

Генерація бібліотеки, а не відразу повнофункціонального додатку дає можливість використовувати її в різних додатках та по-різному інтерпретувати отримані результати.

В даній роботі згенерована бібліотека використовується автоматизованою системою збору й обробки даних медичного призначення для надання рекомендацій щодо покращення стану здоров'я для користувачів, де розгортається у складі Spring Boot додатку для кращого масштабування та забезпечення більшої надійності системи в цілому. Також такий підхід знизить складність міграції автоматизованої системи на мікросервісну архітектуру.

В подальшому планується вдосконалення наявних й додавання нових можливостей предметно-орієнтованої мови, а саме: генерація окремого інтегрованого середовища розробки для медичних працівників, додавання нових концептів, розширена перевірка на типові помилки, інтелектуальне автодоповнення, таблична нотація та ін.

ДЖЕРЕЛА

1. Noncommunicable diseases country profiles 2018 [Електронний ресурс]. – 2018. – Режим доступу до ресурсу: <https://www.who.int/nmh/publications/ncd-profiles-2018/en/>.
2. Voelter M. DSL Engineering: Designing, Implementing and Using Domain-Specific Languages / Markus Voelter., 2013. – 558 с.
3. Spring Boot Reference Documentation [Електронний ресурс] / [P. Webb, D. Syer, J. Long та ін.]. – 2020. – Режим доступу до ресурсу: <https://docs.spring.io/spring-boot/docs/current/reference/htmlsingle/>.

ANALYTIC MARK OF THE RELATIVE IMPORTANCE OF INFORMATIVE TECHNOLOGY SOFTWARE CRITERIA

Bubela M., Moroz A.

Central Ukrainian National Technical University, Kropivnitskiy

The main providing of modern informational technologies is software. At the same time, the production of high-quality software is one of the most important tasks in science and production development. The viability of software as a final result depends on how successfully it was developed. Nowadays, the real problem of criteria importance evaluation of software quality exists. For example, most of the projects do not achieve success and do not fit the deadline or budget [1]. It can be a result of insufficient attention to the task of providing software quality during IT-projects realization. One of the ways out of this problem is software quality criteria ranking for their providing (realization) in order of decreasing relative importance. That is why the task of analytic marks of relative importance (relative) of IT software criteria.

In general, there are many different models of software quality evaluation. The international standard [2] defines 6 main software quality evaluation criteria: functionality, reliability, using convenience (practicality), productivity (efficiency), maintainability, portability. The analytic evaluation of these criteria is based on next. Functionality is one of the main software quality evaluation criteria because this criterion is the enumeration of characteristics which software must include. Reliability is the main software quality evaluation criteria because exactly this criterion regulates software work and the period of its exploitation. The using convenience of software is important if it is oriented on a usual user, who does not have technical knowledge for a full understanding of the software. In cases when this software is used by people with technical knowledge or a person does not take part in work with it, this criterion is not important at all to the authors' mind. Productivity is also an important software quality criteria because it shows the resources required for normal software functioning and defines the level of efficiency of its usage in certain situations. The maintainability is an important criterion because if during software using the mistakes would be found, they must be corrected in the shortest time, also in some time, the software will be needed to be modified for it to continue to be actual. Portability is an important criterion in the case when the software is oriented for a huge amount of users who have different operation system etc.

The synthesis of analytic marks will be made in two steps: 1) software will be used by users who understand its work or software works automatically; 2) software is used by usual users who do not understand its functioning.

So, if the software will be used by users who understand its work or software works automatically, reliability is the criteria number one, because without it the program will not work, that is why there is no sense to check other criteria if this criterion was not done during software realization. The second criterion must be functional because every software must have certain functional, without which there is no sense in the work of the program. The third criterion by importance is considered to be maintainability. Then practicality, which in contradistinction to first three criteria does not influence on the software functioning, then portability. The authors placed using convenience to the last place.

In the second case, if the software is used by usual users who do not understand its work, the using convenience takes the second place. Therefore, due to interface and functional incomprehensibility, the user will not be able to use the software. The rest of the analytic marks match with those, which were received after using the first step.

The conditional representation of analytic marks of IT software quality criteria importance is presented on the fig. 1.

In conclusion, the analytic marks of the relative importance of IT software criteria can be used during IT-projects realization. The perspective of further researches is revealing of dependences and evaluation of the influence of every software quality criteria on the number of mistakes in it.

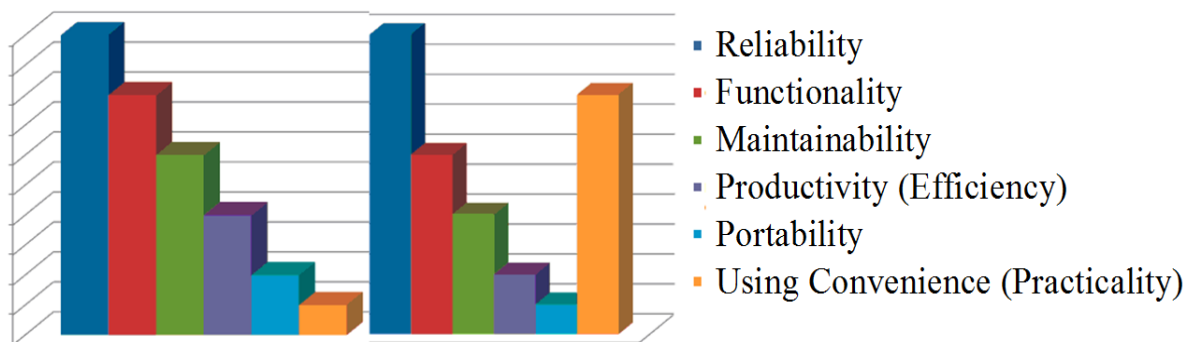


Fig. 1. The histogram of software quality criteria importance in different cases.

REFERENCES

1. Govoruschenko T.O. Analysis of the field of software quality assessment. Computer systems and networks. No. 773. 2012. pp. 41-48.
2. ISO/IEC 9126. Information Technology. Software Quality Characteristics and metrics. International Standard. 2011.
3. Babenko L.P., Lavrisheva K.M. Fundamentals of Software Engineering: Tutorial. K.: Knowledge, 2001. 269 p.
4. Lipaev V.V. Reliability of software tools. M.: Sinteg, 1998. – 231 p.
5. Glass G. A Guide to Reliable Programming. M.: Fin&Stat, 1982. 256 p.
6. Dorenskyi O. The Assessment Method of Test Suites for Testing of Information Security Systems Software. ITSEC : Information Technology Security : Internat. Scientific Conf., 2017, Kyiv. K.: NAU, 2017. P. 48-49.

ВЕБ-ДОДАТОК ДЛЯ МОНІТОРИНГУ І ЕФЕКТИВНОГО ТА ПРОЗОРОГО КЕРУВАННЯ ОСББ

Вільчинський Б.С.

Київський університет імені Бориса Грінченка, м. Київ

В Україні існують об'єднання співвласників багатоквартирного будинку (далі ОСББ), які з'явилися замість житлово-експлуатаційних контор і мають метою покращення обслуговування домівки жильців. ОСББ – це неприбуткова юридична особа, створена власниками квартир та/або нежитлових приміщень багатоквартирного будинку для спільного користування, утримання та управління своїм будинком та прибудинковою територією, а також для юридичного оформлення їхніх майнових прав на будинок та прибудинкову територію. Тобто, ОСББ може керуватись власниками квартир цього будинку, які можуть обрати собі головууючого.

При цьому багато проблем залишилося. Але у сучасному світі є відповідні інформаційні технології для підтримки керування різними об'єктами і їх треба кваліфіковано використовувати.

Таким чином є наступна потреба керування будинком. Треба створити інформаційний ресурс для автоматизованого комплексного керування