

## СРЕДСТВО ДЛЯ ОПРЕДЕЛЕНИЯ КАЧЕСТВА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ МЕТОДАМИ МЕТРИЧЕСКОГО АНАЛИЗА

Д.С. Лаптов, бакалавр  
Челябинский государственный университет  
(Россия, г. Казань)

DOI:10.24412/2500-1000-2023-9-1-245-255

**Аннотация.** Разработано современное программное средство для определения качества программного обеспечения (ПО) методами метрического анализа, позволяющее с помощью показателей качества рассчитать соответствующие метрики и определить значение комплексного показателя качества программного продукта. Выявлены особенности процесса оценки качества ПО, т.е. проанализировано понятие качества программного продукта как предмет стандартизации, а также уровни представления модели качества ПО, что позволило установить возможность ее повышения путем формирования соответствующих требований к критериям оценки качества, совершенствованию моделей метрического анализа его качества и методов количественного ее измерения на всех этапах реализации программного проекта. Установлены особенности использования метрического анализа для определения качества ПО, согласно которым существует отсутствие единых стандартов на метрики, поэтому каждый поставщик измерительной системы предлагает свои методы оценки качества ПО и соответствующие метрики. Также сложна задача интерпретации значений метрик, так как для большинства пользователей как метрики, так и их значения не совсем понятны и информативны. Выявлено, что основными параметрами при выборе варианта реализации ПО является его стоимость и продолжительность процесса разработки и репутация фирмы-проектанта, но решения, принятые на основании этих параметров, не всегда гарантируют надлежащее качество ПО.

**Ключевые слова:** информационные технологии; программный проект; стандарты качества; требования к программному обеспечению; спецификация требований; критерии оценки качества; показатели качества программного обеспечения; комплексный показатель качества.

Качество программного обеспечения (ПО) является основной его характеристикой в различных областях использования информационных технологий (Pleskach, Zatonatska, 2011), которая указывает на степень его соответствия установленным требованиям (ISO 9001, 2008). Обычно такие требования трактуют по-разному, что порождает несколько независимых определений этого термина.

Если внешние характеристики отражают требования к функционированию ПО, внутренние характеристики используют для подготовки плана достижения необходимых значений внешних характеристик его качества (ISO/IEC TR 9126-2, 9126-3, 2003). Как внешние, так и внутренние свойства свойства отражают характери-

стики самого ПО, также взгляды заказчика и разработчика на него.

Однако непосредственного пользователя ПО в основном интересует эксплуатационное качество (ISO/IEC 9126-1, 2001), то есть совокупный эффект от достижения необходимых характеристик программы, значение которых измеряется скоростью и достоверностью полученного результата, а не его свойством. Это понятие значительно шире, чем любая отдельная характеристика качества ПО, например удобство его использования или надежность работы (ISO/IEC TR 9126-4, 2004).

Обычно под оценкой качества ПО понимают действия, определяющие, как именно ПО соответствует своему назначению (Kuliamin, Petrenko, 2008). Качество

ПО оценивают с использованием модели качества (ISO/IEC 9126-1, 2001).

Такая оценка приобретает особое значение с развитием и совершенствованием технологий определения качества ПО, а именно – методами метрического анализа. Все это привело к необходимости разработки соответствующего средства для определения качества ПО соответствующими методами.

Анализ последних исследований и публикаций.

На сегодняшний день процесс оценки качества ПО по стандарту ISO 25010:2011 (Gordieiev et al., 2014) происходит в следующем порядке: на основании атрибутов качества, указанных в стандарте ISO 25023:2016, оценивают подхарактеристики и характеристики качества, которые одновременно предоставляют комплексную оценку качества ПО. Оценивание качества ПО с использованием результатов метрического анализа сводится к тому, что на основании показателей качества рассчитывают значение соответствующих метрик качества и значение комплексного показателя качества будущему программному продукту. Согласно стандарту ISO 24765:2010, метрику определяют как степень степени обладания свойством некоторого продукта, имеющего числовое значение.

Проблеме качества ПО посвящены многие работы украинских и иностранных ученых, а именно: В. Харченко и Б. Конорева (Gordieiev et al., 2014; Kharchenko et al., 2007, 2012; Konorev et al., 2009), Маевского (Maievsky & Козина, 2015; Maievskiy, 2013), В. Яковины (Yakovyna, 2012; Yakovyna, Fedasiuk & Mamrokha, 2010), В. Мищенко и О. Поморовой (Mishhenko, 2010; Mishenko, Pomorova, No2 2011), К. Лаврищевой (Lavrishcheva, 2008, 2013), О. Харченко (Kharchenko, Galay & Yatsyshyn, 2011; Kharchenko & Yatsyshyn, 2009), Г. Майерса (Myers, Badzhett & McConnell, 2013, 2006), Р. Фатрелла (Fatrell, Shafer & Shafer, 2003), И. Соммервилла (Sommerville, 2002), О. Медче (Maedche, Botzenhardt & Neer, 2012), С. Джонса (0, 2); Jones, Bonsignour, 2012).

Однако процедура оценки качества ПО и существующие методы и средства обеспечения этого качества, как, собственно, и процесс разработки самого ПО остаются необеспеченными фундаментальной теорией и эффективной методологией (Andon et al., 2002). Все исследования в области оценки качества ПО, особенно на ранних этапах его жизненного цикла, носят хаотический, несистематизированный характер (Braude, 2004). В то же время, как доказано в работах (Pomogova & Novorushchenko, 2013a, 2013b; Pomogova, Novorushchenko & Tarasek, 2010), именно в конце этапа проектирования архитектуры ПО можно и следует выявлять и устранять до 55% всех недостатков будущего программного продукта. Безусловно, есть много фундаментальных исследований по инженерии ПО (работы Боэма, Дейкстры, Мейера), но отсутствует завершенная, протестированная и апробированная теория и методология разработки сложного и одновременно качественного ПО, а также методы и средства оценки и прогнозирования его качества на ранних этапах реализации программного проекта. Поэтому проблема оценки качества ПО требует неотложных изменений для предотвращения непредвиденных потерь и неприятных инцидентов, вызванных ошибками его работы (Novorushchenko, 2018).

Не претендуя на кардинальные сдвиги в современных технологиях оценки качества ПО, попытаемся внести и свою лепту в методы метрического анализа его качества, особенно программных средств для его определения.

Поэтому, как сегодня, кажется нам актуальным исследование, которое касается разработки адекватного средства для определения качества ПО методами метрического анализа.

Для реализации указанной цели нужно выполнить следующие основные задачи:

- 1) выяснить особенности процесса оценки качества ПО, проанализировать его качество как предмет стандартизации, а также уровни представления модели качества ПО;

- 2) установить особенности использования метрического анализа для определения

качества ПО, позволяющих выяснить причины их неэффективного использования, а также различные интерпретации значений этих метрик;

3) разработать программное средство для определения качества ПО, которое обеспечило бы возможность прогнозирования дальнейшей эффективности процесса его разработки и позволило сформировать соответствующее множество данных для определения значения комплексного показателя качества программного продукта;

4) сделать соответствующие выводы и предоставить рекомендации по практическому использованию средства определения качества ПО методами метрического анализа.

#### Особенности процесса оценки качества программного обеспечения

Современные технологии разработки ПО достигли такого уровня своего развития, что возникла потребность в использовании инженерных методов оценки ре-

зультатов его проектирования на всех этапах реализации программного проекта, контроля степени достижения запланированных показателей качества и их метрического анализа, оценки риска возможных опасностей и степени использования готовых компонентов для снижения стоимости реализации нового проекта (Andon et al., 2002; Лураев, 2001). Основу инженерных методов оценки качества ПО составляет возможность ее повышения путем формирования соответствующих требований к критериям оценки качества, совершенствования моделей метрического анализа его качества и методов количественного ее измерения на всех этапах реализации программного проекта (Pomorova & Novorushchenko, 2009, 2010; Pomorova, Novorushchenko & Onyshchuk, 2011; Поморова, Ховорущенко & Тарасек, 2010). Статические технологии оценки качества ПО приведены на рисунке 1, в то же время как динамические технологии так или иначе связаны с непосредственным тестированием.



Рис. 1. Статические технологии оценки качества ПО

Обычно согласие, достигнутое разработчиком ПО с его заказчиком относительно пользовательских требований к ПО, в т.ч. и его качества, наряду с непосредственным доведением системных инженеров того, что составляет для заказчика качество будущего продукта, требует отдельного обсуждения и формального определения многих особенностей этого согласия (Hrytsiuk, 2018). Прежде всего, системные инженеры должны понимать

содержание, вложенное заказчиком в концепцию качества ПО, его подхарактеристики, характеристики и атрибуты качества на всех этапах реализации программного проекта. Ведь пользовательские требования к ПО и сформированные на них системные требования определяют необходимые характеристики его качества и сформулированные заказчиком соответствующие критерии его оценки и приемки, а также влияют на методы количественно-

го их измерения при подтверждении качества ПО. При этом следует учитывать также как различные риски реализации программного проекта, так и процесса разработки самого ПО.

Доказательство правильности работы ПО – это математическая или логическая методика, используемая для того, чтобы убедить себя (разработчика ПО) и других заинтересованных сторон в том, что разработанное ПО отвечает тем требованиям, которые были определены и согласованы с заказчиком. Такое доказывание является формальным (строгим) методом.

Для хоть какого инженерного продукта существует множество интерпретаций его свойства. Соблюдение планируемых показателей качества ПО можно требовать от его разработчиков в той или иной степени на протяжении всех этапов его разработки. Эти показатели могут быть немного или они могут отражать определенные свойства будущего программного продукта, которые хотели бы видеть его непосредственные пользователи и другие заинтересованные стороны, часто они являются результатом определенного компромисса. Такой подход полностью совпадает с пониманием такого понятия, как "приемлемое качество ПО", что является менее жесткой точкой зрения на обеспечение разработчиком качества ПО как гарантированное достижение его совершенства (Koval & Moroz, 2006; Pomorova, Novorushchenko & Onyshchuk, 2011).

Стоимость качества ПО может быть дифференцирована на стоимость предупреждения дефектов, стоимость оценки эффективности работы, стоимость внутренних и внешних сбоев при работе ПО. Движущей силой успешной реализации программных проектов является желание их руководителей разработать такое ПО, обладающее определенной ценностью, то есть было значимым для решения определенных задач или достижения целей – тактических и стратегических. Ценность ПО можно выражать в форме его стоимости или в некотором другом виде. Заказчик обычно имеет свое представление о максимальной стоимости вложения средств, возврат которых он ожидает в случае до-

стижения основных целей при использовании ПО. Он также может иметь свое видение функциональных возможностей ПО и определенные ожидания его качества.

Обычно заказчик сначала концентрирует свое внимание на функциональных возможностях ПО и не задумывается над вопросами его качества, а тем более над связанной с ними стоимостью его разработки. Поэтому на начальном этапе реализации программного проекта предметом обсуждения может стать вопрос о полном понимании заказчиком выгоды от использования ПО и стоимости его разработки, связанной с достижением того или иного уровня качества ПО, а также его вовлечении в процесс принятия соответствующих решений. В идеальном случае большинство таких решений следует принимать на этапе установления пользовательских требований к ПО, но эти вопросы могут (и должны) подниматься на протяжении всех этапов его разработки. Не существует каких-то "стандартных" правил относительно того, как именно необходимо принимать такие решения. Однако системные инженеры должны четко представлять различные альтернативные способы достижения определенного уровня качества ПО и соответствующую стоимость его разработки для того, чтобы можно было спрогнозировать общую стоимость реализации программного проекта.

Для наглядной демонстрации зависимости стоимости реализации программного проекта от уровня качества ПО рассмотрим особенности разработки системы защиты информации (СЗИ), а именно ее функциональную модель (рис. 2) (Gryciuk & Sivec, 2016; Hrytsiuk & Sivets, 2016a, 2016b). Эта модель не показывает ценность информации – объекта конфиденциальности (например, счета банковских вкладов или коды доступа к ним, поскольку эта информация не теряет своей ценности с течением времени). Итак, на рисунке введены следующие обозначения:  $P$  – уровень (вероятность) обеспечения защиты информации (практически  $0,6 \leq P < 1,0$ );

$Z(P)$  – допустимые затраты на защиту информации как функция от требуемого уровня ее защиты. Эти издержки возрас-

тают при повышении требований к определенному уровню обеспечения защиты информации.

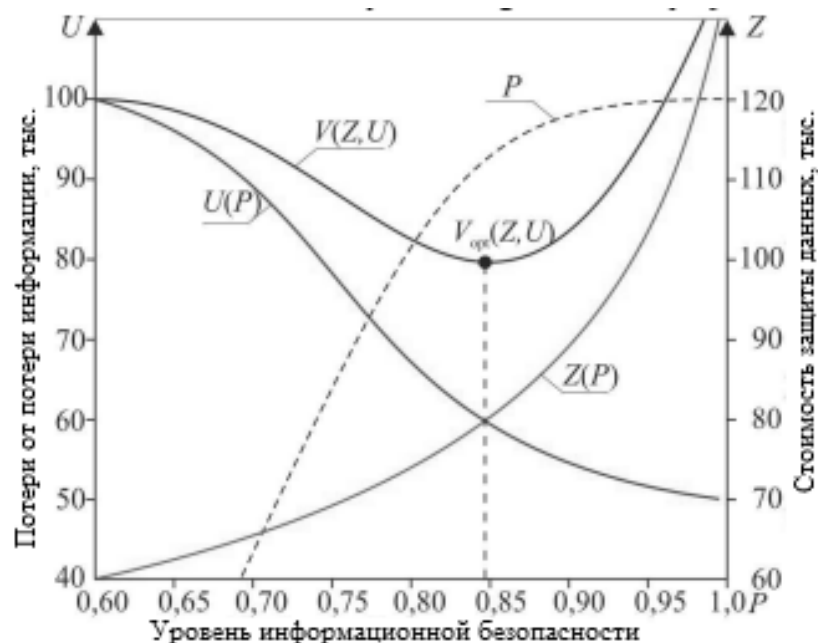


Рис. 2. Основные особенности процесса оценки качества СЗИ

Стремление добиться очень высокого уровня обеспечения защиты информации, как правило, приводит к резкому росту затрат, которые могут превысить ценность самой информации, которую необходимо защитить. Возможные потери (убытки) владельца информации  $U(P)$ , понесенные вследствие ненадлежащего уровня ее защиты, являются функцией от имеющегося уровня обеспечения ее защиты  $P$ . Из рисунка видно, что сумма  $Z(P) + U(P)$  определяет расходы  $V(Z, U)$  для обеспечения защиты информации. При этом оптимальный уровень ее защиты  $V_{opt}(Z, U)$  будет соответствовать минимуму суммы затрат на защиту  $Z(P)$  и возможным убыткам  $U(P)$  вследствие потери информации, то есть неполноты ее защиты.

Стремление превысить его приведет к резкому росту издержек  $Z(P)$  на обеспече-

ние защиты информации; снижение же уровня обеспечения защиты приведет к увеличению возможного ущерба  $U(P)$  вследствие несовершенства функционирования СЗИ.

Следовательно, качество ПО является относительным понятием, смысл которого заказчики понимают только при учете реальных условий его применения, поэтому требования, предъявляемые к качеству ПО соответствующими стандартами, должны соотноситься с условиями его использования и конкретной областью его применения (Pomorova, Novorushchenko & Onyshchuk, 2011). Качество ПО характеризуется следующими компонентами: качеством процессов разработки ПО, качеством продуктов программного проекта и качеством сопровождения или внедрения ПО (рис. 3).



Рис. 3. Основные компоненты оценки качества ПО

### Использование метрического анализа для определения качества программного обеспечения

Современное ПО характеризуется сложностью описания, наличием набора тесно взаимосвязанных компонентов с локальными задачами и глобальными целями, отсутствием полных аналогов и, как следствие, невозможностью использования любых типовых проектных решений. Кроме этого, новое ПО обычно требует полной интеграции с имеющимся ПО, функционированием в неоднородной среде на различных аппаратных платформах, часто обособленностью составляющих в пространстве и разнородностью групп пользователей с разным уровнем квалификации (Pomorova & Novorushchenko, 2009).

Большое количество ошибок ПО возникает на этапе формулировки требований (10-23%), где наблюдается следующая тенденция: чем больше объем ПО, тем больше ошибок вносится концептуальных ошибок (Hrytsiuk, 2018).

Кроме того, в процессе формулирования требований к ПО происходят инфор-

мационные потери из-за неполноты и разного понимания потребностей заказчиков и контекста информации в спецификации требований. Особенно такие потери информации ощутимы для программных проектов, относящихся к стыку различных предметных областей знаний. Программные проекты с неполными требованиями и плохо подготовленными спецификациями требований обычно не могут иметь успешной реализации. При таких условиях актуален и очень важен анализ спецификации требований к ПО независимыми экспертами для того, чтобы избежать ошибок на этапах формулирования требований, проектирования архитектуры ПО и последующего его конструирования (Hrytsiuk, 2018).

Из приведенных данных в таблице 1 видно, что ошибки формулирования требований к ПО и проектирование его архитектуры составляют 25-55% от всех возможных ошибок, причем чем больше объем ПО, тем больше ошибок вносится именно на ранних этапах его разработки.

Таблица 1. Распределение ошибок, допущенных на разных этапах разработки ПО

Этапы разработки ПО	Объем ПО/часть ошибок, %				
	2К	8К	32К	128К	512К
Формирование требований	10	15	20	22	23
Проектирование архитектуры	15	19	25	28	32
Конструирование	75	66	55	50	45

На сегодняшний день оценка качества ПО по стандарту ISO 25010:2011 происходит следующим образом: на основании атрибутов качества, указанных в ISO 25023:2016, оценивают подхарактеристики и характеристики качества ПО, которые в то же время дают комплексную его оценку. Согласно данным стандарта ISO 25010:2011, характеристика – это набор свойств ПО, с помощью которых можно описать и оценить его качество. Подхарактеристика качества ПО выражается средневзвешенным арифметическим показателем с учетом значений атрибутов, оценивающих эту подхарактеристику, и коэффициентов их значимости (Pomogova & Novorushchenko, 2010).

Существует ряд моделей, позволяющих рассчитывать качество ПО, однако многозначность трактовки этих характеристик усложняет следующие расчеты. Большинство моделей базируются на использовании различных метрик ПО (Pomogova, Novorushchenko & Onyshchuk, 2011). История программных метрик насчитывает более четверти века, то есть с того момента, когда стоимость коммерческих программных продуктов начала расти и потребовались научные методы анализа процессов разработки программного обеспечения. Современная ИТ-индустрия накопила большое количество метрик, позволяющих оценить отдельные производственные и эксплуатационные свойства ПО (Andon et al., 2002; Zalewski, Kornecki & Pfister, 2018).

Оценивание качества ПО на основании использования результатов метрического анализа происходит следующим образом: на основании показателей качества ПО рассчитывают значения метрик качества, которые, одновременно, предоставляют комплексную оценку качества ПО.

В свое время введение количественных метрик качества ПО должно способствовать решению некоторых практических задач (Lavrishcheva, 2008):

1) прогнозирование количества ошибок в ПО от начала проектирования;

2) прогнозирование уровня сложности ПО и его сопровождения из анализа результатов проектирования;

3) прогнозирование уровня сложности процессов тестирования и количества не выявленных ошибок из анализа кода программы;

4) прогнозирование окончательного размера кода программы на основании анализа оценок сложности проектирования архитектуры ПО;

5) определение влияния отдельных характеристик программного кода на качество готового ПО;

6) контроль этапов реализации программного проекта;

7) анализ явных и скрытых дефектов в готовом ПО;

8) выявление лучших методов и технологий разработки ПО на основании их сравнения.

Следовательно, качество ПО можно измерить с помощью метрика качества. По определению стандарта ISO/IEC 9126-2, метрика качества ПО является «моделью измерения атрибута, связанного с характеристикой качества ПО. Это комбинация конкретного метода измерения (способа получения значений) атрибута сущности и шкалы измерения» (Andon et al., 2002). Согласно данным стандарта ISO 24765:2010, метрику можно определить как меру степени обладания свойством, имеющим числовое значение. В общем, метрика ПО – это мера, позволяющая получить числовое значение некоторого свойства ПО как средневзвешенное ариф-

метическое с учетом значений показателей, оценивающих эту метрику, и коэффициентов их веса (Lavrishcheva, 2013).

Несмотря на многочисленные исследования программных метрик (Pomogova & Novorushchenko, 2009; Pomogova, Novorushchenko & Tarasek, 2010), однако остается много нерешенных вопросов. Одним из таких вопросов является отсутствие единых стандартов на метрики (создано более тысячи метрик), поэтому каждый поставщик измерительной системы предлагает свои методы оценки качества ПО и соответствующие метрики. Также сложна задача интерпретации значений метрик, так как для большинства пользователей как метрики, так и их значения не совсем понятны и информативны. На этапе проектирования архитектуры ПО основное внимание при выборе подходящего проекта уделяется стоимости его реализации, продолжительности процесса разработки, репутации фирмы-проектанта и технологиям разработки ПО. По статистическим данным (Pomogova & Novorushchenko, 2009; Pomogova, Novorushchenko & Tarasek, 2010) только 1,5% программных компаний пытаются оценить качество процессов и готового продукта количественно с помощью метрик, и только 0,5% этих компаний пытаются улучшить работу, руководствуясь количественными критериями качества ПО в целях изготовления бездефектных продуктов. Кроме этого, современные технологии измерения качества ПО еще не достигли своей «зрелости», поскольку только 0,5% программных компаний работают по так называемой модели зрелости способностей CMM (Pomogova & Novorushchenko, 2013a; Pomogova, Novorushchenko & Tarasek, 2010).

Для организации процесса управления рисками разработки ПО руководителю

проекта необходимо прежде всего научиться прогнозировать причины возникновения тех или иных потенциальных проблем, появления негативных ситуаций или неблагоприятных событий. Под прогнозом следует понимать научно обоснованное суждение о возможных состояниях процесса управления реализацией программного проекта, об альтернативных траекториях управления и сроках его выполнения. Прогнозирование управленческих решений теснейшим образом связано со стратегическим и тактическим планированием рисков реализации программных проектов.

Данное программное средство было разработано в среде разработки Microsoft Visual Studio.NET 2017. Интернет-соединение для корректной работы средства не требуется. Выполнение задания начиналось с детального прототипирования пользовательского интерфейса с постепенным наращиванием функционала программного средства. Разработанный интерфейс программного средства можно посмотреть на рисунке 5,а.

Класс `MetricsQualitySoftware.cs` абстрактен с базовыми функциями, необходимыми для нахождения значения метрики. Среди основных методов класса являются: установка нового значения параметра метрики (`ChangeValue_OfParameter`), получение названия параметра (`GetNameOfParameter`), установление основной информации о метрике (`SetInformation_OfMetric`), установление дефолтных значений параметров метрики (`SetAllParameters WithDefault description_OfMetric`), функция нахождения значения метрики (`FindMetric`); очистка значений параметров метрики (`ClearAllParameters_OfMetric`).



The figure displays three sequential screenshots of a software application interface for software quality metrics analysis. Each screenshot shows a menu bar (Файл, Основне, Пошук, Звіт, Налаштування, Інформація, Допомога) and a toolbar with various icons.

**Top Screenshot: Main Metrics Overview**

Left sidebar: **Множина метрик якості ПЗ**

- Метрики з точними значеннями
  - Метрика зв'язності
  - Метрика зчеплення
  - Метрика звернення до глобальної змінної
  - Метрика часу модифікації моделей
  - Метрика загальної кількості знайдених помилок
- Метрики з прогнозованими значеннями
  - Метрика прогнозованого загального часу розробки ПЗ
  - Метрика часу виконання робіт етапу проектування ПЗ
  - Метрика очікуваної вартості розроблення ПЗ
  - Метрика прогнозованої вартості перевірки якості ПЗ
  - Метрика прогнозованої продуктивності розроблення ПЗ
  - Метрика прогнозованих витрат на реалізацію програмної
  - Метрика прогнозованого функційного розміру
  - Метрика прогнозованої оцінки трудointрат за м
  - Метрика прогнозованої оцінки тривалості проєк

Right panel: **Розрахування метрик**

Buttons: Задати, Оновити, Очистити

Star rating: 0,833

Time: 09:17:11

**Middle Screenshot: Detailed Metrics for Boehm Model**

Left sidebar: Same as top screenshot.

Right panel: **Метрика прогнозування оцінки трудointрат за моделлю Боєма**

№	Значення	Назва параметра
1	3	Кількість тисяч рядків програмного коду
2	3.6	Коефіцієнт COSOMO (a)
3	1.2	Коефіцієнт COSOMO (b)

Buttons: Задати, Оновити, Очистити

Star rating: 13,453

Time: 09:18:27

**Bottom Screenshot: Detailed Metrics for Project Duration**

Left sidebar: Same as top screenshot.

Right panel: **Метрика прогнозування оцінки тривалості проєкту за моделлю Боєма**

№	Значення	Назва параметра
1	5	Кількість тисяч рядків програмного коду
2	3.6	Коефіцієнт COSOMO (a)
3	1.2	Коефіцієнт COSOMO (b)
4	2.5	Коефіцієнт COSOMO (c)
5	0.32	Коефіцієнт COSOMO (d)

Buttons: Задати, Оновити, Очистити

Star rating: 6,988

Time: 09:17:11

Bottom status bar: Готово | Заповнення значеннями вхідних параметрів DP метрики успішно завершено

Рис. 5. Окна программного средства для определения качества ПО методами метрического анализа

Следовательно, разработано программное средство для определения качества ПО методами метрического анализа, которое обеспечивает возможность прогнозирования дальнейшей эффективности процесса его разработки и сформировать соответствующее множество данных для определения значения комплексного показателя качества программного продукта. Приведен пример работы программного средства, а также проведено исследование определения качества некоторого ПО и общей прогнозируемой оценки успешности процесса его разработки.

По результатам исследования можно сделать следующие **основные выводы**:

1. Выявлены особенности процесса оценивания качества ПО, т.е. качества и методов количественного измерения на всех этапах реализации программного проекта.

2. Выявлено, что движущей силой успешной реализации программных проектов является желание их руководителей разработать такое ПО, обладающее определенной ценностью, то есть было значимым для решения определенных задач или достижения целей – тактических и стратегических. Ценность ПО можно выражать в форме его стоимости или в некотором другом виде. Заказчик обычно имеет свое представление о максимальной стоимости вложения средств, возврат которых он ожидает в случае достижения основных целей при использовании ПО. Он также может иметь свое видение функциональ-

ных возможностей ПО и определенные ожидания относительно его качества работы.

3. Установлены особенности использования метрического анализа определения качества ПО, согласно которым существует отсутствие единых стандартов на метрики, поэтому каждый поставщик измерительной системы предлагает собственные методы оценки качества ПО и соответствующие метрики. Также сложна задача интерпретации значений метрик, так как для большинства пользователей как метрики, так и их значения не совсем понятны и информативны. Выявлено, что основными параметрами при выборе варианта реализации ПО является его стоимость и продолжительность процесса разработки и репутация фирмы-проектанта, но решения, принятые на основании этих параметров, не всегда гарантируют надлежащее качество ПО.

4. Разработано программное средство для определения качества программного обеспечения методами метрического анализа, которое обеспечивает возможность прогнозирования дальнейшей эффективности процесса его разработки и сформировать соответствующее множество данных для определения значения комплексного показателя качества программного продукта.

5. Сделаны соответствующие выводы и даны рекомендации по использованию разработанной методики визуализации информации.

#### **Библиографический список**

1. Андон Ф.И., Коваль Г.И., Коротун Т.М., Суслов V.Iu. Основы инженерии качества программных систем, (Сергиенко, И. В. С. Е.). – Киев: Akadempriodika, 2002. – 504 p.
2. Andrushchakevych O.T., Hrytsiuk Ю.І. (2018). Использование метрического анализа для выражения качества программного заболевания // Науки doslidzhennia: закономірності і пара-doksy: zb. mater. mizhdystsyplinar. nauk.-prakt. konf. (pp. 23-29), 18 travnia 2018 г. Киев. – [Электронный ресурс]. – Режим доступа: <http://futurolog.com.ua/publish/8/Zbirnyk.pdf>.
3. Braude, E. Технология разработки программного обеспечения. – St. Petersburg: Izd-vo "Piter", 2004. – 655 p.
4. Fatrell R.T., Shafer D.F., Shafer L.I. Управление прогрессивными проектами: повышение оптимального качества при минимуме затрат, (перевод от English). – Москва: Izd. dom "Viliams", 2003. – 1136 p.
5. Гордиев О., Харьков В., Фоминих Н. и Сляяр В. (2014). Разработка стандартных стандартов ISO 25010 в рамках стандартных стандартов ISO.

6. Gryciuk Ю.І., Sivec О.О. Ground of reasonable sufficiency structure of the system of defence of informative resources of enterprise. Scientific Bulletin of UNFU. – 2016. – № 26(7). – P. 378-388. – Львов: РОВ НЛТУ Украины. <https://doi.org/10.15421/40260759>.

7. Ховорущенко, Т.О. Теоретические и плодовые средства информации-матрионной технологической оценке достаточности информации щитовидной качества при спецификаций вымогут в прогрессивном развитии. Abstract of Doctoral Dissertation for Technical Sciences. Львов: Украинская академия типографии, 2018. – 43 p.

8. Грицюк, Ю.І. Analysis of Software Requirements: Tutorial. – Lviv: Публикация House of Lviv Polytechnic, 2018. – 460 p.

## A TOOL FOR DETERMINING THE QUALITY OF SOFTWARE BY METHODS OF METRIC ANALYSIS

**D.S. Laptov**, *Bachelor*  
**Chelyabinsk State University**  
**(Russia, Kazan)**

**Abstract.** *A modern software tool for determining the quality of software (software) by methods of metric analysis has been developed, which allows using quality indicators to calculate the corresponding metrics and determine the value of a complex indicator of the quality of a software product. The features of the software quality assessment process, i.e. the concept of software product quality as a subject of standardization is analyzed, as well as the levels of presentation of the software quality model, which made it possible to establish the possibility of improving it by forming appropriate requirements for quality assessment criteria, improving models of metric analysis of its quality and methods of quantitative measurement at all stages of the software project implementation, the features of using metric analysis to determine software quality, according to for which there is a lack of uniform standards for metrics, therefore, each supplier of the measuring system offers its own methods of software quality assessment and corresponding metrics. The task of interpreting metric values is also difficult, since for most users both metrics and their values are not completely clear and informative. It is revealed that the main parameters when choosing a software implementation option are its cost and duration of the development process and the reputation of the design firm, but decisions made on the basis of these parameters do not always guarantee the proper quality of the software.*

**Keywords:** *information technology; software project; quality standards; software requirements; specification of requirements; quality assessment criteria; software quality indicators; comprehensive quality indicator.*