and T. Hruška (2018), Evaluation of User Interface Design Metrics using Generator of Realistic-Looking Dashboard Samples, *Expert Systems*, *xxxx*.