

УДК 338.46

DOI: 10.17586/2310-1172-2022-16-2-152-159

Научная статья

## Экономический эффект от внедрения процедуры оценки качества программного обеспечения на малых предприятиях ИТ-сферы

**Василенко Е.А.** elena.vasilenko2@outlook.com

*Канд. экон. наук* **Мишура Л.Г.** mishuralg@rambler.ru

Университет ИТМО

197101, Россия, г. Санкт-Петербург, Кронверкский проспект, д. 49

*Рост спроса на программное обеспечение и всё повышающиеся требования к их характеристикам вместе с возрастающей конкуренцией на рынке ИТ заставляют поднимать вопросы об оценке качества производимого программного продукта. Актуальность данной работы заключается в том, что в условиях активно развивающейся ИТ-отрасли компаниям данной сферы необходимо обеспечивать надлежащую оценку качества своих продуктов при невысоких затратах. С ростом конкуренции приходится оптимизировать затраты и искать более дешёвые и надёжные средства контроля качества. В работе предложена процедура оценки качества программного обеспечения на малых предприятиях ИТ-сферы. Данный сегмент рынка информационных технологий не имеет формализованной базы для проведения всесторонней оценки качества производимого программного продукта. Разработанная процедура включает в себя пять этапов: описание программного обеспечения (ПО), выбор ключевых критериев качества ПО и показателей качества исходного кода, выбор метрик, оценка качества по ключевым критериям и расчет показателей качества исходного кода, оценка качества ПО. Для наглядного представления важности написания качественного программного кода в тексте представлен график, иллюстрирующий зависимость суммарной функциональности от времени в случаях качественного и некачественного исходного кода. В работе предложены формулы расчета показателей качества исходного кода программного обеспечения, а также приведена шкала оценки качества продукта на основе полученного обобщенного показателя качества, оценен экономический эффект от внедрения процедуры оценки на малом предприятии ИТ-сферы. Результаты данного исследования позволяют сделать вывод о том, что внедрение процедуры оценки качества программного обеспечения не потребует дополнительных материальных затрат, что является крайне важным фактором для малых предприятий рассматриваемого сегмента рынка.*

*Ключевые слова:* качество, экономический эффект, программное обеспечение, критерии качества, метрики качества, оценка качества.

Scientific article

## Economic effect of software quality evaluation procedure in small IT businesses

**Vasilenko E.A.** elena.vasilenko2@outlook.com

*Ph.D.* **Mishura L.G.** mishuralg@rambler.ru

197101, Russia, St. Peterburg, Kronverksky pr., 49

*IT industry is growing fast from year to year, which leads to growing competition in the respective market. That is why quality evaluation best practices are becoming more widely appreciated. Relevance of the chosen theme results from the fact that appropriate quality evaluation process is required to make a competitive product in the IT industry. The intention of this paper is to suggest a software quality evaluation procedure for small IT businesses. This market segment does not have a formalized basis for comprehensive quality evaluation. The designed procedure includes five steps: describing a software product, selection of key quality measures of the software product and its source code, selection of appropriate methods for measuring, quality evaluation according to the key measures and calculation source code quality metrics, quality rate calculation. The importance of high-quality source code has been demonstrated by a chart which shows a dependency of total implemented functionality from time in two cases: high-quality source code and low-quality source code. As a conclusion, it is claimed that introduction of quality evaluation procedure does not lead to*

*material costs increase, which is highly important for small IT businesses. There is economic effect analysis of introduction of software quality evaluation procedure in small IT businesses.*

**Keywords:** quality, economic effect, software, quality measures, quality metrics, quality evaluation.

## Введение

Развитие индустрии информационных технологий или как иначе её принято называть сферы IT в настоящее время приобретает статус одной из самых динамично развивающихся отраслей. В экономическом плане IT-отрасль – это относительно молодое и перспективное направление, которое имеет огромный потенциал для развития и дальнейшего роста. Повсеместное распространение продуктов данной отрасли способно оказать влияние на процессы глобализации и развития общества в целом, именно поэтому значимость создаваемых IT-компаниями программных продуктов колоссальна. Управление качеством играет важную роль во всех отраслях, однако для IT-продуктов это понятие дополняется требованиями к сложным техническим характеристикам скорости, безопасности и надежности. Говоря о крупных передовых компаниях изучаемой сферы, стоит отметить, что процесс управления качеством включает в себя ряд сложных и дорогостоящих процедур автоматизированного тестирования, которые позволяют добиваться стабильно высокого качества продаваемых программных продуктов. Малые же компании и компании, которые только набирают обороты на рынке IT, вынуждены в наибольшей степени сокращать расходы на оценку качества, поскольку имеющихся ресурсов зачастую оказывается недостаточно для тщательного управления качеством производимого программного обеспечения.

Значимым фактом является то, что малые компании имеют ряд преимуществ на рынке IT, поскольку им свойственна высокая мобильность в выборе имеющихся практик и возможностей. Это свидетельствует о том, что крайне важно обеспечить всестороннюю поддержку развивающимся малым предприятиям, создав формализованную базу управления качеством их продуктов.

## Качество программного обеспечения

Качество программного обеспечения (Software Quality) – соответствие программного продукта тем требованиям, которые к нему предъявлены. Очевидно, что в общем случае, требования к ПО определяются особенностями той сферы, в которой в значительной степени будет использоваться программное обеспечение. [1]

Согласно известной модели качества программного обеспечения имеется четыре этапа её визуализации. [2]

На первом этапе происходит выделение всех важных критериев качества, которые имеют значения для конечного пользователя. По умолчанию среди них выделяют следующие: удобство, эффективность, надежность, переносимость, функциональность, сопровождаемость:

- Функциональность (Functionality) представляет собой набор тех возможностей, которые предоставляются пользователю программным продуктом.
- Удобство использования (Usability) – это способность продукта быть понятным для потребителя.
- Переносимость (Portability) – легкость переноса программного продукта в различные программные и аппаратные среды.
- Удобство сопровождения (Maintainability) предполагает простоту тестирования, легкость исправления дефектов для того, чтобы упростить процесс дальнейшего обслуживания данного продукта, а также подстроить его к тому окружению, в котором оно используется.
- Производительность (Performance) может выражаться как совокупность различных характеристик, например, время обработки запроса, количество обрабатываемых запросов в единицу времени, пропускная способность, скорость доставки пакета данных.
- Надежность (Reliability) наряду с производительностью является важнейшим условием достижения успеха продукта на рынке IT. В данном случае речь идет также и о корректном восстановлении после сбоя, то есть устойчивость продукта к возникающим сбоям.

На втором этапе для каждого обозначенного критерия определяется список атрибутов, которые детализируют характеристику первого этапа. Совокупность этих атрибутов и определяет качество программного продукта.

На следующем этапе (третий) для каждого обозначенного атрибута качества подбирается способ измерения и подходящая величина. Далее производится расчет.

Последний заключительный этап – ранжирование полученных показателей по степени важности. Полученный список используются для при разработке технического задания проекта или как шаблон для последующих аналогичных задач.

Несмотря на то, что в каждой компании могут самостоятельно и независимо разрабатываться собственные стандарты оценки качества ПО, в общем случае выделяют две большие категории качества: внешнее, которое определяется требованиями заказчика или пользователя, и внутреннее – то, что проявляется в процессе реализации программного продукта и никак не проверяется со стороны заказчика. [3] Таким образом:

- Внешнее качество ПО: пользовательский интерфейс (UI) и пользовательский опыт (UX) – насколько быстро и просто пользователь может решить свою задачу. [4]
- Внутреннее качество ПО: архитектура проекта – насколько исходный код программы структурирован, насколько просто и быстро программист может найти нужный ему в данный момент участок кода.

Для того, чтобы оценить соответствует ли поведение создаваемого программного обеспечения и его компонентов ожидаемому поведению, применяют такой процесс исследования как тестирование. [5]

Тестирование (Software Testing) – оценка корректности работы отдельных элементов программы; является неотъемлемой частью процесса разработки и широко применяется в программировании. Тестирование представляет собой отдельную ветвь профессиональной ориентации сотрудников IT-сферы. [6]

Существует большое количество классификационных признаков, по которым процесс тестирования можно разделить на отдельные категории. Среди них выделяют: по времени, по объекту, по внутреннему строению и т. д. На практике чаще всего используется вид тестирования по степени автоматизации, а именно ручное, машинное или смешанное тестирование. Ключевыми уровнями тестирования, применяемыми преимущественно в крупных компаниях с налаженной внутренней структурой управления качеством разработки, являются: тестирование отдельных компонентов, межсистемное тестирование, полностью системное, которое в свою очередь делится еще на два вида. [7] Информация о каждом уровне тестирования представлена в табл. 1.

Таблица 1

### Уровни тестирования

| Уровень тестирования | Объект тестирования  |
|----------------------|--|
| Компонентное         | Отдельный минимальный компонент системы/подсистемы   |
| Интеграционное       | Интерфейсы между отдельными частями системы/подсистемы с поэтапным подключением каждого из них |
| Системное            | Целостная система  |
| α-тестирование       | Целостная система в условиях имитации реальной работы, т. е. с разворачиванием окружения       |
| β-тестирование       | Предварительная версия, распространенная на небольшой круг реальных пользователей.             |

Существует еще ряд видов тестирования, объектами которых выступает не исходный код программы, а различная документация, сопровождающая проект. Для создания программного обеспечения (ПО) высокого качества важно верно оценить значение качества исходного кода, реализующего данный программный продукт. Большую часть времени программисты проводят за написанием и чтением программного кода, и его качество напрямую влияет на скорость обнаружения ошибок и их устранение. [8]

Для наглядного представления важности написания качественного программного кода, на рис. 1 представлен график, иллюстрирующий зависимость суммарной функциональности от времени в случаях качественного и некачественного исходного кода.

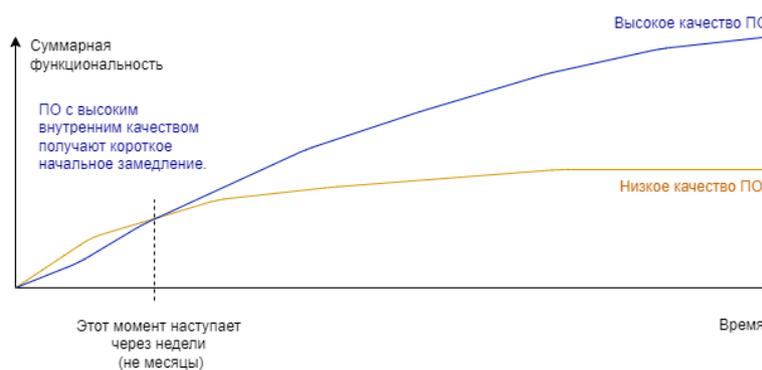


Рис. 1. Зависимость суммарной функциональности от времени в случаях качественного и некачественного исходного кода

Таким образом, среди преимуществ, которые предоставляет высокое качество программного кода, можно выделить следующие:

1. Легкость восприятия для новых участников проекта;
2. Высокая скорость последующей разработки (дополнения функционала);
3. Экономическая выгода (низкая стоимость последующей разработки за счет отсутствия необходимости тратить время на устранение технического долга – разницы между текущим состоянием программного кода и его идеальным состоянием).

Качество программного обеспечения складывается из соблюдения критериев качества, а также напрямую зависит от качества исходного кода программы.

### Процедура оценки качества программного обеспечения

Оценка качества программного обеспечения – сложный и дорогостоящий процесс. Однако крайне важно не пренебрегать этим этапом реализации проекта. Крупные компании на рынке ИТ как правило имеют достаточное количество ресурсов для полномасштабной организации процесса оценки программного обеспечения посредством создания целых подразделений и отделов контроля качества. Малые предприятия данного направления деятельности, как правило, реализуют оценку качества ПО лишь силами ручного тестирования, реже – автоматического, чего зачастую оказывается недостаточно для своевременного и полного выявления дефектов в работе программы и оценки качества готового продукта. Таким образом, крайне важно создать для данного сегмента рынка информационных технологий нормативную базу для комплексной оценки качества программного продукта. [9] На рис. 2 представлена процедура оценки качества программного обеспечения для малого предприятия ИТ-сферы.

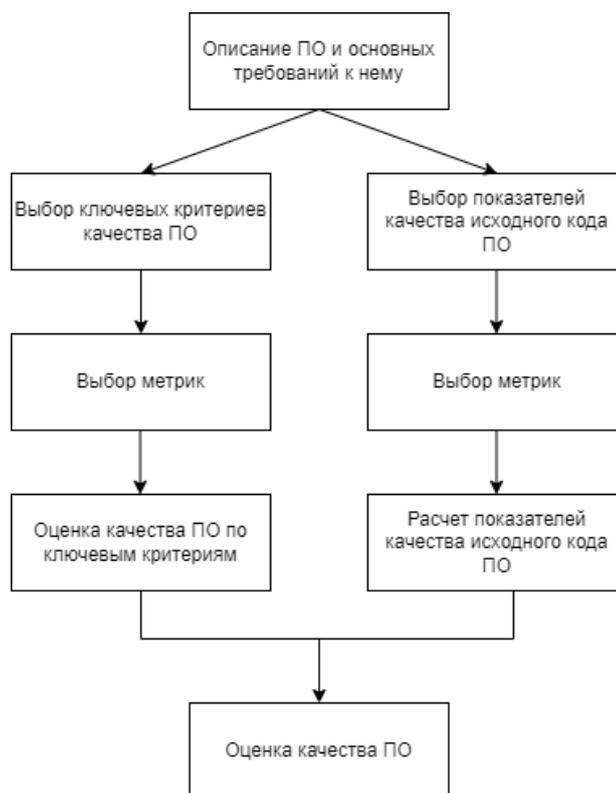


Рис. 2. Процедура оценки качества программного обеспечения

Первый этап в процедуре оценки качества подразумевает описание создаваемого программного обеспечения и формирование списка функциональных требований, которые к нему предъявляются.

Второй этап – это выбор тех критериев качества, которые в наибольшей степени способны всесторонне охарактеризовать степень соответствия требованиям к качеству продукта. На данном этапе так же осуществляется выбор показателей качества исходного кода программы. Далее в работе предложены четыре показателя,

оценивающих качество программного кода, однако их выбор может меняться по требованию технических специалистов. [10]

Говоря о внешнем качестве программного обеспечения, следует отметить, что все указанные критерии – удобство применения, функциональность, надежность, сопровождаемость, производительность и переносимость – не поддаются формализации, поскольку зависят от сложности проекта и его специфики. Измерить данные показатели качества не представляется возможным в отрыве от конкретного проекта.

Провести количественную оценку показателей качества можно применительно ко внутреннему качеству программного продукта, а именно оценив качество исходного программного кода ПО. [11] Оценка качества кода позволяет выявить ошибки на ранних стадиях разработки, что минимизирует риск возникновения серьезных проблем на более поздних стадиях, снижает затраты на разработку и упрощает дальнейшую работу. Кроме того, качество кода напрямую определяет качество конечного программного продукта. Таким образом, далее в работе рассматривается расчет по правой ветви процедуры. Ниже приведены основные показатели качества программного кода и формулы 1–4 для их вычисления, что в схеме процедуры оценки качества ПО соответствует этапам 3 и 4 (правая ветвь). [12]

### 1. Процент покрытия тестами (ППТ)

$$\frac{\text{Количество строк, покрытых тестами}}{\text{Общее количество строк}} \times 100\%$$

Данный показатель в процентном выражении характеризует степень покрытия исходного кода программы тестами, то есть описывает степень тестирования программы.

### 2. Частота возникновения ошибок (ЧВО)

$$\frac{\text{Количество выявленных ошибок}}{\text{Общее количество строк}} \times 100\%$$

Частота возникновения ошибок показывает соотношение ошибок и строк кода в проекте или его части.

### 3. Частота несоответствий стандартам написания кода (ССК)

$$\frac{\text{Число отклонений от стандарта}}{\text{Общее количество строк}} \times 100\%$$

Показатель вычисляет частоту возникновения отклонений от принятого стандарта, выявленных средствами статического анализа кода, то есть без запуска программы.

### 4. Удобство обслуживания (УО)

$$171 - 5.2 \times \ln(\text{показатель Холстеда}) \times 0.23(\text{Сложность}) - 16.2 \times \ln(\text{Количество строк кода})$$

Удобство обслуживания – сложный показатель, рассчитываемый на основе цикломатической сложности программы и показателя Холстеда, служит для вычисления простоты сопровождения кода (исправления ошибок, добавления функционала и т. д.) [13]

Цикломатическая сложность – описание структурной сложности программного обеспечения, которая вычисляется путем построения графа потока управления программы.

Показатель Холстеда – показатель, который так же используется для оценки сложности программного обеспечения на основе синтаксических элементов кода.

Для удобства вычисления сложных показателей Холстеда и цикломатической сложности программы применяют разработанные сервисы математической направленности.

Для оценки качества программного обеспечения и перехода к пятому этапу процедуры оценки качества программного обеспечения необходимо получить обобщенный показатель, который рассчитывается по формуле 5.

## 5. Качество программного обеспечения (КПО)

$$\sum Ki \times ai,$$

где  $Ki$  – это показатель качества программного продукта,  $ai$  – весовой коэффициент.

В данном случае общий показатель качества программного обеспечения выглядит следующим образом:

$$КПО = (100\% - ЧВО - ЧНСК + ППТ + УО + ОКК) \times 0.2,$$

где ОКК – оценка качества программного обеспечения по ключевым критериям. Расчет данного показателя осуществляется в рамках конкретного проекта и в зависимости от способа измерения выбранных критериев качества.

Оценка качества программного обеспечения и интерпретация результатов расчёта показателя (заключительный этап процедуры) может проводиться в соответствии с данными табл. 2.

Таблица 2

### Шкала оценки качества программного обеспечения

| Диапазон результата | Оценка                                 |
|---------------------|--|
| 80 – 100%           | Высокое качество продукта              |
| 60 – 79%            | Приемлемое качество продукта           |
| 40 – 59%            | Удовлетворительное качество продукта   |
| Менее 40%           | Неудовлетворительное качество продукта |

Таким образом, процедура оценки качества программного обеспечения должна включать в себя описание программного обеспечения и основных требований к нему, выбор ключевых критериев качества и показателей качества исходного кода, выбор метрик, вычисление показателей качества с последующим объединением в единый показатель и оценку качества программного обеспечения на основе шкалы оценки качества программного продукта. Стоит отметить, что внедрение данной процедуры в практику оценки качества создаваемого продукта на малом предприятии ИТ-сферы не потребует дополнительных финансовых затрат, а также привлечения большого числа новых сотрудников.

### Экономический эффект

В ходе исследования были проанализированы затраты на реализацию проекта малого предприятия ИТ-сферы, основным видом деятельности которого является разработка компьютерного программного обеспечения. Исходные данные по проекту:

- мобильное приложение для бизнес-коммуникаций;
- требования к функционалу: обмен тестовыми сообщениями, создание каналов и чатов, возможность форматировать текст, прикреплять файлы к сообщению, возможность оставлять комментарии к сообщениям, возможность регистрировать пользователей, блокировать пользователей, приглашать пользователей в чат;
- срок выполнения проекта – 3 месяца;
- состав команды проекта – 6 человек.

Было выявлено, что без внедрения процедуры оценки качества программного обеспечения затраты компании составляют 810 000 рублей, а период доработки проекта равен 1,5 месяцам.

Стоит отметить, что при внедрении процедуры оценки качества программного обеспечения период доработки проекта составляет 25 дней, и затраты компании в период реализации проекта – 450 000 рублей. Экономический эффект от внедрения процедуры оценки качества программного обеспечения составит 360 000 рублей.

Таким образом, применение процедуры оценки качества программного обеспечения экономически обосновано.

### Заключение

Основной задачей любого предприятия, в том числе из ИТ-сектора, является увеличение прибыли и снижение затрат. Для компании из сферы информационных технологий снижение затрат в первую очередь связано с

сокращением часов разработки. [14] Однако с увеличением скорости разработки под угрозой оказывается качество разрабатываемого программного обеспечения. Зачастую возникает ситуация, когда менеджмент компании оказывается перед выбором – чему уделить больше внимания: добавлению нового функционала или повышению качества продукта. К сожалению, иногда выбор делается в пользу первого варианта. Однако это крайне ошибочное решение. Повышение качества программного обеспечения в конечном счете оказывается выгоднее, чем добавление нового функционала. [15]

Внедрение процедуры оценки качества программного обеспечения позволит компании не упускать из виду качество разрабатываемого ПО при увеличении скорости разработки, поскольку всегда можно будет количественно оценить показатель качества. Кроме того, оценка даст возможность менеджменту проекта в нужный момент принять решение о переводе проекта на следующий этап, то есть на опытную эксплуатацию клиентом. Тестирование клиентом уже после тщательной оценки качества продукта разработчиками потенциально способно выявить меньшее количество отклонений и тем самым снизить время на дальнейшую разработку, что приведет к уменьшению затрат компании.

### Литература

1. *Евдокимов И.В.* Проблема и показатели качества программного обеспечения // Труды Братского государственного университета. Серия: Экономика и управление. 2019. Т. 1. С. 121-124.
2. *Звягина А.И.* Анализ подходов к управлению качеством в it-отрасли // Альманах научных работ молодых учёных Университета ИТМО. 2020. С. 138-140.
3. *Струбалин П.В., Фатьянова А. А.* Управление качеством программного обеспечения // Вестник Саратовского государственного социально-экономического университета. 2019. №. 2 (76). С. 108-111.
4. *Ахунова Д.Г., Вострых А.В., Курта П.А.* Оценка пользовательского интерфейса информационных систем посредством моделей качества программного обеспечения // Информатизация и связь. 2020. №. 2. С. 127-135.
5. *Лабинцева В.Р., Овчаров А.П., Лукьяненко Т.В.* Роль тестирования в разработке программного обеспечения // Цифровизация экономики: направления, методы, инструменты. 2019. С. 398-401.
6. *Шарафиев Д.Е.* Тестирование программного обеспечения // ББК 1 А28. 2020. С. 88.
7. *Золотухина Е.Б., Макарова Е.А., Беляков А.А.* Обзор методов тестирования программного обеспечения // Аллея науки. 2018. Т. 4. №. 6. С. 10-18.
8. *Воротникова Т.Ю.* Надежный код: статический анализ программного кода как средство повышения надежности программного обеспечения информационных систем // Информационные технологии в УИС. – 2020. №. 2. С. 22-27.
9. *Босик В.В.* Оценка качества программного обеспечения с помощью метрик/АА Босик, ВВ Садовой, ГВ Сечко. – Минск: Бестпринт. – 2018.
10. *Михеев И.В., Виштак О.В., Кондратов Д.В.* Система количественных характеристик оценки качества программных продуктов // Программные системы и вычислительные методы. 2018. №. 2. С. 28-35.
11. *Смирнов А.В.* Методы оценки и управления качеством программного обеспечения // Известия СПбГЭТУ «ЛЭТИ. – 2019. – №. 2. – С. 20-25.
12. *Бураков Д. П., Кожомбердиева Г. И.* Использование формулы Байеса при оценивании качества программного обеспечения согласно стандарту ISO/IEC 9126 // Программные продукты и системы. – 2019. – №. 1. – С. 034-041.
13. *Цветков В. Я.* Метрики сложных систем // Славянский форум. – 2019. – №. 2. – С. 158-164.
14. *Стефанова Н.А., Курбангелдыев Д.* Оценка стоимости разработки программного обеспечения // Актуальные вопросы современной экономики. – 2020. – №. 1. – С. 67-72.
15. *Смирнов А. А. и др.* Метод количественной оценки рисков разработки программного обеспечения. – 2016.

### Reference

1. *Evdokimov I. V.* The problem and indicators of software quality // *Proceedings of the Bratsk State University. Series: Economics and Management.* 2019. T. 1. S. 121-124.
2. *Zvyagina A. I.* ANALYSIS OF APPROACHES TO QUALITY MANAGEMENT IN THE IT INDUSTRY // *Almanac of Scientific Works of Young Scientists of ITMO University.* 2020. S. 138-140.
3. *Strubalin P. V., Fatyanova A. A.* Software quality management // *Bulletin of the Saratov State Socio-Economic University.* 2019. no. 2 (76). - S. 108-111.
4. *Akhunova D. G., Vostrykh A. V., Kurta P. A.* Evaluation of the user interface of information systems by means of software quality models // *Informatization and communication.* 2020. no. 2. S. 127-135.

5. Labintseva V. R., Ovcharov A. P., Lukyanenko T. V. The role of testing in software development // *Digitalization of the economy: directions, methods, tools*. 2019. S. 398-401.
6. Sharafiev D. E. SOFTWARE TESTING //BBK 1 A28. 2020. S. 88.
7. Zolotukhina E. B., Makarova E. A., Belyakov A. A. Review of software testing methods // *Alley of Science*. 2018. T. 4. No. 6. S. 10-18.
8. Vorotnikova T. Yu. Reliable code: static analysis of program code as a means of improving the reliability of information systems software // *Information technologies in UIS*. 2020. no. 2. S. 22-27.
9. Bosik V. V. Software quality assessment using metrics / AA Bosik, VV Sadovoy, GV Sechko // Minsk: Bestprint. 2018.
10. Mikheev I.V., Vishtak O.V., Kondratov D.V. The system of quantitative characteristics for evaluating the quality of software products // *Program systems and computational methods*. 2018. no. 2. S. 28-35.
11. Smirnov A. V. Methods for assessing and managing the quality of software // *Izvestiya SPbGETU "LETI*. 2019. no. 2. - S. 20-25.
12. Burakov D. P., Kozhombardieva G. I. Using the Bayes formula in assessing the quality of software according to ISO/IEC 9126 // *Software products and systems*. 2019. no. 1. S. 034-041.
13. Tsvetkov V. Ya. Metrics of complex systems // *Slavic Forum*. 2019. no. 2. S. 158-164.
14. Stefanova N. A., Kurbangeldiev D. Estimation of the cost of software development // *Actual issues of modern economics*. 2020. no. 1. S. 67-72.
15. Smirnov A. A. et al. Method for quantitative risk assessment of software development. – 2016.

Статья поступила в редакцию 18.03.2022  
Принята к публикации 19.05.2022

Received 18.03.2022  
Accepted for publication 19.05.2022