



Yu. I. Hrytsiuk, O. R. Ferneza

Lviv Polytechnic National University, Lviv, Ukraine

REFLECTION OF EXPERT-BASED EVALUATION OF SOFTWARE QUALITY

The method of reflections data, which is obtained as a result of processing expert-based evaluation of software quality by different criteria of evaluation using petal charts. Reflection of expert-based evaluation of software quality is found to be the process of reflections data in graphical form for maximum ease of understanding and rapid perception, and also providing a clear overview and shape of any object, process or phenomenon. The software quality evaluation criteria and their weighting factors for each expert are offered, which provide a reliable presentation of the current state of the software development process, a proper understanding of the problems that may arise at any stage of the program project implementation, and the exact characteristics of their components. This mechanism of data reflection enables business analysts to submit qualitatively and quantitatively multiple values of complex software quality indicators that can be obtained from the results of any survey of different experts at a particular stage of the program project implementation. The proposed methodology is suitable for presenting a variety of expert survey results, with a subdivision into unlimited number of the roles of the software quality evaluation participants concerning the importance of each of them. An algorithm for calculating the area of sectorial petals in a polar coordinate system has been developed, using which it is possible to calculate and evaluate the relative software quality according to the proper criteria. The final complex software quality indicators for each expert are determined, and a comprehensive indicator of its quality is summarized as well. An algorithm for calculating the area of a petal diagram in a polar coordinate system has been developed, which allows determining the part of software quality according to all the criteria, which is currently evaluated by one of the experts, as well as the part of software quality, which has yet to be achieved for its 100% completeness. The relevant conclusions are drawn and recommendations are made concerning the use of the developed data reflection methodology.

Ключові слова: software design; data reflection; expert-based values; quality software; software quality evaluation criteria; polar coordinate system; petal chart.

Introduction. The quality of software is the main characteristic of its effective use (Pleskach & Zatonatska, 2011), as it indicates the degree of compliance with the requirements (ISO 9001, 2008; Pomorova & Hovorushchenko, 2013). Software quality usually means a set of properties of a software product that characterize its ability to meet the customer's established or foreseeable needs, which he specified in the form of user requirements for the software at the initial stages of its development (Pomorova, & Govorushchenko, 2013). Software quality is evaluated using a quality model (ISO / IEC 9126-1, 2001). Herein, software quality means the actions that determine how it fits for its purpose. Such evaluation is of particular importance with the development and improvement of expert-based data processing technologies (Kuliamin & Petrenko, 2008; Morhun, 2011). All this has led to the need for developing methods and tools for evaluating various characteristics of complex software quality that would account for some of the uncertainty of the data and subjective assessments of experts (Botsula & Morhun, 2008; Voronin, Ziatdinov & Kulinskii, 2011).

Expert technologies are an integral part of the managers' decision making (DeMarco Tom, 2002) during the software

development, or requirements change control and implementation risks as well as the quality control (Botsula & Morgun, 2014; Paulk & Curtis, 2001). Decision making by professional experts is based on authentic describing of the current situation, correct understanding of problem roots and its details. Each expert, who participates in the software quality evaluation should have domain knowledge, background and practical skills. Modern data analysis tools can help the project manager to formalize quality and quantity expert's assessments and make a complex evaluation of software quality by different attributes (Pleskach & Zatonatskaya, 2011).

Reflection term (Bederson & Shneiderman, 2003; Card, Mackinlay & Shneiderman, 1999; Spence, 2007) originates from Latin *visualis* – visual perception, representation, possibilities for visual observation. In general, reflection is representation of graphical patterns of data which help analytic to find anomalies, structures and relations (Kerren et al., 2008). In computer vision reflection is process of image obtaining by its computer model (Mazza, 2009). Software quality can be evaluated with quality model (ISO/IEC 9126-1, 2001; ISO 9001, 2008) which connected to expert data processing technology (Kulyamin & Petrenko, 2008;

Інформація про авторів:

Грицюк Юрій Іванович, д-р техн. наук, професор, кафедра програмного забезпечення. Email: yurii.i.hrytsiuk@lpnu.ua;

<https://orcid.org/0000-0001-8183-3466>; ResearcherID: V-3995-2017

Фернеза Остап Романович, аспірант, кафедра програмного забезпечення. Email: ostapfer@gmail.com

Цитування за ДСТУ: Грицюк Ю. І., Фернеза О. Р. Reflection of expert-based evaluation of software quality. Науковий вісник НЛТУ України. 2019, т. 29, № 8. С. 152–158.

Citation APA: Hrytsiuk, Yu. I., & Ferneza, O. R. (2019). Reflection of expert-based evaluation of software quality. *Scientific Bulletin of UNFU*, 29(8), 152–158. <https://doi.org/10.36930/40290828>

Morgun, 2011). The development of methods and tools for complex assessment of software quality attributes, which can take into account uncertainties of input information about process state and experts assessment subjectivisms is useful for project managers (Botsula & Morgun, 2011; Voronin, Ziatdinov & Kulinsky, 2011).

By reflection the results of an expert appraisal of the software quality, we will understand the presenting of information in graphical form for maximum convenience of its understanding and quick perception. However, many theorists and practitioners, in the field of information technologies, assume that such understanding of information reflection promotes the minimal intellectual and cognitive activity of the analyst, and visual instrumental tools perform illustrative function only (Botsula & Morgun, 2011; Voronin, Ziatdinov & Kulinsky, 2011). We will try to refute such false thoughts and demonstrate the considerable benefit of information reflection in the field of software development and its quality assessment.

Analysis of recent research and publications. Expert methods of products quality assessing have shown themselves well in various production spheres (Botsula & Morgun, 2011; Pleskach & Zatonatskaya, 2011). So many theoreticians and practitioners in the field of information technology have tried to transfer this experience to the process of software developing and its quality managing (Botsula & Morgun, 2014; Pomorova & Govoruschenko, 2013). In general, the use of expert technology allows analysts to determine the quality of both the future and already developed software and complexity of the software project implementation (DeMarco Tom, 2002). Expert assessments can serve only as a basis for determining the complexity of software project implementing or the trends of software quality attributes, and therefore have a recommendatory nature. Only evaluations by experts of the finalized software product in real exploitation conditions can provide a reliable and final assessment of the software quality (Hrytsiuk & Buchkovska, 2017a; Hrytsiuk & Dalyavskyy, 2018).

The different aspects of the software quality assessment models are considered in (Botsula & Morgun, 2011; ISO/IEC 9126-1, 2001; Nazemi, 2014). The problem of applying methods and tools for peer evaluation of software quality is widely covered in the study (Pleskach & Zatonatskaya, 2011). In various scientific papers, the process of expert assessment of the software quality is described taking into account the competence of experts and with fixing the values of each of them according to the defined criteria (Botsula & Morgun, 2011; Nazemi, 2014).

However, the procedure for assessing the software quality and the existing methods and tools of quality assurance, as well as the software developing process itself, remain unshakable as a fundamental theory and effective methodology. Most studies on software quality assessment, especially in the early stages of software project implementing, are chaotic and non-systematized. At the same time, as proved in papers (Pomorova & Govoruschenko, 2013a; Pomorova & Hovorushchenko, 2013b), at the end of the design stage of the software architecture it is possible and necessary to identify and eliminate up to 55% of all issues of the future software product. Of course, there are many fundamental software engineering researches (Boehm's, Dijkstra's, Meyer's papers), but there is no completed, tested and verified theory and methodology for the develop-

ment of complex and, at the same time, high-quality software, as well as methods and tools for evaluating and predicting its quality at an early stage of software project implementation. Therefore, the theory and practice of software products assessment require significant changes to prevent unforeseen losses both to software customers and developers, as well as incidents during its operation, caused by defects.

We will try to make our contribution to expert methods of software quality evaluating, especially in expert information visualizing. It seems an actual study concerning the development of an adequate methodology for visual presentation of experts' assessments in the form of radar charts, the justification of the criteria for the software quality assessment and their value for each of the experts, as well as the identification of integrated indicators of software quality.

The object of research is the results reflection of the expert software quality assessment. *The subject of the research* is the methods and tools of reflection the data obtained as a result of the processing of software quality expert evaluations according to various criteria, in a form that is convenient for clear perception and effective analysis. *The purpose of the study* is to develop a technique for reflection of information obtained as a result of processing expert estimates of the software quality according to different criteria using polar diagrams.

The purpose of the study is to develop methods of reflection data obtained as a result of processing quality assurance expert assessments by various criteria using polar diagrams. To accomplish this goal, you need to complete the following basic tasks:

- 1) justify the submission of expert evaluations as polar diagrams to help you put the image succinctly that text equivalents will take few paragraphs;
- 2) propose criteria for evaluating software quality and their weights for each of the experts who will provide fair presentation of the current situation, the proper understanding of the problems and specific characteristics of its components;
- 3) determine the final integrated software quality criteria for both each particular expert and generalized expert that will enable business analysts to calculate and evaluate the relative quality of developed software;
- 4) make appropriate conclusions and provide recommendations on the use of techniques developed for data reflection.

The expert assessments in radar chart representation. The results of many studies (Bederson & Shneiderman, 2003; Card, Mackinlay & Shneiderman, 1999; Heer, Card & Landay, 2005) show that the productivity of the analyst who uses visual information grows to 17%. Many scholars (Spence, 2007; Ware, 2000) believe that due to the information reflection, a person can remember such details, which in the text would not attract the attention of even a meticulous reader who is carefully reading and analyzing it. If the information (for example, software business requirements) comes from the customer of the software to the analyst not just in the form of a text, but with the corresponding drawings, diagrams and other visual objects, then this information is undoubtedly perceived by the analyst much faster and significantly easier.

In recent years, tremendous changes are happening in the information technology field – not only the volume of new information has increased, but also its quality has changed: despite the qualitative new information, so-called

informational garbage has appeared. There were also new types of visual information and forms of presentation. Information reflection refers to the interactive study of the graphical representation of abstract data for the enhancement of human knowledge (Mazza, 2009). Abstract data can contain both numeric and textual information as well as its graphical presentation. At the same time, data reflection is a graphical presentation of information, which allows to laconically show in the image while a text equivalent will take several paragraphs. In this study, as in previous papers (Hrytsiuk & Buchkovska, 2017a; Hrytsiuk & Dalyavskyy, 2018), we will try to develop a method for visualizing the results of expert appraisal of software quality according to different criteria, which will take into account the different importance of each criteria and the weight of the experts, which will allow a comprehensive evaluation of software quality, as well as existing already and in development stage.

To visualize the results of expert surveys, try to use polar diagrams based on some criteria for software quality assessment and obtaining its complex indicator. Usually, under a polar diagram, a graphical way of displaying abstract data in the form of a circle, divided into three or more sectors by the corresponding vectors (variables) is understood. These variables reflect the axes of the polar coordinate system with a common origin. Beginning of the countdown and the angle of inclination of vectors, as a rule, in the polar diagram indicate what is useful for both quantitative and qualitative information reflection. In different scientific literature (Kerren et al., 2008; Mazza, 2009; Spence, 2007) can be found on the following names of the polar diagram: web diagram, spider diagram, star sky map, star chart, irregular polygon and petal diagram.

The criteria for software quality assessment will be presented in the form of vectors (variables) in a polar coordinate system that divides the circle into sectors that generally form a polar diagram (Fig. 1,a). Each vector has indicators such as length and angle of inclination to the previous vector. Assumed that the length of the vector corresponds to the quantitative index of software quality according to the defined criteria. As an exception, the maximum length of any of the vectors must correspond to one hundred percent software quality by the defined criteria. Usually, the real length of each vector is only a certain part of its maximum length, which corresponds to the actual software quality by the defined criteria.

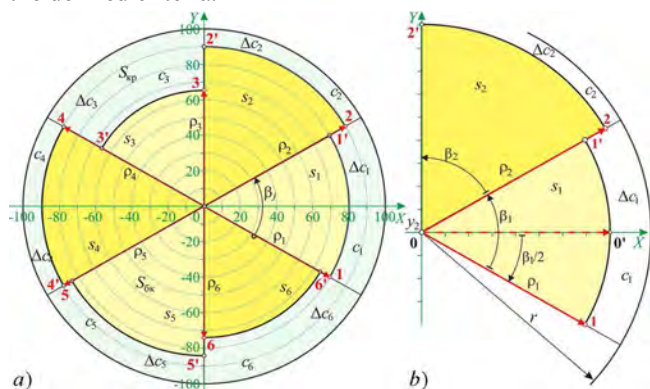


Fig. 1. Displaying of criteria for software quality assessment in the polar coordinate system

The authors (Botsula & Morgun, 2011) argue that the angle β between the vectors forms a sector whose value characterizes the effect of the corresponding criterion on the overall result of the software quality assessment. If all

the criteria have the same effect on the software quality, then the vectors of the corresponding criteria will be evenly distributed around the circle of the polar coordinate system. For example, for six criteria, this angle between all vectors will be $\beta = 2\pi / 6$. In the case of unequal impact of the criteria on the software quality, the angles between the corresponding vectors are determined by the formula

$$\tilde{B} = \left\{ \beta_j = 2\pi \cdot w_j / \sum_{i=1}^N w_i, j = \overline{1, N} \right\} \quad (1)$$

where: $\tilde{W} = \{w_j, j = \overline{1, N}\}$ – weight index of j -th software quality assessment criteria; N – amount of software quality assessment criteria. If the criterion vectors (ρ_1, \dots, ρ_6) defined in a polar coordinate system and, through each point of their vertices, in each sector, hold an arc with ρ_j radius, then we will have the so-called petal diagram **1-1', 2-2', ..., 6-6'** (Fig. 1,a), and the resulting area of the figure (S_{pd}) quantitatively characterizes the software quality according to all the criteria. The areas of sectoral petals (s_1, \dots, s_6), bounded by the polar sectors (c_1, \dots, c_6) with the angle β_j between the vectors, will quantify the software quality according to the defined assessment criteria.

The shape of the petal diagram gives the qualitative characteristics of the software by all criteria at once, and the shape of the sectoral petal (for example, **0,1,1', 0,2,2'** etc.) – according to the defined criterion. If the area of the petal diagram (S_{pd}) will be divided on the circle area (S_c) in which it is located, then we get the share of software quality at the moment, according to some expert estimates. The blank space of the circle ($\Delta S_c = S_c - S_{pd}$) is the part of the software quality that still needs to be achieved for hundred percent complete. If the sectoral petal area (s_j) will be divided on the sector area (c_j) in which it is located, then we get the share of the software quality for the j -th criterion at the moment, according to some expert estimates. The blank area of the circle sector ($\Delta c_j = s_j - c_j$) is the part of the software quality that still needs to be achieved by the appropriate criterion. Please note that the circle radius (r) must correspond to one hundred percent software quality by each evaluation criterion.

The above approach to the definition of software quality complex index and its further analysis is legitimate under certain conditions: 1) the vector-criteria should be at least three; 2) the initial vector-criterion **00'** (Fig. 1,b) must be on the positive ordinate axis of polar coordinate system counterclockwise shifted by $\beta_1/2$ angle. To find the (s_1, \dots, s_6) areas of sector petals (see Fig. 1,b), the following formula through the angle (β_j) between its radii (ρ_j) will be used:

$$s_j = \pi \rho_j^2 \beta_j, j = \overline{1, n} \quad (2)$$

Accordingly, the circle sectors (c_1, \dots, c_6) areas, components of which are sectoral petals, by (β_j) angle between its radii (r) are determined by the formula

$$c_j = \pi r^2 \beta_j, j = \overline{1, n} \quad (3)$$

Consequently, formula (2) gives the opportunity to calculate the sectoral petals areas, by which software quality calculate and estimate according to the relevant criteria. Also, these areas allow to determine part of the software quality according to a certain criterion that we currently have for one expert, as well as the part of the software quality that still needs to be achieved for one hundred percent complete.

Criteria for software quality assessment and their weighting indexes, which are provided to each expert. As noted above, the lengths of the vectors in polar coordinate system should be in proportion to the corresponding criteria value of software quality assessment, which are determined by the respondents' assessments and the role of each of them (Botsula & Morgun, 2011; Mazza, 2009). Usually, software respondents are participants of quality assessment process, which can act in two roles – as expert

and as direct user. The difference in roles is that the software quality assessment provided by an expert should have higher value in the process than the assessment provided by the user, because their qualifications are different. To avoid further confusion, all respondents will be called experts. Each expert will have defined weighting factors for each software quality assessment criteria (Table 1), those values will indicate their competence in a particular subject area (Yakovyna et al., 2010).

Table 1. Criteria for software quality evaluating and their weighting coefficients

| Number | Criteria for evaluation of software quality | Experts: | | | | Average weight coefficients |
|--------|---|---|-------------------|-------------------|-------------------|-----------------------------|
| | | application areas | usability | programming | generalized users | |
| | | The weights / Expert evaluation, points | | | | |
| 1 | Precision control and computing | $\frac{8}{10}$ | $\frac{5}{9}$ | $\frac{9}{10}$ | $\frac{7}{8.05}$ | 7.25 |
| 2 | Degree of standard interfaces | $\frac{7}{9}$ | $\frac{9}{8}$ | $\frac{6}{8}$ | $\frac{5}{7.50}$ | 6.25 |
| 3 | Functional capabilities completeness Software | $\frac{10}{9}$ | $\frac{6}{7}$ | $\frac{9}{9}$ | $\frac{6}{6,10}$ | 7.75 |
| 4 | Resistance to user errors | $\frac{6}{6}$ | $\frac{5}{5}$ | $\frac{10}{8}$ | $\frac{7}{7.70}$ | 7.00 |
| 5 | Expandable features | $\frac{5}{7}$ | $\frac{5}{5}$ | $\frac{10}{8}$ | $\frac{4}{6.05}$ | 6.00 |
| 6 | Ease of execution of tasks | $\frac{9}{9}$ | $\frac{9}{7}$ | $\frac{7}{7}$ | $\frac{10}{7.85}$ | 8.75 |
| 7 | Easy maintenance software | $\frac{9}{10}$ | $\frac{7}{9}$ | $\frac{6}{10}$ | $\frac{10}{7.35}$ | 8.00 |
| 8 | Compliance with applicable standards | $\frac{6}{6}$ | $\frac{5}{8}$ | $\frac{10}{7}$ | $\frac{5}{5.55}$ | 6.50 |
| 9 | Tolerability between software and hardware | $\frac{8}{9}$ | $\frac{6}{7}$ | $\frac{9}{6}$ | $\frac{6}{7.85}$ | 7.25 |
| 10 | Ease user training | $\frac{7}{6}$ | $\frac{8}{5}$ | $\frac{6}{9}$ | $\frac{10}{4.30}$ | 7.75 |
| | Total / Average score | $\frac{75}{8.10}$ | $\frac{67}{7.00}$ | $\frac{82}{8.20}$ | $\frac{71}{6.83}$ | 7.25 |

Typically, software quality assessments interactively collected from each expert are stored in a repository accessed through the appropriate software tool (Table 2 and 3). Experts' evaluations should be conducted in the form of a survey using a ranged scale for each criterion (Voronin, Ziatdinov & Kulinsky, 2011). Experts should make appropriate assessments, each of which is considered due to the relevant weighting factors (Table 1). Of course, each of the criteria will have a different impact on the complex index of software quality, the value of which is subsequently determined for each expert. Depending on the expert qualifications, each of them will also have different weighting factors values (Table 2).

Table 2. Roles of experts and their importance factors

| Roles of experts | Factors of importance: | |
|------------------------|------------------------|----------|
| | absolute | relative |
| Expert the application | 7 | 0.70 |
| Expert ease of use | 8 | 0.80 |
| Expert programming | 9 | 0.90 |
| Expert – average users | 5 | 0.50 |
| Total score | 29 | 2.90 |

Let's introduce a set of weight coefficients for each of the software quality evaluating criteria provided to each expert:

$$\tilde{W} = \{ \tilde{W}_i = \{ w_{i,k} = [0(1)10], k = \overline{1, K}, i = \overline{1, M} \} \}, \quad (4)$$

where: $w_{i,k}$ – weight coefficient of i -th criterion for software quality evaluating given to k -th expert; $0(1)10$ – range of expert assessment values from 0 to 10 with step 1; K – number of experts; M – number of software quality evaluation

criteria. Average value of weight coefficients in Table 1 for the i -th criterion for the software quality evaluation is determined by the following formula

$$\tilde{W}^c = \left\{ w_i^c = \frac{1}{K} \sum_{k=1}^K w_{i,k}, i = \overline{1, M} \right\}. \quad (5)$$

For each individual expert involved in the procedure of software quality assessments, the database should store the set of evaluations they have put (see Table 1). Also, in this database, there are signs of expert's role (Botsula & Morgun, 2011; Paulk & Curtis, 2001) and their importance coefficients (Table 2). The values of weighting factors are expressed both in absolute units and in relative terms. These values should be used to resolve generalized software quality indicators that will be addressed separately by static (professional) and dynamic (potential users) experts. The initial values of the weighting factors of experts are usually taken empirically, based on their importance at the initial stage of software development. In fact, these values need to be determined through appropriate tests on a 100-point scale, the results of which indicate the actual values of the experts weighting factors.

Let's introduce a set of weighting factors for each of the experts during software quality assessments:

$$\tilde{Q} = \{ q_k = [0(0,10)1], k = \overline{1, K} \}, \quad (6)$$

where q_k – weighting factor of k -th expert during software quality assessments. Consequently, the criteria for software quality assessments and their weighting factors for each expert are provided, it should provide a reliable representation of the existing state of the software development process, a

correct understanding of the problems root that may arise at any stage of the software project implementation and the exact characteristics of their components. The expert's roles to evaluate the software quality according to the relevant criteria is given, and the estimated coefficients of their importance are proposed.

Defining of software quality complex indicators. To determine of the software quality complex indicators, a set of assessments that are provided by relevant experts – participants of its quality evaluating process will be used (Table 2 and 3). The task of determining the total complex indicators of software quality for each expert and the integrated index of software quality for all experts in general appears (Botsula & Morgun, 2014; Morgun, 2011). Software quality assessments set that an expert defines according to a certain evaluation criterion, can be introduced:

$$\tilde{U} = \{u_i = [1(1)10], i = \overline{1, M}\}, \quad (7)$$

where u_i – software quality assessment, defined by expert for i -th evaluation criteria. Each separate software quality evaluation by the appropriate criterion provided by any expert belongs to this set:

$$\tilde{X} = \{\tilde{X}_i = \{x_{i,k} \in u_i, k = \overline{1, K}\}, i = \overline{1, M}\}, \quad (8)$$

where $x_{i,k}$ – software quality assessment for i -th evaluation criteria, defined by k -th expert at any moment of software project realization. For each expert can be introduced a complex indicator of software quality by the appropriate evaluation criterion, which can be calculated using the following formula:

$$\tilde{G} = \{\tilde{G}_i = \{g_{i,k} = x_{i,k} \cdot w_{i,k} \cdot q_k, k = \overline{1, K}\}, i = \overline{1, M}\}, \quad (9)$$

where $g_{i,k}$ – complex indicator of software quality by the i -th evaluation criterion, which relates to the k -th expert. For an integrated expert, so-called complex index of software quality according to the defined evaluation criterion is calculated using the following formula

$$\tilde{G} = \{\tilde{G}_i = \{g_{i,K+1} = \sum_{j=1}^K x_{i,j} \cdot w_{i,j} \cdot q_j / \sum_{j=1}^K q_j, i = \overline{1, M}\}\}, \quad (10)$$

where $g_{i,K+1}$ – complex indicator of software quality by the i -th evaluation criterion, which relates to the $(k+1)$ -th expert.

Table 3. Calculation results of software quality complex indicators and its averaged values

| Number | Criteria for evaluating software quality | Expert evaluation: | | | Generalized users | Averaged grades | |
|--------|---|--------------------|-----------|----------|-------------------|--------------------|-------------|
| | | areas | usability | programm | | $\sum w_{i,k} q_k$ | \bar{x}_i |
| | Weighting coefficients of experts | 0,70 | 0,80 | 0,90 | 0,50 | 0,725 | |
| 1 | Precision control and computing | 80 | 45 | 90 | 56,35 | 69,37 | 9,57 |
| 2 | Degree of standard interfaces | 63 | 72 | 48 | 37,50 | 56,43 | 8,36 |
| 3 | Functional capabilities completeness Software | 90 | 42 | 81 | 36,60 | 64,76 | 8,36 |
| 4 | Resistance to user errors | 36 | 20 | 80 | 53,90 | 48,33 | 7,16 |
| 5 | Expandable features | 35 | 20 | 80 | 24,20 | 42,97 | 7,47 |
| 6 | Ease of execution of tasks | 81 | 63 | 49 | 78,50 | 65,67 | 7,51 |
| 7 | Easy maintenance software | 90 | 63 | 60 | 73,50 | 70,40 | 8,80 |
| 8 | Compliance with applicable standards | 36 | 40 | 70 | 27,75 | 46,23 | 7,11 |
| 9 | Tolerability between software and hardware | 72 | 42 | 54 | 47,10 | 53,04 | 7,43 |
| 10 | Ease user training | 42 | 40 | 54 | 43,00 | 45,34 | 5,85 |
| | Total evaluations | 8,33 | 7,10 | 8,12 | 6,83 | 7,77 | 7,76 |
| | Estimates taking into account the weight of exp | 5,83 | 5,68 | 7,31 | 3,42 | 7,60 | 7,67 |

If take into account, that experts are doing software quality evaluation ($x_{i,k}$), during the survey on a 10-point scale (Tables 2 and 3), the weighting factor of the assessment criterion $w_{i,k}$ is also determined by the 10-point scale, and the q_k factor of the expert – dimensionless value from 0 to 1, then the complex index of software quality ($g_{i,k}$) will have value from 0 to 100. To calculate the integrated complex in-

dicator of software quality for each expert the following formula will be used:

$$\tilde{D} = \left\{ d_k = q_k \cdot \sum_{i=1}^M x_{i,k} \cdot w_{i,k} / \sum_{i=1}^M w_{i,k}, k = \overline{1, K} \right\} \quad (11)$$

and integrated index of software quality for all experts in total will be calculated using the following formula

$$d^y = \sum_{k=1}^K d_k / \sum_{k=1}^K q_k. \quad (12)$$

Calculation results of the software quality complex indicators and their averaged values, taking into account its evaluation criteria weighting factors, as well as the rank of each expert are shown in Table 3.

Algorithm for calculating the petal diagram area. Complex indicators of the software quality will be presented in the form of polar coordinate system vectors, which should form a petal diagram for each expert in particular and integrated expert in total. Each such vector is characterized by the length and angle to the preceding vector. As noted above, the length of the vector in any case should correspond to the quantitative value of the software quality complex index by the appropriate criterion. Point attention, that the petal diagram area quantifies the software quality according to all criteria at the same time, and the shape of the diagram gives a qualitative characteristic of the software. To find the coordinates of the petal diagram vertices **1-1', 2-2', ..., 6-6'** (see Fig. 1, a) the following calculation algorithm is used. In the case of unequal influence of the criteria on software quality (see formula (1)), the angles between the corresponding vectors taking into account (4) are determined by the following formula

$$\tilde{B}_k = \left\{ \beta_{i,k} = 2\pi \cdot w_{i,k} / \sum_{j=1}^M w_{j,k}, i = \overline{1, M}, k = \overline{1, K} \right\}, \quad (13)$$

and for average value of the software quality assessment (i.e., $k = K + 1$), with taking into account (5), this formula

$$\tilde{B}_k = \left\{ \beta_{i,k} = 2\pi \cdot w_{i,k}^c / \sum_{j=1}^M w_{j,k}^c, i = \overline{1, M}, k = K + 1 \right\}. \quad (14)$$

Since the polar sector with the β_j angle must start with the criterion vector (see formula (1)), then the first vector-criterion must be on the ordinate axis in the polar coordinate system, but counterclockwise shifted on $\beta_1/2$ angle. Therefore, starting the reference $\beta_{1,k} (\forall k \in K+1)$ angle, which corresponds to the 1-st polar sector, starts with the $\alpha_{1,k} = -\beta_{1,k}/2 (\forall k \in K+1)$ angle, and all others angles are calculated by the following formula

$$\tilde{A}_k = \left\{ \alpha_{1,k} = -\beta_{1,k} / 2; \alpha_{i,k} = \alpha_{i-1,k} + \beta_{i,k}, i = \overline{2, M}, k \in K + 1 \right\}. \quad (15)$$

To construct the petal diagram, it is necessary to draw the arcs at the corresponding angle $\alpha_{i,k} (\forall i \in M, \forall k \in K+1)$ from the ends of each vector. With the values of the lengths of the criterion vectors obtained by the formula (9) or (10), as well as the angles between them, obtained by the formula (15), it is possible to construct petal diagrams for any expert in particular, and for the integrated $(K + 1)$ expert in total (Fig. 2).

As noted above, the form of a petal diagram, constructed by the vertices of the vector-criteria, for any expert gives a qualitative characteristic of the software according to the selected evaluation criteria. At the same time, the resulting area of the petal will quantitatively characterize the software quality by all criteria. To find the area of the petal diagram,

using the vertices coordinates of its vector-criteria, the following formula can be used:

$$S_k^{na} = \pi \sum_{i=1}^M g_{i,k}^2 \cdot \beta_{i,k}, k \in K + 1. \quad (16)$$

To determine the share of software quality value, according to estimates of defined expert, its need to divide the petal diagram area on the circle area in which it is located:

$$z_k = \frac{S_k^{na}}{\pi r^2}, k \in K + 1, \quad (17)$$

where: z_k – the share of software quality value, according to estimates of k -th expert; r is the circle radius. As mentioned above, the software quality complex indicator ($g_{i,k}$) will have a 100 maximum value, that is, the circle radius will be 100 units. The circle part that is not filled – that part of the software quality, which still needs to be achieved for one hundred percent completeness.

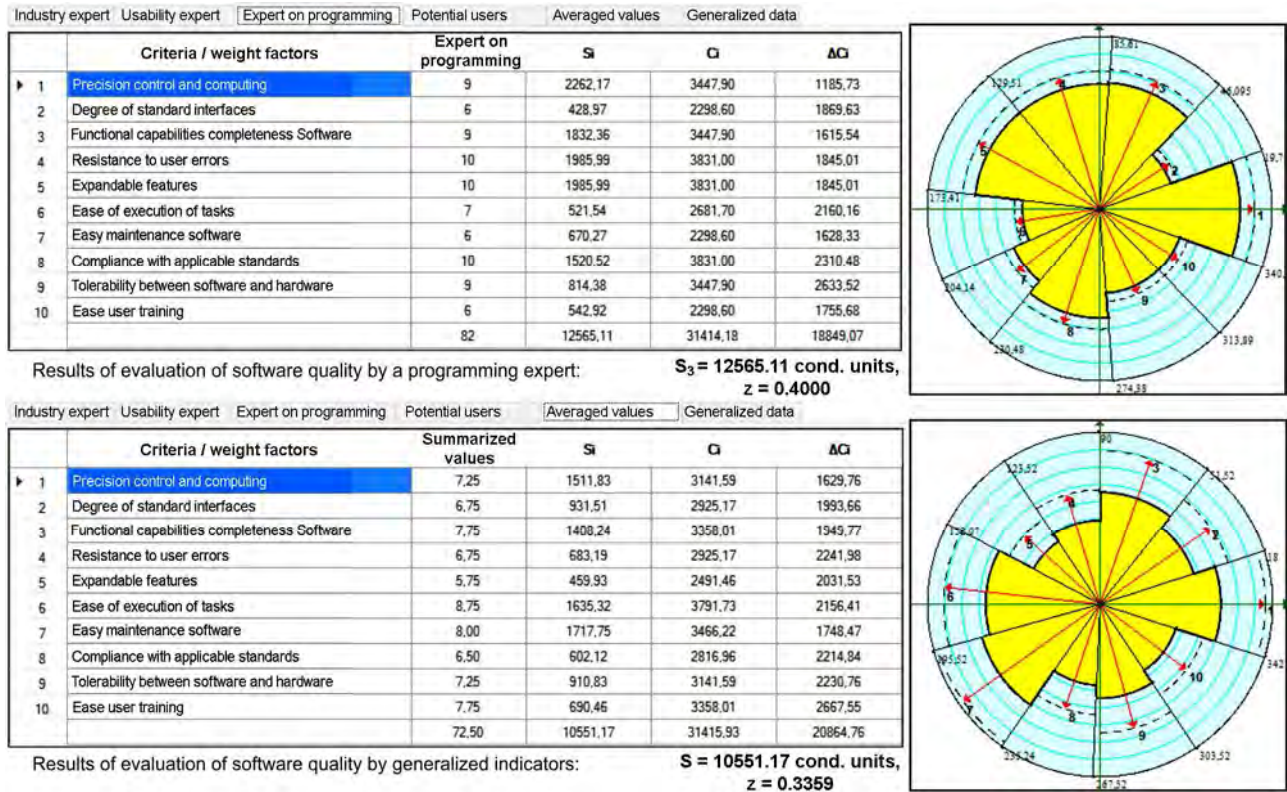


Fig. 2. Results of software quality evaluation in the form of petal diagrams for corresponding experts

Therefore, a technique for visualizing the results of expert software quality evaluation is developed, result of the evaluation is the set of petal diagrams constructed in the polar coordinate system according to individual experts' estimates, taking into account the rank of each evaluation criteria and the experts rank. Such information reflection technique provides an opportunity for business analysts to qualitatively and quantitatively represent multiple values of software quality complex indicators, which can be obtained as the survey results of various experts at a certain stage of the software project implementation. The proposed methodology is suitable for representing a set of expert survey results with a division on an unlimited number of participants roles in the software quality assessment, taking into account the experts rank.

Conclusions. The technique of information reflection, which is obtained as a result of processing expert's software quality assessment according to different criteria using petal diagrams, is developed. According to the research results, the following main conclusions can be made.

1) The representation of experts' assessments in the form of polar diagrams, which allows to concisely indicate software quality assessment, is substantiated. The algorithm of sectoral petals areas calculating in a polar coordinate system is developed, with the help of which it is possible to

calculate and estimate the relative software quality according to the defined evaluation criteria.

2) The criteria for software quality assessment and their weighting factors for each expert are proposed, which provide a reliable representation of the existing state of the software development process, correct understanding problems roots that can arise at any stage of the software project implementation.

3) The software quality complex indicator of for each expert, and integrated quality complex indicator of all experts are determined. The algorithm for calculating the petal diagram area in the polar coordinate system is developed, which allows to define share of the software quality according to all the criteria by each expert, as well as share of the software quality indicator, which still needs to be achieved for one hundred percent completeness.

References

- Bederson, B., & Shneiderman, B. (2003). *The Craft of Information Visualization: Readings and Reflections*. Morgan Kaufmann. 410 p.
- Botsula, M. P., & Morgun, I. A. (2011). A method for obtaining a comprehensive evaluation of the quality of web materials using the polar coordinate system. *Bulletin of the Vinnitsa Polytechnic Institute, 1*, 84–88. Retrieved from: <https://visnyk.vntu.edu.ua/index.php/visnyk/article/view/1367/conferences.vntu.edu.ua>. [In Ukrainian].
- Botsula, M. P., & Morgun, I. A. (2014). New method and information technology for data processing for quality management of electronic training courses. *Information Technology and Computer Engi-*

- neering: *International Scientific and Technical Journal*, 3, 25–33. Retrieved from: http://nbuv.gov.ua/UJRN/Itki_2014_3_6. [In Ukrainian].
- Card, S. K., Mackinlay, J. D., & Shneiderman, B. (1999). *Readings in Information Visualization: Using Vision to Think*. Morgan Kaufmann Publishers, 686 p.
- DeMarco Tom. (2002). *Controlling Software Projects: Management, Measurement and Estimation*, 279 p.
- Heer, J., Card, S. K., & Landay, J. (2005). Prefuse: a toolkit for interactive information visualization, (10 p.). In: *ACM Human Factors in Computing Systems CHI 2005*, 280 p.
- Hrytsiuk, Yu. I., & Buchkovska, A. Yu. (2017a). Visualization of the Results of Expert Evaluation of Software Quality Using Polar Diagrams. *Scientific Bulletin of UNFU*, 27(4), 137–145. <https://doi.org/10.15421/40271025>
- Hrytsiuk, Yu. I., & Dalyavskyy, V. S. (2018). Using Petal Diagram for Visualizing the Results of Expert Evaluation of Software Quality. *Scientific Bulletin of UNFU*, 28(9), 95–104. <https://doi.org/10.15421/40280919>
- ISO 9001:2008. *Quality Management System – requirements*. Retrieved from: <https://www.iso.org/standard/46486.html>
- ISO/IEC 9126-1:2001. *Software Engineering – Product Quality. Part 1: Quality model*. Retrieved from: <https://www.iso.org/standard/22749.html>
- Kerren, A., Stasko, J. T., Fekete, J.-D., & North, C. (Eds.) (2008). Information Visualization, (pp. 1–18). In: *Human-Centered Issues and Perspectives*. Vol. 4950 of LNCS State-of-the-Art Survey. Springer-Verlag Berlin Heidelberg 2008.
- Kulyamin, V. V., & Petrenko, O. L. (2008). *Place of testing among software quality assessment methods*. Moscow: ISP RAS. Retrieved from: <http://software-testing.ru/library/5-testing/117-2008-10-13-19-25-13>. [In Russian].
- Mazza, R. (2009). *Introduction to Information Visualization, University of Lugano Switzerland*. Springer-Verlag London Limited 2009, 139 p. <https://doi.org/10.1007/978-1-84800-219-7>
- Morgun, I. A. (2011). Expert evaluation method of software quality. *Software Engineering: Materials of the International Scientific and Practical Conference of Postgraduate Students and Students*, 2(6), 33–37. Vinnytsia. Retrieved from: <http://jrn1.nau.edu.ua/index.php/IPZ/article/view/3086>. [In Ukrainian].
- Nazemi, K. (2014). *Adaptive Semantics Visualization. Dissertation zur Erlangung des akademischen Grades eines Doktor-Ingenieurs*. Eurographics Association for Computer Graphics. 360 p. Retrieved from: <http://diglib.eg.org/handle/10.2312/12076>
- Paulk, M. C., & Curtis, B. (2001). *The Capability Maturity Model: Guideline for Improving the Software Process*. Carnegie Mellon University. 434 p.
- Pleskach, V. L., & Zatonatskaya, T. G. (2011). *Information systems and technologies at enterprises: textbook*. Kyiv: Knowledge, 718 p. Retrieved from: http://pidruchniki.com/1194121347734/informatika/analiz_yakosti_pro_gramnogo_zabezpechennya#42. [In Ukrainian].
- Pomorova, O. V., & Govoruschenko, T. O. (2013). *Modern problems of software quality assessment*. Radioelectronic and Computer Systems, 5, 319–327. Kharkiv: NAU "KhAI". [In Ukrainian].
- Pomorova, O., & Hovorushchenko, T. (2013b). Intelligent Assessment and Prediction of Software Characteristics at the Design Stage. *American Journal of Software Engineering and Applications (AJSEA)*, 2(2), 25–31. Retrieved from: <http://article.sciencepub.lishinggroup.com/pdf/10.11648.jajsea.20130202.11.pdf>
- Spence, R. (2007). *Information Visualization: Design for Interaction* (2nd ed.). Prentice Hall, 304 p.
- Voronin, A. N., Ziatdinov, Yu. K., & Kulinsky, M. V. (2011). *Multi-criteria tasks: models and methods: monograph*. Kiev: NAU, 348 p. [In Russian].
- Ware, C. (2000). *Information Visualization: Perception for design* (2nd ed.). San Francisco, CA: Morgan Kaufmann Publishers, 34 p.
- Yakovyna, V. S., Seniv, M. M., Chabanyuk, Ya. M., Fedasyuk, D. V., & Khimka, U. T. (2010). The criterion of the sufficiency of the software testing process. *Bulletin of the National University "Lviv Polytechnic". Series: Computer Science and Information Technology*, 672, 346–358. Retrieved from: <http://ena.lp.edu.ua/handle/ntb/7964?mode=full>. [In Ukrainian].

Ю. І. Грицюк, О. Р. Фернеза

Національний університет "Львівська політехніка", м. Львів, Україна

ВІДОБРАЖЕННЯ ЕКСПЕРТНИХ ОЦІНОК ЯКОСТІ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Розроблено методику відображення даних, яку отримують внаслідок оброблення експертних оцінок якості програмного забезпечення (ПЗ) за різними критеріями його оцінювання з використанням пелюсткових діаграм. Встановлено, що під відображенням експертних оцінок якості ПЗ розуміють подання даних у графічному вигляді для максимальної зручності їх розуміння та швидкого сприйняття, а також надання оглядової та зрозумілої форми будь-якому об'єкту, процесу чи явищу. Запропоновано критерії оцінювання якості ПЗ та їхні вагові коефіцієнти для кожного з експертів, які забезпечують достовірне подання наявного стану процесу розроблення ПЗ, правильне розуміння суті проблем, що можуть виникнути на будь-якому етапі реалізації програмного проекту, і точні характеристики їх складових. Наведено ролі експертів, які мають оцінювати якість ПЗ за відповідними критеріями, і запропоновано орієнтовні коефіцієнти їхньої вагомості. Такий механізм відображення даних надає можливість аналітикам якісно та кількісно подати множини значень комплексних показників якості ПЗ, які можна отримати за результатами будь-якого опитування різних експертів на певному етапі реалізації програмного проекту. Запропонована методика є придатною для подання множини результатів опитувань експертів з поділом на необмежену кількість ролей учасників оцінювання якості ПЗ з врахуванням вагомості кожного з них.

Розроблено алгоритм розрахунку площ секторних пелюстків у полярній системі координат, за допомогою яких можна обчислити і оцінити відносну якість ПЗ за відповідними критеріями. Визначено підсумкові комплексні показники якості ПЗ для кожного з експертів і узагальнений комплексний показник його якості. Розроблено алгоритм розрахунку площі пелюсткової діаграми у полярній системі координат, який дає змогу визначити ту частину якості ПЗ за всіма критеріями, яку маємо на даний момент за оцінками одного з експертів, а також ту частку якості ПЗ, яку ще потрібно досягти для стовідсоткової її повноти. Зроблено відповідні висновки та надано рекомендації щодо використання розробленої методики відображення даних.

Ключові слова: програмний проект; відображення даних; експертні оцінки; якість програмного забезпечення; показники якості програмного забезпечення; критерії оцінювання якості програмного забезпечення; полярна система координат; пелюсткові діаграми.