

Министерство образования Республики Беларусь
Учреждение образования
«Белорусский государственный университет
информатики и радиоэлектроники»

На правах рукописи

УДК 331.103:005.8

БАДЕЕВ
Артем Александрович

**МЕТОДЫ И СРЕДСТВА ОЦЕНКИ ТРУДОЗАТРАТ ДЛЯ
СОВЕРШЕНСТВОВАНИЯ РАЗРАБОТКИ IT-ПРОЕКТОВ**

АВТОРЕФЕРАТ
диссертации на соискание степени
магистра экономических наук

по специальности 1-25 80 08 «Математические и инструментальные
методы экономики»

Минск 2016

Работа выполнена на кафедре экономической информатики учреждения образования «Белорусский государственный университет информатики и радиоэлектроники»

Научный руководитель:

Поттосина Светлана Анатольевна,
кандидат физико-математических наук, доцент кафедры экономической информатики учреждения образования «Белорусский государственный университет информатики и радиоэлектроники»

Рецензент:

Ткалич Татьяна Алексеевна,
доктор экономических наук, профессор кафедры информационных технологий учреждения образования «Белорусский государственный экономический университет»

Защита диссертации состоится «24» июня 2016 г. года в 14⁰⁰ часов на заседании Государственной комиссии по защите магистерских диссертаций в учреждении образования «Белорусский государственный университет информатики и радиоэлектроники» по адресу: 220013, г.Минск, ул. П.Бровки, 6, 5 уч. корп., ауд. 806, тел.: 293-89-92, e-mail: kafei@bsuir.by.

С диссертацией можно ознакомиться в библиотеке учреждения образования «Белорусский государственный университет информатики и радиоэлектроники».

ВВЕДЕНИЕ

В любом программном проекте приходится балансировать между стоимостью, временем, качеством и объемом реализуемой функциональности. Соответственно, точный расчет ресурсов, необходимых для реализации данного продукта с заданными требованиями к качеству, является одной из основных проблем в области управления проектами. При этом расчете возникают сложности учета огромного количества факторов, влияющих на жизненный цикл программного обеспечения. Как следствие, сегодня многие компании сталкиваются с серьезными проблемами в случае неправильных расчетов необходимых сроков.

Недооценка стоимости, времени и ресурсов, требуемых для создания информационной системы, влечет за собой недостаточную численность проектной команды, чрезмерно сжатые сроки разработки и, как результат, утрату доверия к разработчикам в случае нарушения графика. С другой стороны, перестраховка и переоценка могут оказаться ничуть не лучше. Если для проекта выделено больше ресурсов, чем реально необходимо, причем без должного контроля за их использованием, то ни о какой экономии ресурсов говорить не приходится. Такой проект окажется более дорогостоящим, чем должен был быть при грамотной оценке, и приведет к запаздыванию с началом следующего проекта.

На современном рынке крупных программных систем потери могут исчисляться миллиардами долларов. По оценке The Standish Group во всем мире в 2015 году было потрачено 750 млрд. долларов на проекты разработки и внедрения программных продуктов. При этом затраты в провальные проекты оцениваются в размере 120 млрд. долларов, объем затрат на спорные проекты составил 80 млрд. долларов. То есть получается, что 200 млрд. долларов были потрачены без экономической выгоды.

Точные оценки издержек производства программного обеспечения важны как разработчикам так и клиенту. Они могут использоваться при переговорах о контракте, планировании, контроле. Таким образом, возникает реальная потребность в разработке методов и средств, позволяющих менеджеру оценить требуемые временные и человеческие ресурсы на основе всех имеющихся характеристик проекта: истории предыдущих проектов, опыта и производительности сотрудников, специфики компании. Кроме того, требуется также возможность перерасчета и уточнения сроков и ресурсов уже на этапе разработки системы с учетом текущих тенденций, наблюдаемых при реализации проекта. Это поможет менеджеру своевременно обнаружить отклонения от установленного графика и принять соответствующие меры в управлении проектом.

ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

Актуальность темы исследования

Точный расчет ресурсов, необходимых для реализации данного программного продукта с заданными требованиями к качеству, является одной из основных проблем в области управления проектами. Однако при таком расчете возникают сложности учета огромного количества факторов, влияющих на жизненный цикл ПО. Выяснить скрытые закономерности и связи между этими параметрами, а также произвести аналитический расчет в подавляющем большинстве случаев не представляется возможным. В любом программном проекте приходится балансировать между стоимостью, временем, качеством и объемом реализуемой функциональности. Как следствие, на сегодняшний день многие компании сталкиваются с серьезными проблемами в случае неправильных расчетов необходимых сроков.

The Standish Group в 2015 году проанализировала работу сотен корпораций и итоги выполнения нескольких десятков тысяч проектов по всему миру, связанных с разработкой ПО, в своем докладе с названием «Хаос» пришла к следующим неутешительным выводам:

а) только 39% проектов завершились в срок, не превысили запланированный бюджет и реализовали все требуемые функции и возможности;

б) 43% проектов завершились с опозданием, расходы превысили запланированный бюджет, требуемые функции не были реализованы в полном объеме. Среднее превышение сроков составило 120%, среднее превышение затрат 100%, обычно исключалось значительное число функций;

в) 18% проектов полностью провалились и были аннулированы до завершения.

Показательно, что за последние 8 лет процент успешных проектов вырос всего на 10%, а доля провальных проектов практически не изменилась.

На данный момент есть несколько методов оценки стоимости проектов, но многие из них устарели в настоящее время. Исходя из вышеизложенного, можно сделать вывод о том, что проблема получения адекватной стоимостной и временной оценки реализации проекта является актуальной.

Степень разработанности проблемы

В настоящий период времени выполнено достаточно большое количество исследований в области моделирования оценки стоимости. В основном эти исследования проводятся университетами США и Европы.

Наиболее значимые результаты исследований отражены в работах Нельсона (одно из первых исследований в области оценки стоимости), Волвертона (модель Волвертона), Барри Боэма и Суниты Девнани-Чулани (модель СОСО-МОП), Лоуренса Путнама (концепция функциональных точек, модель SLIM), Каперса Джонса (система Checkpoint), Вячеслава Колдовского.

Классификация моделей оценки стоимости, методы калибровки систем оценки стоимости приведены в трудах Суниты Девнани-Чулани, Костаса Каву-санакиса, Терри Слоана.

Описание современных существующих программных продуктов по оценке стоимости приведено в работе Джини Кингстон и Мартина Бурка.

Анализ исследовательских работ показал, что нет общего мнения по поводу выработки требований к системе оценки стоимости разработки IT-проектов. Так же нет четкого представления, каким набором функций должна обладать система, чтобы отвечать современным запросам рынка. Хотя в достаточной степени проведено исследование моделей по оценке стоимости, слишком малое внимание уделено непосредственно программным продуктам, их сильным и слабым сторонам.

Цель и задачи исследования

Целью работы является повышение точности оценки трудозатрат для создания IT-проектов с помощью анализа и разработки инструментальных средств и методов.

Для достижения поставленной цели необходимо решить **следующие задачи**:

1. Изучить методы оценки трудозатрат на создание программного обеспечения.
2. Разработать двухэтапный метод для оценки времени и стоимости разработки IT-проектов на основе алгоритмической модели СОСОМО II.
3. Разработать инструментальное средство для подсчета стоимости программного обеспечения.

Объектом выступают IT-проекты.

Предметом являются методы оценки стоимости проектов.

Область исследования. Содержание диссертационной работы соответствует образовательному стандарту высшего образования второй ступени (магистратуры) специальности 1-25 80 08 «Математические и инструментальные методы экономики».

Теоретическая и методологическая основа исследования

В ходе исследования изучены труды российских и зарубежных ученых в области моделирования оценки стоимости трудозатрат на разработку программного обеспечения, таких как Волвертон, Барри Боэм, Лоуренс Путнам, Каперс Джонс, Вячеслав Колдовский.

В процессе исследования были использованы методы системного анализа (при раскрытии понятий стоимостной и временной оценки проектов), сравнительный анализ (при выборе конкретных направлений совершенствования модели оценки стоимости и времени проекта). На одном из этапов моделирования инструментального средства использовалась CASE-технология UML.

Информационная база исследования основана на статистических данных и текущих результатах по проектам ЗАО «Оксаджайл».

Научная новизна исследования заключается в разработке двухэтапного метода на основе модели оценки стоимости СОСОМО II с целью повышения точности оценок, в разработке нового программного продукта для совершенствования процесса оценки стоимости проекта, позволяющего производить прогнозирование стоимости, сроков проекта с учетом изменяющихся требований заказчика, размера команды (персонала разработчиков), с возможностью адаптироваться под нужды конкретного предприятия, позволяющего хранить большой объем данных и получать быстрый доступ к ним.

Основные положения, выносимые на защиту

На защиту выносятся следующие основные результаты:

1. Аналитический обзор существующих методов оценки стоимости и сроков выполнения IT-проектов.
2. Двухэтапный метод оценки стоимости и времени разработки программных проектов на основе алгоритмической модели СОСОМО II.
3. Инструментальное средство сбора фактических данных о проектах для исследования применимости полученного метода.

Теоретическая значимость диссертации заключается в том, что в ней предложены модели оценки трудозатрат различных проектов, позволяющие учитывать влияние таких факторов, как опыт и производительность разработчиков, спецификация деятельности компании и история реализации подобных работ.

Практическая значимость диссертационной работы заключается в разработке инструментального средства, которое собирает и хранит фактические данные по реализованным проектам компании, позволяет производить оценку трудозатрат различными методами, рассчитывать продолжительность проекта и оптимальный размер команды.

Апробация и внедрение результатов исследования.

Результаты исследования представлены на Международной научно-практической конференции «Естественно-научные исследования, народное хозяйство, современные технологии и технический прогресс» («Научно-исследовательские публикации», Воронеж, 2016 г.).

Публикации.

Основные положения работы и результаты диссертации изложены в двух работах общим объемом 0,6 п.л. (авторский объем 0,6 п.л.).

Структура и объем работы. Структура диссертационной работы обусловлена целью, задачами и логикой исследования. Работа состоит из введения, общей характеристики работы, трех глав, заключения, списка использованных источников. Общий объем диссертации составляет 71 страниц. Работа содержит 5 таблиц, 28 рисунков. Библиографический список включает 37 наименований.

ОСНОВНОЕ СОДЕРЖАНИЕ РАБОТЫ

Во **введении** рассматриваются общие проблемы в области управления проектами, дается краткая характеристика тенденциям и современной ситуации в области оценки стоимости разработки IT-проектов в мире, выделяется актуальность темы исследований.

В **общей характеристике работы** сформулированы ее цель и задачи, показана связь с научными проектами, даны сведения об объекте исследования и обоснован его выбор, представлены положения, выносимые на защиту, приведены сведения о личном вкладе соискателя, наличие публикаций, а также, структура и объем диссертации.

В **первой главе** дается описание основных проблем при создании программного обеспечения. Выделены критерии успешности проекта. Проект считается успешным, если удовлетворены все требования заказчика и участников проекта. Поэтому у проекта разработки ПО сегодня четыре фактора успеха:

- а) выполнен в соответствие со спецификациями;
- б) выполнен в срок;
- в) выполнен в пределах бюджета;
- г) каждый участник команды удовлетворен результатом своей работы.

Приведен статистический анализ затрат на разработку программного обеспечения. Согласно исследованиям экспертов The Standish Group, только 32% проектов завершились в срок, не превысили запланированный бюджет и реализовали все требуемые функции и возможности. Статистика завершения IT-проектов по годам представлена в таблице 1.

Таблица 1 – Статистика завершения IT-проектов

	2007	2009	2011	2013	2015
Провальные	31%	15%	19%	19%	24%
Спорные	53%	51%	53%	46%	44%
Успешные	16%	34%	29%	35%	32%

Выделяют следующие причин неудач проектов либо выхода проекта за рамки сроков, бюджета и не достижения цели:

- неполные требования;
- низкая степень вовлечения заказчика и конечных пользователей в процесс разработки;
- недостаточное обеспечение ресурсами;
- недостаток планирования;
- и другие.

Переоценка ресурса времени приводит к расслаблению проектной команды заказчика и исполнителя, что не всегда положительно сказывается на качестве работ. Также переоценка времени недопустима в нашу динамическую эпоху.

Переоценка бюджета приводит к бесполезному, бессмысленному расходованию денежных средств.

Недооценка времени сказывается на увеличении бюджета, а при его невозможности увеличения на качестве проекта, что считается недопустимым. Недооценка бюджета также сказывается на качестве проекта.

На практике сталкиваются с 3 проблемами, имеющими принципиальное значение:

- выбор модели оценки;
- выбор метрики размера ПО. Можно выбрать LOC (англ. Lines Of Code – число строк кода), FP (англ. Function Points – функциональные точки или точки свойств) и т.д.;
- понимание того, что вообще является хорошей оценкой.

Размер программного обеспечения – самый важный фактор, определяющий трудоемкость реализации ПО. Число строк исходного кода и функциональные точки – самые популярные метрики.

LOC (Lines Of Code) – число непустых строк исходного текста, исключая комментарии. Несмотря на то, что эта метрика существенно зависит от выбранного языка программирования, она до сих пор остается самой используемой метрикой размера ПО.

Наиболее удачной заменой количеству строк кода для измерения размера стали функциональные точки (Function points). Общее количество функциональных точек программы зависит от количества элементарных процессов пяти типов:



Рисунок 2 – Типы элементарных процессов используемых в методе FP

- EI (англ. External inputs – входящие транзакции) – транзакции, получающие данные от пользователя;
- EO (англ. External outputs – исходящие транзакции) – транзакции, передающие данные пользователю;
- EQ (англ. External inquiries – взаимодействия с пользователем) – интерактивные диалоги взаимодействия с пользователем (требующие от него каких-либо действий);

– ILF (англ. Internal logical files – файлы внутренней логики) – файлы (логические группы информации), используемые во внутренних взаимодействиях системы;

EIF (англ. External interface files – файлы внешних взаимодействий) – файлы, участвующие во внешних взаимодействиях с другими системами.

Также разобрана классификация методов оценки трудозатрат, выделены главные достоинства и недостатки основных методов. Все методы оценки можно разделить на алгоритмические и неалгоритмические.

К неалгоритмическим методам относится оценка по аналогии. Этот метод требует наличия одного или нескольких законченных проектов, которые подобны новому проекту. Оценка получается путем сравнения текущего проекта с предыдущими, с использованием реальных наблюдавшихся в них показателей. Достоинство этого метода: оценка основана на фактическом проектных результатах.

Конечно не всегда корректно переносить результаты предыдущих проектов на текущий, т.к. нельзя с уверенностью сказать, что ограничения и условия предыдущего проекта можно перенести и использовать в новом проекте.

Используемый в алгоритмических моделях математический аппарат весьма разнообразен и варьируется от простых линейных формул, использующих математическое ожидание и среднеквадратичное отклонение, до сложных регрессионных моделей и дифференциальных уравнений. Как правило, чтобы улучшить точность алгоритмических моделей, их требуется приспособить (например, с помощью пересчета основных параметров и коэффициентов) к конкретным обстоятельствам. Причем, даже после такой калибровки точность может оставлять желать лучшего. К алгоритмическим методам относится метод СОСОМО II, который взят за основу двухэтапного метода и подробнее описан во второй главе.

Во **второй главе** произведен анализ алгоритмического метода СОСОМО II, который взят за основу для двухэтапного метода.

Методика СОСОМО позволяет оценить трудоемкость и время разработки программного продукта. Впервые была опубликована Бари Боэмом в 1981 году в виде результат анализа 63 проектов компании «TRW Aerospace». В 1997 методика была усовершенствована и получила название СОСОМО II. Калибровка параметров производилась по 161 проекту разработки. В модели используется формула регрессии с параметрами, определяемыми на основе отраслевых данных и характеристик конкретного проекта.

Различаются две стадии оценки проекта: предварительная оценка на начальной фазе и детальная оценка после проработки архитектуры.

Главной особенностью методики является то, что для того, чтобы оценить трудоемкость, необходимо знать размер программного продукта в тысячах строк исходного кода (KSLOC – Kilo Source Lines Of Code). Размер программного продукта может быть, например, оценен экспертами с применением метода PERT.

В методике используются пять факторов масштаба SF, которые определяются следующими характеристиками проекта:

а) PREC – прецедентность, наличие опыт аналогичных разработок (Very Low – опыт в продукте и платформе отсутствует. Extra High – продукт и платформа полностью знакомы);

б) FLEX – гибкость процесса разработки (Very Low – процесс строго детерминирован. Extra High – определены только общие цели);

в) RESL – архитектура и разрешение рисков (Very Low – риски неизвестны/не проанализированы. Extra High – риски разрешены на 100%);

г) TEAM – сработанность команды (Very Low – формальные взаимодействия. Extra High – полное доверие, взаимозаменяемость и взаимопомощь);

д) PMAT – зрелость процессов (Very Low – CMM Level 1. Extra High – CMM Level 5);

Значения факторов масштаба приведены в таблице 1.

Таблица 1 – Значение фактора масштаба, в зависимости от оценки его уровня

Фактор масштаба	Оценка уровня фактора					
	Very Low	Low	Nominal	High	Very High	Extra High
PREC	6.20	4.96	3.72	2.48	1.24	0.00
FLEX	5.07	4.05	3.04	2.03	1.01	0.00
RESL	7.07	5.65	4.24	2.83	1.41	0.00
TEAM	5.48	4.38	3.29	2.19	1.10	0.00
PMAT	7.80	6.24	4.68	3.12	1.56	0.00

Так же для предварительной оценки трудоемкости программного проекта необходимо оценить уровень семи множителей трудоемкости:

а) PERS – квалификация персонала (Extra Low – аналитики и программисты имеют низшую квалификацию, текучесть больше 45%. Extra High – аналитики и программисты имеют высшую квалификацию, текучесть меньше 4%);

б) RCPX – сложность и надежность продукта (Extra Low – продукт простой, специальных требований по надежности нет, БД маленькая, документация не требуется. Extra High – продукт очень сложный, требования по надежности жесткие, БД сверхбольшая, документация требуется в полном объеме);

в) RUSE – разработка для повторного использования (Low – не требуется. Extra High – требуется переиспользование в других продуктах);

г) PDIF – сложность платформы разработки (Extra Low – специальные ограничения по памяти и быстродействию отсутствуют, платформа стабильна. Extra High – жесткие ограничения по памяти и быстродействию, платформа нестабильна);

д) PREX – опыт персонала (Extra Low – новое приложение, инструменты и платформа. Extra High – приложение, инструменты и платформа хорошо известны);

е) FCIL – оборудование (Extra Low – инструменты простейшие, коммуникации затруднены. Extra High – интегрированные средства поддержки жизненного цикла, интерактивные мультимедиа коммуникации);

ж) SCED – сжатие расписания (Very Low – 75% от номинальной длительности. Very High – 160% от номинальной длительности);

Влияние множителей трудоемкости в зависимости от их уровня определяется их числовыми значениями, которые представлены в матрице, приведенной ниже в таблице 2

Таблица 2 – Значения множителей трудоемкости, в зависимости от оценки их уровня

	Оценка уровня множителя трудоемкости						
	Extra Low	Very Low	Low	Nominal	High	Very High	Extra High
PERS	2.12	1.62	1.26	1.00	0.83	0.63	0.5
RCPX	0.49	0.60	0.83	1.00	1.33	1.91	2.72
RUSE	n/a	n/a	0.95	1.00	1.07	1.15	1.24
PDIF	n/a	n/a	0.87	1.00	1.29	1.81	2.61
PREX	1.59	1.33	1.22	1.00	0.87	0.74	0.62
FCIL	1.43	1.30	1.10	1.0	0.87	0.73	0.62
SCED	n/a	1.43	1.14	1.00	1.00	1.00	n/a

Из этой таблицы, в частности, следует, если в нашем проекте низкая квалификация аналитиков, то его трудоемкость возрастет примерно в 4 раза по сравнению с проектом, в котором участвуют аналитики экстра-класса.

К преимуществам использования LOC, как единицы размера ПО, относят простоту, а недостатками являются следующие:

- размер проекта в LOC может быть определён только после его завершения;
- LOC зависит от языка программирования;
- LOC не учитывает качество кода.

Считается, что чем больше строк кода, тем выше производительность разработчика. Однако, очевидно можно реализовать одну и ту же функцию, написав меньшее количество строк кода. Единица размера LOC не отражает функциональные свойства кода. Поэтому, если разработчик стремится оптимизировать процесс разработки, с целью уменьшения трудозатрат на реализацию проекта, то при использовании LOC как основной единицы размера проекта под уменьшением трудозатрат подразумевается уменьшение количества строк кода в программе, при этом не оценивается его функциональность.

Основной проблемой современных методов оценки трудозатрат является сложность их адаптации к каждому конкретному проекту (калибровки коэффициентов модели). Каждый проект – это своя команда, свои условия труда, своя специфика разработки. Обобщенные модели, такие как СОСОМО, разработанные с учетом данных о большом количестве проектов не могут полностью учесть этой специфики.

Исходя из вышеописанных проблем сделан вывод, что современные методы не способны давать адекватную оценку стоимости и времени на реализацию проекта. Для того чтобы увеличить точность оценки был предложен двухэтапный метод, основными принципами которого являются обеспечение простоты применимости метода на практике и учет информации, специфической для каждого конкретного проекта. В его основе лежит идея разделения общей задачи на фиксированную и динамическую составляющие. При решении фиксированной части используется модель СОСОМО II. Главная же составляющая предлагаемого подхода заключается в способе решения динамической части задачи. Динамическая часть – оценка ресурсов, необходимых для устранения дефектов и других проблем, возникающих в процессе разработки системы, не укладываемых в имеющейся модели. Данный подход предполагает создание базы данных ошибок, которая в последствии позволит учитывать и контролировать ошибки, найденные в программе, а также следить за процессом их устранения. Использование этой БД создает идеальные условия для применения разрабатываемой модели.

Кроме того, данный подход позволяет провести все необходимые расчеты любому менеджеру, даже не являющемуся экспертом в области предсказания сроков, т.к. база данных может быть проанализирована автоматически. Стоимость конечной реализации предлагаемого подхода во много раз меньше имеющихся систем, а оценки, производимые с помощью данного метода, оказались в среднем выше на 4% точнее оценки произведенные с помощью СОСОМО II.

Сформулированы следующие условия применимости метода:

- наличие багтрекера, включающего обязательные поля;
- наличие в распоряжении какого-либо средства для вычисления фиксированной составляющей (например, пакета на основе СОСОМО II);
- базы данных ошибок от предыдущих проектов (чем больше, тем лучше), в которых участвовали те же разработчики, что и в текущем проекте. В случае отсутствия таких баз данных для этого коллектива можно воспользоваться базами данных другого коллектива, работавшего над проектами, подобными текущему, но эффективность метода заметно ухудшается.
- практика регрессионных методов тестирования на протяжении всего жизненного цикла проекта.

В **третьей главе** описывается постановка задач для успешной реализации инструментальной системы. Данная система позволит произвести оценку проекта с помощью нескольких методов, а так же просматривать статистику о результатах предыдущих проектов. На рисунке 3 представлена страница расчета стоимости проекта, используя двухэтапный метод.

Опыт компании

Название Функции: Созд Оценка трудозатраты: 0 ч)

оценка трудозатрат

количество функций

Добавить

Создать кнопку "ОК"

Создать кнопку "Отмена"

Описание функции	оценка	Количество функций	
Создать кнопку "ОК"	2	12	Удалить
Добавление педжинации	3	43	Удалить
Авторизация	1	1	Удалить

Фамилия

Должность

Ставка 0

Добавить

Фамилия	Должность	Ставка	
Бадеев	java-developer	20	Удалить

Посчитать

Время на разработку: 154 (часы)

Количество человек из проекта: 1 (чел)

Итого: 30805

Сохранить Результаты

Рисунок 3 – Страница расчета стоимости проекта с использование двухэтапного метода

На выходе мы получаем оценку стоимости и времени на разработку проекта, а так же расчет человеческих ресурсов необходимых для реализации проекта.

Произведено информационное моделирование Базы Данных. Это позволило разработать инструментальную систему в короткие сроки и с уменьшенными затратами, сделать её легко сопровождаемой и управляемой.

При помощи языка UML была разработана диаграмма вариантов использования, которая является исходным концептуальным представлением или концептуальной моделью системы в процессе ее проектирования и разработки.

ЗАКЛЮЧЕНИЕ

Для эффективного управления проектами необходимо на начальной стадии давать точную оценку стоимости и времени выполнения проекта.

Основной задачей диссертации является увеличение точности оценки трудозатрат на реализацию IT-проектов.

В работе исследованы несколько методов оценки трудозатрат на создание IT-проектов, были выявлены сильные и слабые стороны каждого из методов. разобраны основные метрики размеров проектов. В результате предложен двухэтапный метод на основе алгоритмической модели СОСОМО II, который может быть использовать для повышения точности оценки стоимости проектов.

Разработана инструментальная система для оценки трудозатрат на создание проектов. В программе предусмотрена работа с несколькими методами для оценки проектов, а так же хранение статистической информации по результатам выполненных проектов.

Интерфейс программы динамически изменяется в зависимости от роли пользователя в данной системе, облегчает пользователю ввод необходимых полей, сохраняет передаваемые данные между страницами (JSF), что делает работу с программой более удобной и приятной.

Инструментальная система оценки затрат программного обеспечения реализована на кроссплатформенном языке java с использованием современных технологий. Бизнес логика система представлена распределённой компонентной технологией Spring, доступ к данным осуществляется технологией Hibernate. Интерфейс пользователя реализован при помощи технологии JSF (Primefaces).

Использование данного приложения позволяет повысить точность оценки затрат на разработку программного обеспечения, тем самым предоставляет возможность эффективного планирования времени на отдельные задачи, сокращает время оценки и затраты на проект. А так же позволяет специалистам тратить меньше времени на сам процесс оценки проектов.

Приложение разработано в соответствии с поставленными задачами и спецификой предметной области, обладает функцией расчета трудозатрат и стоимости проекта, исходя из заданных пользователем параметров.

Список опубликованных работ

1–А. Бадеев, А.А. Об оценке трудозатрат на разработку IT-проектов / А.А. Бадеев, С.А. Поттосина // Сборник научных трудов, приуроченный к 20-летию ИЭФ, Минск, 2016 г. – С. 131-134

2–А. Бадеев, А.А. Разработка инструментального средства для поддержки процесса оценки трудозатрат на создание IT-проектов. / А.А Бадеев, С.А. Поттосина // Научно-исследовательские публикации / Естественно-научные исследования, народное хозяйство, современные технологии и технический прогресс: сборник статей по материалам международной научно-практической конференции, Воронеж, 2016 (в печати).

РЕЗЮМЕ

Бадеев Артем Александрович

Методы и средства оценки трудозатрат для совершенствования разработки IT-проектов

Ключевые слова: методы оценки трудозатрат для разработки IT-проектов алгоритмическая модель СОСОМО II, метрики размера проектов, метод PERT критерии успешности проекта.

Цель работы: повышение точности оценки трудозатрат для создания IT-проектов с помощью анализа и разработки инструментальных средств и методов.

Полученные результаты и их новизна:

1. Выполнен сравнительный анализ существующих методов оценки стоимости и сроков IT-проектов.
2. Построена двухэтапная модель для увеличения точности оценки времени и стоимости разработки программных проектов на основе алгоритмической модели СОСОМО II.
3. Разработано инструментальное средство сбора фактических данных о проектах для исследования применимости полученной модели.

Область применения: компании, занимающиеся разработкой IT-проектов.

РЭЗЮМЭ

Бадзеяу Арцем Аляксандравіч

Метады і сродкі ацэнкі працавыдаткаў для ўдасканалення распрацоўкі IT-праектаў

Ключавыя словы: метады ацэнкі працавыдаткаў для распрацоўкі IT-праектаў алгарытмічная мадэль СОСОМО II, метрыкі памеру праектаў, метады PERT, крытэрыі паспяховасці праекта.

Мэта працы: павышэнне дакладнасці ацэнкі працавыдаткаў для стварэння IT-праектаў з дапамогай аналізу і распрацоўкі інструментальных сродкаў і метадаў.

Атрыманыя вынікі і іх новизна:

1. Выкананы параўнальны аналіз існуючых метадаў ацэнкі кошту і тэрмінаў IT-праектаў.
2. Пабудавана мадыфікаваная мадэль для павелічэння дакладнасці ацэнкі часу і кошту распрацоўкі праграмных праектаў на аснове алгарытмічнай мадэлі СОСОМО II.
3. Распрацавана інструментальны сродак збору фактычных дадзеных аб праектах для даследавання дастасавальнасці атрыманай мадэлі.

Вобласць ужывання: кампаніі, якія займаюцца распрацоўкай ІТ-праектаў.

SUMMARY

Badeev Artiom Alexandrovich

Methods and tools of evaluation effort to improve the development of IT-projects

Keywords: methods of evaluation effort for the development of IT-projects COCOMO II of algorithmic model, metric sized projects, PERT method, criteria for success of the project.

Objective: To improve the accuracy of estimation of effort to create IT-projects through the analysis and development of tools and methods.

The results and their novelty:

1. A comparative analysis of the existing methods for assessing the cost and timing of IT-projects.

2. The modified model was built to increase the accuracy of estimating the time and cost of software development projects on the basis of algorithmic model COCOMO II.

3. Development of software product to compile evidence on the projects to study the applicability of the model.

Sphere of Applications: companies involved in the development of IT-projects.