

Министерство образования и науки Российской Федерации  
Государственное образовательное учреждение высшего  
профессионального образования  
**«ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ  
УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ» (ТУСУР)**  
Кафедра автоматизации обработки информации (АОИ)

УТВЕРЖДАЮ  
Зав. кафедрой АОИ,  
д.т.н., профессор  
Ю.П. Ехлаков  
\_\_\_\_\_“    ” \_\_\_\_\_ 2011 г.

## **ТЕХНИКО-ЭКОНОМИЧЕСКОЕ ОБОСНОВАНИЕ СТОИМОСТИ ПРОГРАММНЫХ СИСТЕМ**

**Методические указания  
по выполнению экономической части  
дипломного проекта для студентов специальности 230102  
«Автоматизированные системы обработки  
информации и управления»**

Разработчики:  
д.т.н., профессор  
\_\_\_\_\_ Ю.П.Ехлаков  
ст. преп. каф. АОИ  
\_\_\_\_\_ Б.А. Рыбалов

## Оглавление

<b>1. ОБЩИЕ ПОЛОЖЕНИЯ</b> .....	4
1.1. Требования к выполнению и оформлению экономического раздела дипломного проекта .....	5
1.2. Содержание экономического раздела ДП .....	6
1.3. Описание программного продукта .....	6
1.4. Резюме .....	9
<b>2. ТЕХНИКО-ЭКОНОМИЧЕСКОЕ ОБОСНОВАНИЕ ДОГОВОРНОЙ ЦЕНЫ</b> .....	11
2.1. Общие положения .....	11
2.2. Типы и основные требования к программным системам .....	13
2.3. Методы определения размеров программной системы.....	14
2.3.1. Прямой метод определения технико-экономических показателей проекта.....	14
2.3.2. Определение технико-экономических показателей программной системы с использованием метода функциональных точек.....	19
2.3.3. Определение технико-экономических показателей проекта на основе размерности базы данных программной системы.....	35
2.4. Определение стоимости программной системы.....	37
2.4.1. Определение фонда оплаты труда на разработку и комплексные испытания программной системы.....	37
2.4.2. Определение фонда оплаты труда на проведение опытной эксплуатации программной системы.....	39
2.4.3. Структура договорной цены на программное обеспечение.....	41
2.4.4. Краткие рекомендации по подготовке раздела «Технико-экономическое обоснование договорной цены».....	44

<b>3. ОПРЕДЕЛЕНИЕ И АНАЛИЗ РЫНОЧНОЙ СТОИМОСТИ ПРИКЛАДНОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ</b> .....	48
3.1. Концепция безубыточности .....	48
3.2. Виды и составляющие издержек .....	49
3.3. Определение точки безубыточности.....	51
Литература .....	55
Приложение 1. Пример расчета договорной цены на разработку автоматизированной информационной системы.....	56
Приложение 2. Пример анализа рыночной стоимости программного продукта.....	83

## 1. ОБЩИЕ ПОЛОЖЕНИЯ

Настоящие методические указания содержат рекомендации по структуре и содержанию экономического раздела дипломных проектов студентов технических специальностей, связанных с разработкой программных продуктов (информационных технологий, систем, баз данных и т.д.) различного назначения.

Независимо от направленности и характера дипломного проекта (ДП) в экономическом разделе необходимо осуществить экономическое обоснование принимаемых технических решений, поскольку одним из важнейших критериев прогрессивности создаваемых программных продуктов являются экономические параметры.

При этом экономическая часть дипломного проекта должна содержать материал, раскрывающий рыночную компоненту, разрабатываемую в ходе дипломного проектирования программного продукта (информационной технологии, информационной системы). Предполагается, что программный продукт может быть создан как под конкретного заказчика, так и для рыночного распространения.

В первом случае (при заказном проектировании) перед разработчиком стоит сложная задача – обосновать перед заказчиком стоимость (договорную цену) и согласовать вопросы авторского сопровождения разработки.

При рыночном варианте распространения программного продукта имеют место существенные затраты на создание первой версии и низкие – на тиражирование и распространение копий. В этом случае разработчику надо заранее спланировать издержки (себестоимость) первой версии, определить рыночную цену дальнейшей продажи программного продукта.

Наконец, для успешной реализации и продвижения разработанной системы на рынке программных продуктов необходимы расчеты минимального объема продаж для обеспечения безубыточной работы организации (фирмы).

В методических указаниях приведены рекомендации по разработке основных разделов экономической части ДП.

В помощь студентам приведен список литературы, а в приложениях – примеры описания программного продукта, расчета договорной цены на создаваемое программное обеспечение и точки безубыточности фирмы, занимающейся продвижением на рынок разработанного программного продукта.

### **1.1. Требования к выполнению и оформлению экономического раздела дипломного проекта**

Экономическая часть дипломного проекта должна быть:

- связанной с основной частью ДП, являться ее логическим продолжением;
- связанной с профилем технической специальности студента;
- учитывать данные преддипломной практики, либо НИРС студента;
- достаточно современной, актуальной;
- содержать в себе новые методические положения, действующие расценки, нормативы, рыночные ориентиры.

Объем экономического раздела должен составлять не более 20-25 страниц.

Все расчеты должны сопровождаться соответствующими пояснениями, ссылками на источники получения исходных данных. Формулы должны приводиться с расшифровкой условных обозначений.

В расчетах допускается использовать текущие рыночные цены и тарифы на продукцию, работы, услуги, действующие на момент разработки проекта, курсы иностранных валют для пересчета валютной выручки и цен в иностранной валюте.

Большинство расчетов целесообразно представлять в табличной форме, оформленной в соответствии с требованиями действующих стандартов.

Основные результаты и расчеты экономического раздела ДП могут быть вынесены на специальный демонстрационный лист (плакат), либо представлены в виде компьютерных слайд-презентаций.

## 1.2. Содержание экономического раздела дипломного проекта

Структура экономической части ДП должна состоять из следующих разделов:

- описание программного продукта;
- технико-экономическое обоснование договорной цены;
- определение и анализ рыночной стоимости прикладного программного обеспечения;
- резюме.

## 1.3. Описание программного продукта

Первым разделом экономической части ДП является *описание программного продукта*.

Программное обеспечение должно быть представлено как объект продажи и поставки и ориентировано на конкретного покупателя. При этом потенциальные покупатели могут сравнивать свои требования к программному продукту с возможностями, описанными в документе, проводить анализ конкурентных преимуществ с другими (аналогичными) программными системами.

Описание программного продукта должно соответствовать базовому Российскому стандарту в области информационных технологий ГОСТ Р ИСО/МЭК 12119-2000 «Информационная технология. Пакеты программ. Требование к качеству и тестирование», в котором даны соответствующие рекомендации по структуре документа «Описание программного продукта» [1].

Вышеуказанный документ должен содержать следующие основные разделы.

### **Обозначения и указания:**

- обозначение описания продукта (описанию продукта присваивается индивидуальное обозначение - как документу, оно может иметь наименование, отличное от «описания продукта», например «Описание функциональных возможностей»,

«Информация о продукте», «Формуляр продукта», «Карта описания продукта»).

- обозначение продукта - по крайней мере должно включать наименование продукта и обозначение его версии или даты выпуска;

- поставщик (наименование и адрес по крайней мере одного поставщика);

- рабочая задача (возможности программного продукта, доступные конечному пользователю, а также предусмотренный результат функционирования рабочей системы.);

- соответствие нормативным документам;

- необходимая программно-аппаратная платформа;

- интерфейсы с другими продуктами;

- объекты поставки (определяется каждый физический компонент поставляемого продукта, в частности техническая документация и все носители данных, в обязательном порядке необходимо установить вид поставляемых программ, например исходные программы, объектные (рабочие) модули или загрузочные модули);

- ввод в действие (инсталляция) – указывается, будет ли инсталляция продукта проводиться пользователем, или нет;

- поддержка (предлагается ли поддержка при эксплуатации продукта);

- сопровождение (предлагается ли сопровождение продукта).

***Функциональные возможности:***

- пригодность и согласованность (приводится набор функций, реализуемых в данной версии, при этом необходимо указать соответствие этих функций стандартам, соглашениям, положениям законов, методическим рекомендациям);

- граничные значения (если использование продукта ограничено конкретными граничными значениями: минимальные или максимальные значения; длины ключей; максимальное число записей в файле; максимальное число критериев поиска; минимальный объем выборки и т.д.);

– защита (информация по средствам предотвращения случайного или преднамеренного несанкционированного доступа к программам и данным).

**Надежность** - набор атрибутов, относящихся к способности программного средства сохранять свой уровень качества функционирования при установленных условиях за установленный период времени.

В описание продукта должна быть включена информация по процедурам сохранения данных. Данную информацию можно привести, указав, например, возможности резервирования данных с помощью функций операционной системы. Могут быть описаны дополнительные характеристики продукта, которые обеспечивают его функциональные возможности (проверки достоверности исходных данных; защита против серьезных последствий ошибки пользователя; восстановление работоспособности программной системы при ошибках).

**Практичность** - набор атрибутов, относящихся к объему работ, требуемых для использования и индивидуальной оценки такого использования определенным или предполагаемым кругом пользователей, в том числе:

– интерфейс пользователя (тип интерфейса, например, строка команд, меню, окна, функциональная клавиша, функция подсказки);

– требуемая квалификация пользователя (конкретные знания, которые необходимо усвоить пользователю для применения соответствующего продукта: знание соответствующей технической области; знание операционной системы; знания, получаемые в результате специального обучения и т.д.);

– адаптация к требованиям пользователя (если продукт может настраиваться пользователем, то должны быть указаны инструментальные средства для проведения такой настройки и условия их применения: изменение параметров, алгоритмов вычислений; назначение функциональных клавиш и т.д.);

– защита от нарушения авторских прав (техническая защита от копирования; запрограммированные даты окончания использования продукта; интерактивные напоминания об оплате за копии и пр.).



**Эффективность** - набор атрибутов, относящихся к соотношению между уровнем качества функционирования программного средства и объемом используемых ресурсов при установленных условиях (информация о характере поведения продукта, например времени ответа и оценки производительности при реализации заданных функций при установленных условиях - для заданных конфигураций системы и профилей загрузки; информация по эффективности применения продукта и удовлетворению им потребностей пользователя).

**Сопровождаемость** подразумевает: возможность диагностики в случае отказов; определение условий для модификации, либо изменения режимов эксплуатации; возможность тестирования модифицированных частей программного продукта.

**Мобильность (переносимость)** - возможность адаптации продукта к различным условиям эксплуатации без применения дополнительных сервисов, простота внедрения в новых условиях; взаимозаменяемость с другим (аналогичным) программным продуктом.

**Примечание:** разделы *надежность, практичность, эффективность, сопровождаемость, мобильность* должны соответствовать ГОСТ 28195-89 Оценка качества программных средств. Общие положения.

Пример описания программного продукта приведен в Приложении 1.

## 1.4. Резюме

Резюме – завершающий раздел экономической части ДП, который является своеобразной визитной карточкой всего проекта.

Цель этого раздела: кратко изложить содержание проекта для обоснования его возможности, целесообразности и основных преимуществ. Подготовка краткого содержания особенно сложна. Квалифицированно это можно сделать только тогда, когда программный продукт уже разработан и уже очевидны основные мероприятия, которые необходимо выполнить по его внедрению (тиражированию) и проведены основные финансо-

во-экономические расчеты. Поэтому разрабатывать его следует после завершения всей работы.

В резюме в предельно краткой форме излагаются следующие моменты (положения):

- суть проекта;
- возможности реализации проекта в конкретных рыночных условиях;
- назначение программного продукта, его функциональные особенности;
- основные технико-эксплуатационные параметры;
- сроки выполнения проекта;
- необходимый объем финансирования;
- срок окупаемости проекта;
- потенциальные выгоды от инвестирования в проект.

Излагать резюме следует в предельно короткой, но вместе с тем не лишенной эмоциональности форме, исключая сложную техническую терминологию, упрощенные бытовые термины, экономические и другие штампы.

Следует подчеркнуть инвестиционную привлекательность, актуальность, социальную значимость предлагаемого проекта. Именно этот раздел должен вызвать интерес у потенциальных партнеров и инвесторов, убедить их в высоком уровне компетентности и профессионализме автора проекта.

Резюме должно быть кратким – не более одной страницы и должно трактоваться как самостоятельный документ, т.к. в нем содержатся все основные ключевые положения, характеризующие представленный программный продукт.

## 2. ТЕХНИКО-ЭКОНОМИЧЕСКОЕ ОБОСНОВАНИЕ ДОГОВОРНОЙ ЦЕНЫ

### 2.1. Общие положения

Под *технико-экономическим обоснованием стоимости* (договорной цены) программной системы будем понимать методику оценивания трудовых, временных и финансовых ресурсов по созданию программной системы, соответствующей требованиям заказчика.

В основу определения требуемых объемов ресурсов должны быть положены:

- совокупность бизнес-процессов, реализуемых в будущей программной системе и их относительная важность (приоритет) для заказчика;
- требования к функциональной полноте и качеству реализации каждого бизнес-процесса.

В качестве основных показателей оценки стоимости программной системы используются:

- сложность (размеры) программной системы;
- трудозатраты на разработку;
- длительность разработки программной системы в целом и ее отдельных этапов;
- численность и квалификация специалистов, привлекаемых к созданию программной системы;
- фонд оплаты труда специалистов на создание программной системы в целом и по конкретному этапу жизненного цикла;
- прочие прямые затраты и накладные расходы, связанные с созданием программной системы.

В основу определения размеров программной системы положено понятие *«сложности»*, под которой понимается количество элементов программной системы (программных компонент, файлов, входных и выходных документов) и взаимосвязей между ними.

Под термином *трудозатраты* будем понимать суммарный объем труда специалистов для создания программного продукта.

В качестве универсального измерителя трудозатрат используется показатель – *человеко-месяц*. Каждый человеко-месяц содержит 160 человеко-часов (четыре недели, пять рабочих дней, восьмичасовой рабочий день).

Длительность разработки и численность специалистов определяются на основе трудозатрат и нормативной производительности труда программиста, выражаемой в количестве строк кода, создаваемых программистом в единицу времени. При этом выделяются следующие этапы жизненного цикла программной системы:

- анализ предметной области и разработка требований к программной системе;
- проектирование;
- программирование;
- тестирование и комплексные испытания;
- опытная эксплуатация.

В реализации проекта на каждом этапе принимает участие три группы специалистов:

- руководитель проекта, системные аналитики;
- непосредственные разработчики программных систем и специалисты по комплексированию;
- технический персонал, обеспечивающий тестирование, документирование и опытную эксплуатацию программного обеспечения.

Фонд оплаты труда на реализацию проекта определяется исходя из производительности труда специалиста и согласованной базовой месячной заработной платы.

Все нормативы и другие статистические данные, используемые в методиках технико-экономического обоснования стоимости основываются на статистических данных, обобщающих зарубежный и российский опыт разработки программных систем [2].

## 2.2. Типы и основные требования к программным системам

По уровню сложности все множество программных систем (ПС) следует разбить на три типа.

К *первому типу* относятся:

- комплексные программные системы (КПС) и технологии, отдельные части которых реализованы на различных платформах;
- территориально-распределенные программные системы и технологии;
- системы автоматизированного либо автоматического управления, функционирующие в режиме реального времени.

*Второй тип* составляют программные информационно-справочные системы (ИСС), обеспечивающие информационную поддержку основных бизнес-процессов организации с большим количеством типов исходной информации.

К *третьему типу* относятся инженерные и научно-технические пакеты программ (ППП) и технологий, характеризующихся четко заданным алгоритмом обработки и малыми объемами исходных данных.

При формировании требований к программной системе необходимо использование стандарта ISO/IEC 9126-1:2001 [3].

С учетом этого, системы первого и второго типа должны удовлетворять следующим требованиям:

- архитектура системы должна соответствовать текущим и перспективным целям и стратегическим, функциональным задачам создаваемой системы;
- в структуре и компонентах следует предусматривать обеспечение максимально возможной сохранности капитальных вложений заказчика в аппаратные и программные средства, а также в базы данных при длительном развитии, сопровождении и модернизации системы;
- для обеспечения перспективы развития системы должны быть предусмотрены возможность интеграции компонентов и мобильность ПС на различные аппаратные и операционные платформы на основе концепции и стандартов Открытых систем;

- архитектура информационной системы должна быть достаточно гибкой и допускать простое, без коренных структурных изменений, развитие и наращивание функций и ресурсов системы в соответствии с расширением сфер и задач ее применения;
- необходимо обеспечить эффективное использование ресурсов системы и минимизировать интегральные затраты на обработку данных в типовых режимах ее функционирования с учетом текущих эксплуатационных затрат и капитальных вложений в создание ПС;
- следует обеспечить комфортный, максимально упрощенный доступ конечных пользователей к управлению и результатам функционирования системы на основе современных графических средств и наглядных пользовательских интерфейсов.

## **2.3. Методы определения размеров программной системы**

### **2.3.1. Прямой метод определения технико-экономических показателей проекта**

Прямой метод определения технико-экономических параметров программной системы основан на использовании *метода экспертных оценок*.

Будущую программную систему следует декомпозировать до уровня элементарных компонент, а для оценки размеров каждого из компонент использовать либо внешних экспертов, имеющих опыт разработки подобных систем и готовые прототипы, либо, в качестве экспертов могут выступать специалисты разработчика и заказчика [4].

При декомпозиции целесообразно использовать следующие термины и определения, изложенные в [5] (рис. 2.1):

- ***интегрированная программная система*** – совокупность двух и более программных систем, в которых функционирование одной из них зависит от результатов функционирования другой;

– **программная система** – совокупность программных комплексов, реализующих множество бизнес-процессов организации;

– **программный комплекс** – совокупность программных компонент, реализующих конкретный бизнес-процесс;

– **сложная программная компонента** – совокупность программных кодов, реализующих сложную функцию бизнес-процесса;

– **программная компонента** – совокупность программных кодов, реализующих элементарную функцию бизнес-процесса.

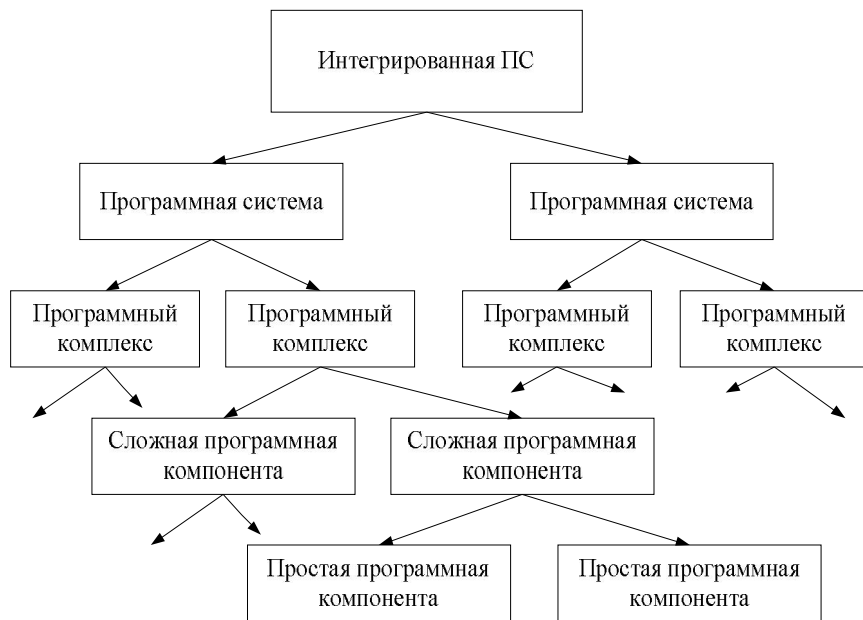


Рис. 2.1. Структура программной системы

Размеры программной системы определяются в виде количества строк исходного кода в терминах Lines of code-LOC [2].

При оценке количества строк исходного кода следует учитывать следующие положения:

– строка исходного кода содержит только один оператор;

- определение (описание) исходных данных учитывается один раз;
- не учитываются строки, содержащие комментарии и отладочные операторы;
- учитывается каждая инициализация, вызов, либо включение макроса в качестве исходного кода.

Каждый из экспертов должен дать оптимистическую ( $o$ ), пессимистическую ( $p$ ) и реалистическую ( $b$ ) оценки (табл. 2.1).

Средняя оценка по бета-распределению определяется путем умножения реалистической оценки на 4, добавлением оптимистической и пессимистической оценок и делением полученного результата на 6 (формула 2.1).

Таблица 2.1

Бланк экспертного оценивания размерности программной системы

Состав программной системы	Оценки		
	Оптимистическая	Реалистическая	Пессимистическая
1. Программная система			
1.1. Программный комплекс			
1.1.1. Программная компонента 1			
1.1.2. Программная компонента 2			
.....			

$$r^{k}_{ij} = (o_{ij} + 4b_{ij} + p_{ij}) / 6 \quad (2.1)$$

После оценивания всех компонент на каждом уровне, начиная с нижнего, происходит суммирование результатов измерения по принципу «снизу -вверх».

$$R = \sum_{k=1}^q \sum_{i=1}^n \sum_{j=1}^{m_i} r^{k}_{ij} / q, \quad (2.2)$$



где  $q$  - количество экспертов,  $m_i$  - количество программных компонент на  $i$ -ом уровне,  $N$  - число уровней.

Очевидно, что эффективность оценивания может быть существенно повышена при наличии прототипов будущей программной системы. В этом случае эксперту предлагается оценить необходимость степени модернизации имеющегося прототипа, увеличив, либо сократив исходный размер программной компоненты на некоторое количество строк.

### **Определение трудозатрат, длительности реализации проекта и средней численности разработчиков**

Оценка трудозатрат, длительности и средней численности разработчиков при реализации проекта основывается на согласовании между разработчиком и заказчиком производительности труда программиста -  $P$ .

В [4] приводятся следующие среднестатистические оценки производительности труда программиста:

- при разработке программных систем первого класса сложности (КПС) преимущественно на языке ассемблер – 60-80 строк/чел.-месяц;

- при разработке программных систем второго класса сложности (ИСС) на языках высокого уровня – 250-260 строк/чел.-месяц.

В таблице 2.2. представлены статистические показатели производительности, рекомендуемые в **базовой** модели Constructive Cost Model [2].

**CO**nstructive **CO**st **MO**del (COCOMO – модель издержек разработки) – это алгоритмическая модель оценки стоимости разработки программного обеспечения, разработанная Барри Бозмом (Barry Boehm). Модель использует простую формулу регрессии с параметрами, определенными из данных, собранных по ряду проектов.

COCOMO была впервые опубликована в 1981 году в книге Бозма «Экономика разработки программного обеспечения» в качестве модели для оценки трудоемкости, себестоимости и плана-графика для проектов по разработке ПО.

Она использовала исследование 63 проектов в аэрокосмической компании TRW, в которой Бозм был директором отдела исследований программного обеспечения и технологий.

В исследовании проекты классифицировались по размеру в зависимости от количества строк кода (от 2 до 100 тысяч), а также по языку программирования (от ассемблеров до высокоуровневого языка PL/1).

В 1997 году была разработана модель СОСОМО II, которая стала наследником первоначальной модели и более подходящей для оценивания современных проектов разработки ПО. Она предоставляет более полную поддержку современных процессов разработки ПО и построена на обновленной базе проектов.

В целом, СОСОМО состоит из иерархии трех последовательно детализируемых и уточняемых форм.

Первый уровень – **Базовый**, хорош для быстрых ранних оценок стоимости разработки ПО, однако не принимает во внимание различия в аппаратных ограничениях, качестве и опыте персонала, а также использованию современных техник и средств разработки и других факторов, которые невозможно учесть на ранних стадиях разработки.

**Средний** уровень СОСОМО учитывает эти факторы, тогда как **Детальный** уровень дополнительно учитывает влияние отдельных фаз проекта на его общую стоимость.

Средний уровень (СОСОМО Model 2: Intermediate) – рассчитывает трудоемкость разработки как функцию от размера программы и множества «факторов стоимости», включающих субъективные оценки характеристик продукта, проекта, персонала и аппаратного обеспечения. Детальный уровень (СОСОМО Model 3: Advanced/Detailed) – включает в себя все характеристики среднего уровня с оценкой влияния данных характеристик на каждый этап процесса разработки ПО.

Приведенные ниже нормативы отражают не только трудоемкость непосредственного написания текстов программ, но и процессы комплексирования и испытания всего программного комплекса. С учетом вышеизложенного, трудозатраты на разработку системы могут быть определены следующим образом:

$$T = R / P \quad (2.3)$$

## Нормативы трудоемкости разработки программ

Класс сложности ПС	Размеры и трудоёмкость создания ПС	
	простые – до 30 тыс. строк	сложные – до 500 тыс. строк
Первый тип – КПС	до 140 строк/чел.-месяц	до 80 строк/чел.-месяц
Второй тип – ИСС	до 220 строк/чел.-месяц	до 160 строк/чел.-месяц

Длительность разработки (Д) может быть задана директивно заказчиком, исходя из реальных потребностей его бизнеса и наличия финансовых ресурсов. В этом случае средняя численность специалистов, которые должны быть привлечены к реализации программной системы определяется по формуле:

$$Z = T / Д \quad (2.4)$$

Прямой метод целесообразно использовать на ранних стадиях проектирования при разработке концепции и технического задания на будущую программную систему. Это позволит разработчику и заказчику определить трудоемкость реализации каждого бизнес-процесса, проранжировать их в соответствии с пожеланиями заказчика, соизмерить финансовые возможности заказчика и сроки реализации проекта.

### 2.3.2. Определение технико-экономических показателей программной системы с использованием метода функциональных точек

*Метод функциональных точек* (Function point – FP) основывается на том, что размеры программной системы оцениваются в терминах количества и сложности бизнес-процессов (функций), реализуемых в данном программном коде [2].

*Функциональная точка* – это комбинация свойств программного обеспечения:

- интенсивности использования ввода и вывода внешних данных;
- взаимодействия системы с пользователем;
- внешних интерфейсов;
- файлов, используемых системой.

Будущая система с использованием методологии структурного анализа и проектирования описывается в виде многоуровневой графической модели, представленной в виде совокупности взаимосвязанных функциональных диаграмм (пользовательских бизнес-процессов).

Каждый из бизнес-процессов включает в себя входные и выходные данные, преобразования и внешние интерфейсы.

Процедура оценивания размеров программной системы соотносится с одним из пользовательских бизнес-процессов и состоит из следующей последовательности этапов (ввод, вывод, опросы, структуры данных, интерфейсы):

- выделение множества бизнес-процессов;
- подсчет количества функциональных точек бизнес-процесса в разрезе каждой категории;
- определение весовых коэффициентов сложности каждой функции;
- учет факторов и требований среды разработки программной системы;
- вычисление интегральных показателей сложности;
- вычисление итогового количества функциональных точек;
- определение размеров программного комплекса в показателях LOC;
- определение размеров программной системы в целом.

При определении количества функций каждого бизнес-процесса следует руководствоваться следующими требованиями:

- учитываются только сложные функции, перечисленные в техническом задании;
- при декомпозиции сложной функции учитываются все логические преобразования с данными.

Расчет количества функциональных точек по каждому бизнес-процессу рекомендуется сводить в следующую таблицу (табл. 2.3)

Таблица 2.3  
Рабочая таблица определения количества функциональных точек

№ п.п.	Категории простых функций	Простые	Средние	Сложные	Количество функциональных точек
1	Количество выводов	$\alpha_{11} \cdot X_{11}$	$\alpha_{12} \cdot X_{12}$	$\alpha_{13} \cdot X_{13}$	$X_1$
2	Количество вводов	$\alpha_{21} \cdot X_{21}$	$\alpha_{22} \cdot X_{22}$	$\alpha_{23} \cdot X_{23}$	$X_2$
3	Количество опросов вывода	$\alpha_{31} \cdot X_{31}$	$\alpha_{32} \cdot X_{32}$	$\alpha_{33} \cdot X_{33}$	$X_3$
4	Количество опросов ввода	$\alpha_{41} \cdot X_{41}$	$\alpha_{42} \cdot X_{42}$	$\alpha_{43} \cdot X_{43}$	$X_4$
5	Количество файлов	$\alpha_{51} \cdot X_{51}$	$\alpha_{52} \cdot X_{52}$	$\alpha_{53} \cdot X_{53}$	$X_5$
6	Количество интерфейсов	$\alpha_{61} \cdot X_{61}$	$\alpha_{62} \cdot X_{62}$	$\alpha_{63} \cdot X_{63}$	$X_6$
Общее количество функциональных точек					$\sum_{i=1}^{i=6} X_i$

Примечание:  $\alpha_{ij}$  — весовой коэффициент сложности  $i$ -й функции  $j$ -й категории сложности;  $X_{ij}$  — количество элементов данных  $i$ -й функции  $j$ -й категории сложности.

**Определение количества выводов.** Под выводами будем понимать следующие единицы информации, получаемые на выходе рассматриваемого бизнес-процесса:

– файлы, продуцируемые в данном бизнес-процессе для передачи другим бизнес-процессам, либо за пределы программной системы;

– единицы деловой информации, предназначенные для конечных пользователей, оформленные в виде экранных форм, либо бумажных документов.

Каждый из выводов, в зависимости от количества файлов, используемых при формировании выходов, рекомендуется отнести к одной из категорий сложности: простой, средний, сложный. В табл. 2.4 представлены весовые коэффициенты сложности выводов.

Таблица 2.4

Весовые коэффициенты сложности выводов

Количество файлов \ Количество элементов данных	от 1 до 5	от 6 до 19	20 и более
1	$\alpha_{11} = 4$	$\alpha_{12} = 4$	$\alpha_{13} = 5$
2-3	$\alpha_{11} = 4$	$\alpha_{12} = 5$	$\alpha_{13} = 7$
4 и более	$\alpha_{11} = 5$	$\alpha_{12} = 7$	$\alpha_{13} = 7$

**Определение количества вводов.** Под вводами будем понимать следующие единицы информации, поступающие на вход рассматриваемого бизнес-процесса:

– входные файлы, полученные из других бизнес-процессов, либо других программных систем;

– уникальная единица деловой информации, вводимая конечным пользователем.

По аналогии с выводом все вводы также рекомендуется разделять на простые, средние и сложные (см. табл. 2.5).

Таблица 2.5

## Весовые коэффициенты сложности ввода

Количество элементов данных \ Количество файлов	от 1 до 5	от 6 до 19	20 и более
1	$\alpha_{21} = 4$	$\alpha_{22} = 4$	$\alpha_{23} = 5$
2-3	$\alpha_{21} = 4$	$\alpha_{22} = 5$	$\alpha_{23} = 7$
4 и более	$\alpha_{21} = 5$	$\alpha_{22} = 7$	$\alpha_{23} = 7$

**Определение количества опросов ввода, вывода.** Под опросами будем понимать следующие действия, исполняемые программной системой в рассматриваемом бизнес-процессе:

- обращение к внешним процедурам, оформленным в виде специфических команд или запросов, генерируемых извне и выполняемых программной системой;
- выполнение процедур, обеспечивающих непосредственный доступ к базе данных и выполняющих выборку с помощью простых ключей в режиме реального времени, но не выполняющих функции обновления.

Рекомендуется учитывать каждую уникальную единицу опроса, если:

- формат опроса отличается от формата ввода, вывода;
- формат опроса совпадает с форматом ввода, вывода, но требует дополнительной логики обработки.

При определении количества опросов не следует учитывать запросы к базам данных, использующие несколько ключей и выполняющие определенные операции, либо вычисления с последующим оформлением выводов.

Все опросы также рекомендуется разделять на простые, средние и сложные. В таблицах 2.6 и 2.7 приведены рекомендации по выбору весовых коэффициентов.

Таблица 2.6

## Весовые коэффициенты сложности опросов вывода

Количество элементов данных \ Количество файлов	от 1 до 5	от 6 до 19	20 и более
1	$\alpha_{31} = 4$	$\alpha_{32} = 4$	$\alpha_{33} = 5$
2-3	$\alpha_{31} = 4$	$\alpha_{32} = 5$	$\alpha_{33} = 7$
4 и более	$\alpha_{31} = 5$	$\alpha_{32} = 7$	$\alpha_{33} = 7$

Таблица 2.7

## Весовые коэффициенты сложности опросов ввода

Количество элементов данных \ Количество файлов	от 1 до 5	от 6 до 19	20 и более
1	$\alpha_{41} = 3$	$\alpha_{42} = 3$	$\alpha_{43} = 4$
2-3	$\alpha_{41} = 3$	$\alpha_{42} = 4$	$\alpha_{43} = 6$
4 и более	$\alpha_{41} = 4$	$\alpha_{42} = 6$	$\alpha_{43} = 6$

**Определение количества файлов.** Под файлами будем понимать следующие единицы информации, используемые программной системой в рассматриваемом бизнес-процессе:



- внутренние логические файлы программной системы;
- структуры данных, представляющие собой первичную логическую группу пользовательских данных, которые постоянно находятся внутри границ программной системы;
- внешние файлы, доступные пользователям с помощью ввода, вывода, опросов, либо интерфейсов.

Весовые коэффициенты оценки сложности файлов, в зависимости от количества взаимосвязей между таблицами представлены в табл. 2.8.

Таблица 2.8

Весовые коэффициенты сложности структурных данных (файлов)

Количество элементов данных / Количество логических взаимосвязей	от 1 до 19	от 20 до 50	более 51
Одна логическая запись типа формат/взаимосвязь	$\alpha_{51} = 7$	$\alpha_{52} = 7$	$\alpha_{53} = 7$
От 2 до 5 записей	$\alpha_{51} = 7$	$\alpha_{52} = 10$	$\alpha_{53} = 10$
более 6 записей	$\alpha_{51} = 10$	$\alpha_{52} = 15$	$\alpha_{53} = 10$

**Определение количества интерфейсов.** Под интерфейсами, используемыми рассматриваемым бизнес-процессом, будем понимать:

- файлы, сгенерированные другими программными системами и использующиеся в данной ПС;
- потоки данных, хранящихся за пределами программной системы, но используемых при управлении вычислительным процессом в любом направлении пересылки;
- структуры данных, использующихся в нескольких программных системах.

Весовые коэффициенты оценки сложности интерфейсов представлены в табл. 2.9.

Таблица 2.9

Весовые коэффициенты сложности интерфейсов

Количество элементов данных / Количество логических взаимосвязей	от 1 до 19	от 20 до 50	более 51
Одна логическая запись типа формат/взаимосвязь	$\alpha_{61} = 5$	$\alpha_{62} = 5$	$\alpha_{63} = 7$
От 2 до 5 записей	$\alpha_{61} = 7$	$\alpha_{62} = 7$	$\alpha_{63} = 10$
более 6 записей	$\alpha_{61} = 7$	$\alpha_{62} = 10$	$\alpha_{63} = 10$

*Общее количество функциональных точек* определяется по следующей формуле:

$$F = \sum_{i=1}^6 \sum_{j=1}^3 \alpha_{ij} \cdot x_{ij} \quad (2.5)$$

**Учет факторов и требований среды разработки**

Сложность предметной области и качества создаваемого программного обеспечения зависит от среды разработки приложений и требований конечных пользователей.

Влияние этих факторов на размеры программного обеспечения оценивается по ряду показателей (см. табл. 2.10). Каждый из показателей, в свою очередь, оценивается по пятибалльной шкале. Рекомендуемая шкала измерения показателей приведена в таблице 2.11.

## Факторы среды разработки

№ п.п.	Факторы среды	Варианты
1	Каналы передачи данных	Входные и выходные данные передаются по локальной сети, магистральным каналам связи, по Internet
2	Распределенные вычисления	Пользовательские приложения используют данные, хранящиеся в едином хранилище, получают из других систем
3	Производительность системы	Скорость передачи данных и время отклика системы существенно для данного бизнес-процесса
4	Конфигурирование	Пользовательские приложения выполняются с применением интенсивно используемой, ограниченной, либо наполненной конфигурации
5	Частота транзакций	Использование приложений приводит к высокому сетевому трафику, экранные формы динамично изменяются, наблюдается высокая концентрация выходных форм, графика
6	Интерактивная обработка	Частота использования пользовательских приложений, участие пользователя при выполнении запросов
7	Пользовательский интерфейс	Значимость взаимодействия конечных пользователей с программной системой (необходимость дополнительного учета человеческого фактора)

№ п.п.	Факторы среды	Варианты
8	Интерактивное обновление базы данных	Степень динамики обновления данных
9	Сложность обработки запросов	Уровень сложности алгоритмов обработки, количество транзакций, требования к безопасности и надежности
10	Сложность инсталляции (установки) системы	Наличие автоинсталляции, качество технической документации
11	Сложность эксплуатации системы	Наличие процедур запуска, резервирования, копирования, восстановления при ошибках, уровень (сложность) участия пользователей в этих процессах
12	Степень распределенности системы	Количество и удаленность пользовательских приложений
13	Гибкость изменения функций	Модульная реализация, наличие настроек, уровень поддержки со стороны пользователей, возможность изменения запросов

Таблица 2.11

## Шкала измерения факторов внешней среды

Влияние фактора не существенно	Влияние фактора существенно	Влияние фактора очень существенно
[0 — 1]	[2 — 3]	[4 — 5]

Уровень влияния факторов внешней среды рекомендуется определять по следующей формуле [2]:

$$Z = 0.65 + (0.01 \cdot N), \quad (2.6)$$

где  $N$  – суммарное значение весовых коэффициентов факторов внешней среды.

Уточненное количество функциональных точек с учетом факторов внешней среды определяется по формуле [2]:

$$R(F) = F \cdot Z \quad (2.7)$$

В качестве базового показателя количества строк исходного кода следует использовать число операторов языка ассемблер.

Варианты преобразования размеров программы, оцененной по этому измерителю в размеры программы кода, написанного на других языках программирования и наоборот, представлены в таблице 2.12 [9].

Таблица 2.12.

Соответствие среднего числа строк текста программы на языке Ассемблер одной строке других языков программирования

№ п.п.	Язык программирования	Ассемблер (LOC)	Показатель LOC на 1 функциональную точку
1	2	3	4
1	Basic Assembler	1	320
2	Macro Assembler	1,5	213
3	C	3	107
4	C#	5,4	59
5	Perl	5,6	57
6	JavaScript	5,8	55
7	Java	6	53
8	C++	6	53
9	Visual Basic	6,2	52
10	Excel	7	46
11	HTML	7,6	42
12	Access	8,5	38
13	Oracle Developer/2000	10,6	30
14	SQL	10,6	30
15	Oracle, Sybase	11	29
16	Delphi	11	29
17	Smalltalk	16,8	19
18	Web Scripts	21,3	15

Размерность программного обеспечения для конкретного языка программирования определяется с учетом нормативов, представленных в таблице 2.12 по формуле:

$$R(LOC) = R(F) \cdot LOC, \quad (2.8)$$

где  $LOC$  – среднее количество операторов конкретного языка программирования, требующегося для реализации одной функциональной точки (табл. 2.12, столбец 4).

Итоговая размерность программной системы определяется путем суммирования величины каждого  $R(LOC)$  бизнес-процесса.

### **Определение трудозатрат, длительности и средней численности специалистов на основе базовой конструктивной модели трудозатрат – COCOMO**

В основу *оценки трудозатрат* положена степенная функция следующего вида [2]:

$$T = A \cdot R^E(KLOC)/12, \quad (2.9)$$

где  $T$  – трудозатраты, выраженные в человеко-месяцах;

$R(KLOC)$  – размерность программной системы, выраженная в тысячах строк кода.

Первый множитель  $A$  является доминирующим, он прямо пропорционален размерности программного обеспечения  $R$  и отражает линейную зависимость роста трудозатрат от размерности.

Второй множитель  $R^E$  отражает тот факт, что при увеличении размерности программной системы возрастает относительная трудоемкость разработки каждой строки программного кода за счет увеличения количества взаимосвязей между компонентами.

Значения параметров  $A$  и  $E$ , полученные путем статистической обработки данных по результатам реализации множества проектов, представлены в табл. 2.13 [4].

Оценки по СОСОМО получены в результате обработки статистических данных по 160 реальным зарубежным проектам, а оценки по ПРОМЕТЕЮ – результат обобщения статистики – по 250 отечественным.

Таблица 2.13

Коэффициенты математической модели оценки трудозатрат в зависимости от типа программных систем

Тип программной системы	СОСОМО		ПРОМЕТЕЙ	
	$A$	$E$	$A$	$E$
Первый тип - КПС	3,6	1,2	10	1,21
Второй тип - ИСС	3	1,12	6,1	1,17
Третий тип - ППП	2,4	1,05	—	—

**Длительность разработки** программной системы определяет общие сроки разработки ПС, начиная от разработки технического задания (требований) на систему и завершая этапом проведения комплексных испытаний и может быть задана директивно заказчиком, исходя из реальных потребностей его бизнеса и наличия финансовых ресурсов.

**Средняя численность сотрудников**, занятых в проекте, определяется по формуле:

$$Z = T / Д \quad (2.10)$$

Расчет длительности и, соответственно, численности специалистов для разработки программного обеспечения может быть произведен также из среднестатистической производительности труда программиста.

## Определение трудозатрат, длительности и средней численности специалистов на основе модифицированной модели – СОСОМО II

В модифицированной модели СОСОМО II (средний уровень) при определении трудоемкости *учитываются дополнительно пять групп факторов*, влияющих на технико-экономические показатели проекта:

- масштабность проекта;
- требования к показателям качества программного обеспечения;
- квалификация коллектива разработчиков;
- характеристики технологической среды разработки;
- характеристики программно-аппаратной среды разработки.

В табл. 2.14 представлен перечень конкретных показателей по каждой из вышеперечисленных групп и их максимальные значения. Конкретные значения показателей определяются совместно представителями разработчика и заказчика. При значении показателя равным *единице* считается, что соответствующий фактор не влияет на трудоемкость разработки программной системы.

Таблица 2.14  
Состав и максимальные значения факторов модифицированной модели СОСОМО II

Наименование фактора	Обозначение	Макс. значение
1	2	3
<b><i>1. Масштабные факторы</i></b>		
1.1. Новизна проекта	F1	1,33
1.2. Согласованность с требованиями и интерфейсами	F2	1,26



1	2	3
1.3. Управление рисками и архитектурой проекта	F3	1,39
1.4. Слаженность работы коллектива	F4	1,29
1.5. Технологическая зрелость обеспечения разработки	F5	1,43
<b><i>2. Требования и характеристики объекта разработки</i></b>		
2.1. Надежность функционирования	M1	1,54
2.2. Размер базы данных	M2	1,42
2.3. Сложность функций и структуры	M3	2,38
2.4. Требование повторного использования компонентов	M4	1,31
2.5. Полнота и соответствие документации проекта	M5	1,52
<b><i>3. Характеристики коллектива специалистов</i></b>		
3.1. Квалификация аналитиков	M9	2,00
3.2. Квалификация программистов	M10	1,76
3.3. Стабильность коллектива	M11	1,51
3.4. Опыт работы по тематике проекта	M12	1,51
3.5. Опыт работы в инструментальной среде	M13	1,40
3.6. Опыт работы с языками программирования	M14	1,43
<b><i>4. Технологическая среда разработки</i></b>		
4.1. Уровень инструментальной поддержки проекта	M15	1,50

Продолжение табл. 2.14

1	2	3
4.2. Необходимость распределенной разработки проекта	M16	1,53
4.3. Ограничения длительности разработки проекта	M17	1,43
<b>5. Аппаратно-вычислительная среда разработки</b>		
5.1. Ограниченность времени исполнения программ	M6	1,63
5.2. Ограниченность доступной оперативной памяти	M7	1,46
5.3. Изменчивость виртуальной среды разработки проекта	M8	1,49

Оценка трудоемкости разработки программной системы по модифицированной модели производится по выражению:

$$T = A \times R^E (KLOC) \times \left( \prod_{i=1}^n M(i) \right)^{\frac{1}{n}} / 12, \quad (2.11)$$

где  $A = 2,94$      $B = 0,91$      $E = B + 0,01 \sum_{J=1}^5$

Данная модель предусматривает возможность прогнозирования длительности разработки проекта на основе регрессионной модели:

$$Д = G \cdot T^H, \quad (2.12)$$

где  $G = 3,67$ ,  $K = 0,28$ ,  $B = 0,91$ ,  $H = K + 0,02 * (E - B)$

Средняя численность сотрудников составляет:

$$Z = T / Д \quad (2.13)$$

Использование модифицированной модели СОСОМО II позволяет в среднем на 5 – 10% повысить точность определения технико-экономических показателей проекта.

### 2.3.3. Определение технико-экономических показателей проекта на основе размерности базы данных программной системы

Размерность программной системы определяется количеством объектов, атрибутов и их взаимосвязями на объектных диаграммах бизнес-процессов [6].

**Атрибут** – простейший элемент базы данных информационной модели, содержащей одну из характеристик предметной области и вводимой либо непосредственно пользователем, либо заносимый в базу из справочников и классификаторов.

**Объект** – элемент базы данных, формируемый из атрибутов и содержащий информацию о реальном процессе, явлении, предмете.

Размерность программного обеспечения определяется по следующей формуле:

$$R = 2N \cdot 5K_1 \cdot 10M, \quad (2.14)$$

где  $N$  – количество объектов (таблиц) предметной области, количество связей между таблицами неограниченно и определяется структурой базы данных;

$K_1$  – суммарное количество взаимосвязей между объектами;

$M$  – суммарное количество атрибутов предметной области, **приходящихся на один объект.**

Количество связей между атрибутами определяется количеством источников формирования атрибутивной информации.

Нормализованной величиной при создании программной системы является количество формируемых атрибутов, входящих в электронные таблицы посредством установленных связей.

При значениях  $N$ ,  $K_1$ ,  $M$ , равных единице, величина, выражающая их количество равна 100. Трудозатраты разработки определяются по формуле 2.15 на основе статистических нормативов трудоемкости, приведенных в табл. 2.15 [6].

$$T = 0.01 \cdot R \cdot \theta, \quad (2.15)$$

где  $\theta$  – норматив трудоемкости разработки программной системы.

Таблица 2.15  
Нормативы трудоемкости разработки программной системы

Категория сложности	Значение норматива $\theta$ (чел./месяц)
* Разработка прикладных программ (пользовательских приложений) с использованием стандартных средств СУБД ** Количество прикладных программ (не более 3-х) *** Размерность базы данных (до 90 тыс. полей)	0,00566
* Разработка прикладных программ (пользовательских приложений) с использованием стандартных пакетов прикладных программ ** Количество прикладных программ (от 3-х до 10-ти) *** Размерность базы данных (от 90 тыс. до 200 тыс. полей)	0,00808
* Разработка прикладных программ (пользовательских приложений) с использованием языков высокого уровня ** Количество прикладных программ (не ограничено) *** Размерность базы данных (от 200 тыс. до 500 тыс. полей)	0,01537

Длительность разработки может быть задана директивно заказчиком, исходя из реальных потребностей его бизнеса и наличия финансовых ресурсов, при этом средняя численность специалистов определяется по формуле:

$$Z = T / Д \quad (2.16)$$

Данный метод рекомендуется использовать при разработке программных систем на базе стандартных СУБД при:

- больших размерностях базы данных, формируемой из различных источников;
- наличии специализированных компонент, реализующих произвольные информационные запросы пользователей.

## 2.4. Определение стоимости программной системы

### 2.4.1. Определение фонда оплаты труда на разработку и комплексные испытания программной системы

В основу определения фонда оплаты труда положены:

- длительность реализации каждого этапа жизненного цикла проекта;
- количество и качественный состав специалистов, привлекаемых на каждом этапе проекта;
- базовая месячная ставка специалиста-программиста.

В табл. 2.16, 2.17 приведены среднестатистические распределения первых двух величин по основным этапам жизненного цикла создания программных систем [4].

Таблица 2.16

Распределение трудозатрат и длительности по основным этапам жизненного цикла создания программных систем

№ п.п.	Этапы жизненного цикла	Трудозатраты $\alpha$ (%)	Длительность $\beta$ (%)
1	Анализ предметной области и разработка требований	10	10
2	Проектирование	22	30
3	Программирование	40,5	35
4	Тестирование и комплексные испытания	27,5	25

Используя эти распределения, по выражению 2.17 можно рассчитать среднюю численность сотрудников, занятых на каждом из этапов создания программной системы.

$$Z_i = \alpha_i T / \beta_i D, \quad i=1,4 \quad (2.17)$$

В табл. 2.17 дано относительное распределение численности специалистов на каждом из четырех этапов жизненного цикла создания программной системы.

Таблица 2.17

Распределение специалистов по этапам  
жизненного цикла программной системы

№ п.п.	Этапы жизненного цикла	Типы специалистов (%)		
		Аналитики	Программисты	Технические специалисты
1	Анализ предметной области и разработка требований	40	20	40
2	Проектирование	35	35	30
3	Программирование	10	65	25
4	Тестирование и комплексные испытания	15	60	25

Численность каждого типа специалистов на каждом из этапов жизненного цикла создания программной системы определяется по выражению:

$$Z_{ij} = P_{ij} \cdot Z_i, \quad i=1,4, j=1,3, \quad (2.18)$$

где  $P_{ij}$  – относительная доля (%) специалистов  $j$ -го типа, привлекаемых для реализации проекта на  $i$ -ом этапе.

Фонд заработной платы для реализации  $i$ -го этапа проекта определяется по формуле:

$$S_i = \sum_{j=1}^3 Z_{ij} \cdot D_i \cdot S_j, \quad i = 1,4, \quad (2.19)$$

где  $D_i$  – длительность  $i$ -го этапа проекта;

$S_j$  – месячный фонд заработной платы специалиста  $j$ -го типа.

В основу определения  $S_j$  может быть положена месячная базовая ставка программиста, размер которой может быть принят как одна из альтернатив: базовая ставка программиста заказчика; базовая ставка программиста разработчика; среднерыночная базовая ставка программиста в данном регионе.

В качестве ориентира для согласования базовой ставки может быть принята стоимость разработки одной строки исходного кода программы 20-50 \$ US [4] и принятой производительности труда (раздел 2.3.1.).

При этом месячная ставка специалиста-программиста соотносится к месячной ставке системного аналитика как 1:1,3, а к ставке технического специалиста – как 1:0,7.

Общий фонд заработной платы на реализацию проекта определяется по формуле:

$$S = \sum_{i=1}^4 S_i \quad (2.20)$$

#### **2.4.2. Определение фонда оплаты труда на проведение опытной эксплуатации программной системы**

На этапе опытной эксплуатации программной системы в соответствии с ГОСТ 34.603-92 [7] производится заполнение нормативной базы, справочников, классификаторов, эксплуатация программного обеспечения в регламентном режиме, доработка программного обеспечения и рабочей документации в случае несоответствия текущей версии системы требованиям технического задания. Срок опытной эксплуатации оговаривается в техническом задании.

Численность сотрудников, привлекаемых к опытной эксплуатации определяется по формуле:

$$Z_{on} = t_{on} \cdot N, \quad (2.21)$$

где  $t_{on}$  — срок опытной эксплуатации;

$N$  – норматив трудоемкости при проведении опытной эксплуатации.

Нормативы трудоемкости опытной эксплуатации программной системы определяются исходя из среднестатистических нормативов [6] согласно таблице 2.18.

Таблица 2.18

## Нормативы трудоемкости опытной эксплуатации

Измеритель – 1 сеанс работы

Категория сложности	Значение норматива $N$ (чел./месяц)
* Количество пользователей (не более 10) ** Количество сеансов работы с системой в течение года (от 10 до 150)	0,00354
* Количество пользователей (не более 20) ** Количество сеансов работы с системой в течение года (от 150 до 650)	0,00504
* Количество пользователей (не ограничено) ** Количество сеансов работы с системой в течение года (от 650 до 6000)	0,0095

Опытная эксплуатация проводится группой внедрения разработчика с привлечением, в случае необходимости, программистов, при этом относительный норматив заработной платы специалиста составляет 0,85.

Фонд заработной платы на проведение опытной эксплуатации определяется по формуле:

$$S_{on} = Z_{on} \cdot t_{on} \cdot S_n \cdot 0,85, \quad (2.22)$$

где  $S_n$  — месячная базовая ставка программиста.



### 2.4.3. Структура договорной цены на программное обеспечение

Структура договорной цены представлена в табл. 2.19 и состоит из следующих разделов:

1) **прямые расходы:**

- фонд оплаты труда;
- страховые взносы в Пенсионный фонд (ПФР), Фонд социального страхования (ФСС) и Федеральный фонд обязательного медицинского страхования (ФОМС);
- увеличение стоимости основных средств – по данной статье приобретаются все основные средства организации (в т.ч. средства вычислительной техники);
- амортизация средств ВТ стоимостью от 40000 рублей и выше, используемых разработчиком при реализации проекта, исходя из установленного в организации срока полезного использования техники (25-36 месяцев);
- командировочные расходы – при необходимости (определяются на договорной основе), а для бюджетных учреждений – на основании Приказов Минфина РФ;
- оплата коммунальных услуг, размер которых определяется исходя из численности специалистов и нормативов площади на 1 рабочее место;
- прочие расходы (расходные материалы, услуги связи и т.д.) определяются на договорной основе – по согласованию с заказчиком, при этом услуги связи (телефоны, Интернет) определяются действующими расценками со стороны региональных отделений ОАО «Ростелеком» и провайдеров сети Интернет;

2) **накладные расходы** организации-разработчика (расходы на АУП, охрану, обслуживающий персонал и т.д.), размер которых подтверждается соответствующими документами;

3) **фонд развития производства** устанавливается как определенный процент от прямых расходов по согласованию между разработчиком и заказчиком (образуется за счет прибыли и определяется инновационной политикой предприятия);

4) **налог на добавленную стоимость** определяется Налоговым кодексом РФ и формой организации предприятия-

разработчика. Следует при этом учесть, что согласно налоговому кодексу РФ не подлежат налогообложению (освобождаются от НДС) следующие виды работ:

– выполнение научно-исследовательских и опытно-конструкторских работ за счет средств бюджетов, а также средств Российского фонда фундаментальных исследований, Российского фонда технологического развития и образуемых для этих целей в соответствии с законодательством РФ внебюджетных фондов министерств, ведомств, ассоциаций;

– выполнение НИР и ОКР учреждениями образования и науки на основе хозяйственных договоров.

Таблица 2.19

## Структура договорной цены

№ п.п.	Наименование статей расходов	Методика определения
1	Оплата труда непосредственных исполнителей	Определяется по методике, изложенной в разделе 2.4. настоящего документа
2	Страховые взносы в ПФР, ФСС, Федеральный ФОМС.	Определяется Налоговым кодексом РФ и формой организации предприятия-разработчика
3	Увеличение стоимости основных средств	Средства, связанные с осуществлением уставной деятельности хозяйствующего субъекта, т.е. с производством продукции и другими видами деятельности (здания, сооружения, оборудование, вычислительная техника, машины и др.)
4	Амортизация средств вычислительной техники стоимостью от 40000 рублей и выше	Определяется исходя из установленного в организации срока полезного использования техники (25-36 месяцев)

Продолжение табл. 2.19

№ п.п.	Наименование статей расходов	Методика определения
5	Прочие выплаты (командировки в части суточных)	Определяется приказами Минфина РФ – для бюджетных организаций, на договорной основе – для организаций других форм собственности
6	Транспортные услуги (командировки в части оплаты транспортных расходов)	Определяется на договорной основе
7	Прочие услуги (командировки в части проживания, оплата услуг сторонних организаций по НИР, ОКР и технологическим работам, договора подряда с ЕСН)	Определяется на договорной основе
8	Прочие расходы (расходные материалы, услуги связи и т.д.)	Определяются на договорной основе – по согласованию с заказчиком
9	Накладные расходы организации-разработчика (расходы на АУП, охрану, обслуживающий персонал и т.д.)	Подтверждаются соответствующими документами организации-разработчика
10	Фонд развития производства	Устанавливается как определенный процент от прямых расходов по согласованию между разработчиком и заказчиком (образуется за счет прибыли и определяется инновационной политикой предприятия)
11	Налог на добавленную стоимость	Определяется Налоговым кодексом РФ и формой организации предприятия-разработчика.

#### 2.4.4. Краткие рекомендации по подготовке раздела «Технико-экономическое обоснование договорной цены»

Обобщая вышеизложенный материал, приведем краткую последовательность действий (сценарий) при подготовке технико-экономического обоснования договорной цены программной системы.

1. Составляется описание программного продукта (пример – в приложении 1).

2. Определяется тип системы и сложность системы (раздел 2.1.-2.2.).

3. Фиксируется язык программирования.

4. Выбираются (задаются преподавателем-консультантом) два из трех методов определения технико-экономических параметров программной системы, которые необходимо применить при подготовке экономической части ДП:

– прямой метод определения технико-экономических показателей проекта (метод экспертных оценок);

– метод функциональных точек;

– метод определения ТЭП проекта на основе размерности базы данных ПС.

5. Задается плановый срок на разработку системы, установленный заказчиком.

6. При работе *с прямым методом* определения ТЭП проекта:

1) производится анализ объекта автоматизации (декомпозиция бизнес-процессов) и определяется состав программных комплексов и компонент ПС;

2) определяется (задается) количество экспертов, определяющих размерность программной системы;

3) эксперты заполняют бланки экспертного оценивания размерности программной системы (табл. 2.1.);

4) с помощью формул 2.1. и 2.2. вычисляется размерность программной системы;

5) проводится оценка трудозатрат и средней численности разработчиков при реализации проекта (табл. 2.2, формула 2.3.).

7. При работе с методом определения ТЭП *на основе размерности базы данных программной системы:*

1) производится анализ объекта автоматизации (ER-моделирование) и строится физическая (логическая, концептуальная) модель базы данных ПС;

2) определяются параметры полученной БД, необходимые для расчета размерности ПС (количество объектов, взаимосвязей и атрибутов);

3) определяется размерность программного обеспечения (в данном случае – базы данных) – формула 2.15;

4) определяются трудозатраты и средняя численность разработчиков при реализации проекта, при этом трудозатраты определяются по формуле 2.16 на основе статистических нормативов трудоемкости разработки программной системы, приведенных в табл. 2.15 (раздел 2.3.3).

8. При определении ТЭП с помощью *метода функциональных точек:*

1) выделяется множество бизнес-процессов;

2) производится подсчет количества функциональных точек бизнес-процесса в разрезе каждой категории функций (простые/средние/сложные) для заданной последовательности этапов (ввод, вывод, опросы, структуры данных, интерфейсы) – таблицы 2.3-2.9, раздел 2.3.2 – формула 2.5.;

3) производится учет факторов и требований среды разработки программной системы (табл. 2.10, 2.11) – формула 2.6;

4) по формуле 2.7. рассчитывается уточненное количество функциональных точек с учетом факторов внешней среды;

5) по формуле 2.8. определяется размерность программного обеспечения для конкретного языка программирования с учетом нормативов, представленных в таблице 2.12;

6) производится определение трудозатрат при заданной длительности и средней численности специалистов на основе

базовой конструктивной модели трудозатрат – СОСОМО: трудозатраты определяются по формуле 2.9., таблица 2.13;

7) по формуле 2.10. определяется средняя численность сотрудников, занятых в проекте.

**Примечание.** В настоящих методических указаниях дано также определение трудозатрат, длительности и средней численности специалистов на основе модифицированной модели – СОСОМО II, позволяющей производить более качественную оценку ТЭП проекта на основе учета пяти дополнительных групп факторов, влияющих на технико-экономические показатели (масштабные факторы проекта, требования и характеристики объекта разработки, характеристики коллектива специалистов, технологическая и аппаратно-вычислительная среда разработки).

Использование модифицированной модели позволяет в среднем на 5–10% повысить точность определения ТЭП проекта.

Вместе с тем, при написании экономической части дипломного проекта использование модифицированной модели – СОСОМО II **не является обязательным требованием.**

Таким образом, расчет технико-экономических показателей проекта на основе вышеприведенных методов, позволяет получить исходные показатели для следующего шага – определения стоимости создания программной системы.

**9. Стоимость (договорная цена)** на создание программной системы определяется исходя из фонда оплаты труда на разработку, комплексные испытания и опытную эксплуатацию ПС:

1) выбираются исходные данные для расчета договорной цены, полученные с использованием двух из трех методов определения ТЭП – трудоемкость и длительность разработки;

2) проводится распределение трудоемкости и длительности разработки по основным этапам жизненного цикла создания ПС – заполняется таблица 2.16. с использованием формулы 2.17, при этом выполняется расчет средней численности сотрудников, занятых на каждом из этапов создания ПС и длительности каждого этапа;

3) далее происходит распределение численности специалистов по этапам жизненного цикла системы по выражению 2.18 на базе статистического распределения из таблицы 2.17 и заполняется соответствующая таблица (см. приложение 1);

4) определяется фонд заработной платы для реализации каждого этапа проекта (формула 2.19) на основе базовой месячной ставки специалиста-программиста (например, рыночная базовая ставка программиста в данном регионе) и заполняется таблица распределения фонда заработной платы по этапам жизненного цикла программной системы (см. приложение 1);

5) вычисляется общий фонд заработной платы на разработку и комплексные испытания ПС;

6) определяется фонд оплаты труда на проведение опытной эксплуатации, для чего используются выражения 2.21 и 2.22, а также нормативы трудоемкости опытной эксплуатации из таблицы 2.18;

7) суммируется фонд оплаты труда на разработку ПС с фондом оплаты труда на проведение опытной эксплуатации и получается общий фонд заработной платы на разработку и внедрение системы.

10. Заключительным этапом определения стоимости программной системы является **подготовка структуры договорной цены** (сметы затрат на разработку и внедрение), которая достаточно подробно изложена в разделе 2.4.3. и приводится в виде таблицы 2.19. (см. также пример – приложение 1).

Пример расчета договорной цены на разработку и внедрение автоматизированной информационной системы приведен в Приложении 1.

### 3. ОПРЕДЕЛЕНИЕ И АНАЛИЗ РЫНОЧНОЙ СТОИМОСТИ ПРИКЛАДНОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

#### 3.1. Концепция безубыточности

История успеха любой компании измеряется величиной полученной прибыли и изменением ее динамики. Для успешного ведения бизнеса необходимо не только просчитывать, сколько компания заработает при достижении запланированного объема продаж, но и четко представлять, какой минимальный объем продаж необходим для обеспечения безубыточной работы.

Задача стоит в определении того объема продаж, ниже которого предприятие будет терять деньги, выше которого – зарабатывать. Этот минимально допустимый объем продаж, который покрывает все затраты на изготовление продукции, не принося при этом ни прибыли, ни убытков, получил название **точка безубыточности** (она же – точка равновесия, она же – break-event point).

С точки зрения экономической теории безубыточность – есть нормальное состояние фирмы на современном конкурентном рынке, находящемся в состоянии долгосрочного равновесия. При этом мы принимаем в рассмотрение экономическую прибыль, то есть то определение прибыли, при котором в расходы фирмы включается среднерыночная ставка дохода на инвестированный капитал, а также нормальный доход предприятия.

При таких допущениях определение безубыточности звучит следующим образом: **«Точка безубыточности - это такой объем продаж продукции фирмы, при котором выручка от продаж полностью покрывает все расходы на производство продукции, в том числе среднерыночный процент на собственный капитал фирмы и нормальный предпринимательский доход».**

В этом случае, если фирма имеет бухгалтерскую прибыль, то есть сальдо доходов от продаж и денежных затрат на производство проданной продукции положительно и при этом прибыль фирмы меньше, чем среднерыночный процент на собст-



венный капитал фирмы, то она может не достичь точки безубыточности в смысле экономической прибыли. Следовательно, существуют более выгодные способы использования капитала, которые позволяют получить более высокую прибыль.

Таким образом, понятие точки безубыточности является одновременно и неким критерием эффективности деятельности фирмы. Фирма, не достигающая точки безубыточности действует неэффективно с точки зрения сложившейся рыночной конъюнктуры. Однако этот факт сам по себе не служит однозначной причиной для прекращения существования фирмы. Для того, чтобы ответить на этот вопрос необходимо детально исследовать структуру издержек фирмы.

Из курса микроэкономической теории известно, что начиная с некоторого объема выпуска кривая переменных издержек будет являться возрастающей, а кривая предельного дохода - убывающей. Для целей максимизации прибыли основным является соотношение между предельным доходом и предельными издержками при увеличении выпуска на одну единицу. Очевидно, что в случае, когда предельные издержки меньше предельного дохода, увеличение выпуска повлечет за собой увеличение дохода фирмы. В случае же когда предельные издержки больше предельного дохода, к увеличению предельного дохода фирмы приведет уменьшение выпуска продукции.

Таким образом, можно сформулировать **критерии точки безубыточности**, при которой достигается максимальная прибыль: максимальная прибыль фирмы достигается при таком объеме производства, при котором предельный доход равен предельным издержкам.

## 3.2. Виды и составляющие издержек

В рамках определения уровня безубыточности все затраты разделяют на две группы: **условно-переменные** и **условно-постоянные**.

**FC (Fixed Cost)** – **постоянные (фиксированные) издержки** – денежные издержки, в целом не изменяющиеся в зависимости от изменения объема выпускаемой продукции.

К основным составляющим фиксированных издержек можно отнести следующие виды затрат:

- зарплата руководящего состава и административного персонала (служащих);
- аренда и лизинг;
- амортизация зданий и оборудования;
- налоги (подходные, на платежную ведомость, и т.д.);
- платежи за внешние услуги;
- плата за телефон и коммунальные услуги;
- плата за страхование;
- уплата процентов по кредитам;
- общие административные расходы.

Вместе с тем, при достижении определенного объема производства постоянные издержки могут быть увеличены на определенную величину.

*VC (Variable Cost) – переменные издержки* – издержки, меняющиеся пропорционально объёму производства. Переменными издержками являются:

- затраты на сырьё и труд основных производственных рабочих (заработная плата);
- комиссионные отчисления торговым агентам;
- затраты на приобретение тары, упаковочного материала;
- транспортные расходы.

Деление затрат на постоянные и переменные даёт возможность финансовому менеджеру:

- определить сроки окупаемости затрат;
- определить запас финансовой прочности предприятия;
- рассчитать оптимальную величину прибыли предприятия.

Кроме того, иногда выделяют и смешанные издержки (постоянно-переменные затраты), которые включают в себя элементы как постоянных, так и переменных расходов: оплата топлива, почтовые расходы, телефон, отопление, затраты на текущий ремонт оборудования, электроэнергию и т. д.

Постоянно-переменные затраты увеличиваются при увеличении объёма производства, но не пропорционально его росту, а

в другой пропорции. При конкретных расчётах необходимо выделять в составе смешанных издержек постоянную и переменную части, причисляя их к соответствующему виду затрат. Текущие затраты, обеспечивающие жизнедеятельность предприятия – это постоянная составляющая постоянно-переменных затрат, а затраты, связанные с развитием производства, – это переменная составляющая.

*ТС (Total Cost) – полные издержки* – сумма постоянных и переменных издержек.

*Маржа на продажах С (Contribution margin)* – выручка за вычетом переменных издержек, в других терминах – средства, получаемые в распоряжение на операциях производства (закупки) и продажи партии товара. Причем эти средства не обязательно вкладывать на покрытие постоянных издержек. Предприятие будет рентабельным при условии, если маржинальный доход будет выше постоянных затрат.

*Маржинальные издержки МС (Marginal Cost)* – издержки производства на выпуск дополнительной продукции, при условии, что постоянные (фиксированные) издержки фирмой уже погашены.

*Невозвратные издержки SC (Sunk Cost)* – одноразовые невозвратные издержки.

*Упущенная выгода ОС (Opportunity Cost)* – доход, который можно было бы получить, но он не получен из-за отказа от использования предоставленной возможности.

В качестве примера в приложении 2 рассмотрен вид деятельности фирмы, специализирующейся на рынке прикладного программного обеспечения: от заказа на разработку программного продукта, – до его тиражирования.

### 3.3. Определение точки безубыточности

Основным методом определения точки безубыточности является *CVP-анализ (Cast Value Profit - затраты, объем, прибыль)* [8], основанный на анализе соотношений затрат и объемов выпуска и прибыли.

Чистая прибыль фирмы определяется как разница между выручкой и переменными и постоянными издержками.

$$P = s \cdot x - (a + b \cdot x) = (s - b) \cdot x - a, \quad (3.1)$$

где  $P$  – прибыль фирмы;

$x$  – объем выпуска продукции;

$S$  – договорная цена продажи единицы продукции;

$a$  – величина фиксированных расходов;

$b$  – величина переменных издержек на единицу продукции.

В этом случае объем выпуска при котором достигается точка безубыточности (прибыль фирмы равняется нулю) определяется по следующей формуле:

$$x_0 = \frac{a}{(s - b)} \quad (3.2)$$

В случае, если объем рынка определен (задан) можно определить рыночную цену единицы продукции при нулевом уровне прибыли:

$$s_0 = \frac{a + bx_0}{(x_0)} \quad (3.3)$$

Если фирма хочет получить дополнительную прибыль (сверх нормативной) и рыночная цена известна, то объем продаж при заданном уровне прибыли  $P_0$  и рыночной цене  $S_0$  можно определить по следующей формуле:

$$x_0 = \frac{P_0 + a}{(s_0 - b)} \quad (3.4)$$

В некоторых случаях в качестве исходных данных при определении необходимого объема продаж используется величина предельного (маржинального) дохода от реализации единицы продукции как разницы между ценой и переменными издержками:  $m_0 = s_0 - b$ .

В этом случае выражение 3.4 выглядит следующим образом:

$$x_0 = \frac{P_0 + a}{m_0} \quad (3.5)$$

Графическая интерпретация определения и анализа точки безубыточности представлена на рисунке 3.1.

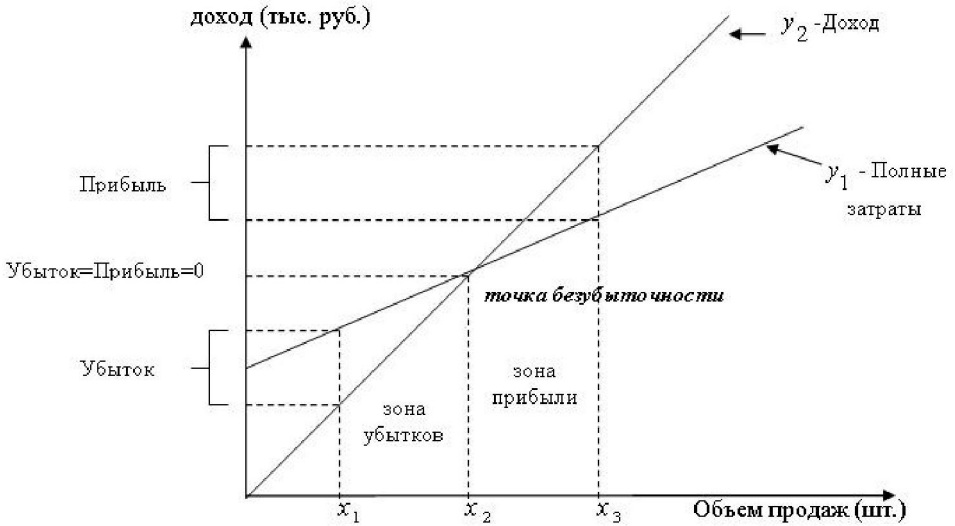


Рис. 3.1. Графическая интерпретация определения и анализа точки безубыточности

Следует отметить, что корректное использование методов CVP-анализа верно лишь **в ограниченном диапазоне объемов выпуска**. Ограниченность проистекает, прежде всего, из-за того, что при достаточно большом объеме выпуска продукции перестают быть верными многие предпосылки, лежащие в основе CVP-анализа, например, неизменный характер и величина постоянных расходов, и т.д. В этой связи в литературе выделяют следующие условия применимости CVP-анализа при определении точки безубыточности:

- изменение общих затрат и выручки жестко определено и линейно в пределах области релевантности;

- цены на продукцию, материалы и услуги, используемые в производстве, неизменны;
- производительность труда специалистов постоянна;
- ассортимент продукции и объемы выпуска каждого вида неизменны – отсутствуют структурные сдвиги в производстве;
- на конец анализируемого периода объем производства равен объему продаж, либо изменения начального и конечного уровня запасов незначительны.

Пример расчета точки безубыточности приведен в приложении 2.

В приведенном примере в число фиксированных расходов входит выплата ежемесячных отчислений и процентов по банковскому кредиту, который предположительно берется фирмой для реализации проекта на разработку программного продукта.

В этом случае, необходимое количество продаж программного продукта  $x_n$  для полного возмещения затрат составит:

$$x_n = \frac{K * a}{(s - b)} \quad (3.6)$$

где  $K$  – срок банковского кредита (соответствует сроку разработки системы, установленному Заказчиком);

$S$  – договорная цена продажи единицы продукции;

$a$  – величина фиксированных расходов;

$b$  – величина переменных издержек на единицу продукции.

Наконец, срок окупаемости проекта при количестве продаж, необходимых для полного возмещения затрат  $x_n$  в точке безубыточности  $x_0$  определяется по формуле:

$$C_{ok} = \frac{x_n}{x_0} \quad (3.7)$$

## Литература

1. ГОСТ Р ИСО/МЭК 12119-2000 «Информационная технология. Пакеты программ. Требование к качеству и тестирование».
2. Роберт Т. Фатрелл, Дональд Ф. Шафер, Линда И. Шафер. Управление программными проектами. Достижение оптимального качества при минимуме затрат.: Персона.—М.: Издательский дом «Вильямс», 2004г. – 1136 с.
3. Международный стандарт ISO/IEC 9126-1:2001. Программирование. Качество продукта. Часть 1. Модель качества.
4. Липаев В.В. Техничко-экономическое обоснование проектов сложных программных систем.— М.: СИНТЕГ, 2004, – 284 с.
5. ГОСТ 34.003-90 Информационная технология. Автоматизированные системы. Термины и определения.
6. Сборник временных норм на работы по ведению Государственного мониторинга геологической среды, информационной деятельности, цифровому картографированию. Министерство природных ресурсов Российской Федерации.—М. 2000г. – 24 с.
7. ГОСТ 34.603-92 Информационная технология. Виды испытаний автоматизированных систем.
8. Баймухамбетова С.С. Финансовый менеджмент. М., 2004.
9. The QSM Project Database. Function Point Languages Table, Version 4.0 (November, 2009).  
<http://www.qsm.com/resources/function-point-languages-table>.

## Приложение 1

### Пример расчета договорной цены на разработку автоматизированной информационной системы

#### Основные задачи

1. Подготовить описание программного продукта – разработанной в ходе преддипломной практики и дипломного проектирования программной системы.
2. Не менее, чем двумя из трех предложенных в методических указаниях методов рассчитать технико-экономические показатели (ТЭП) проекта.
3. Выбрав из полученных вариантов наиболее выгодный, с коммерческой точки зрения, определить стоимость (договорную цену) разработки и внедрения программной системы.

#### Карта описания программного продукта

**АИС «ТиС»** - «Автоматизированная информационная система «Торговля и склад», версия 1.1.

**Поставщик АИС «ТиС»** – лаборатория объектно-ориентированного моделирования информационных систем кафедры автоматизации обработки информации ТУСУР. Адрес: 634045, г. Томск, ул. Вершинина, 74, корп. ФЭТ ТУСУРа, ауд. 405. тел. (3822) 41-44-70.

АИС «ТиС» предназначена для информационной поддержки процессов управления складами и торговыми представительствами заказчика.

АИС «ТиС» соответствует ГОСТ Р ИСО/МЭК 12119-2000 «Информационная технология. Пакеты программ. Требование к качеству и тестирование».

#### **Программно-технические средства:**

- программное обеспечение АИС «ТиС» разработано в архитектуре «клиент-сервер», средой разработки является язык программирования Delphi, СУБД – Firebird 1.5, под управлением операционной системы MS Windows 2000/XP;



- техническое обеспечение:

- Сервер: CPU Intel Pentium IV 2,6 GHz/1024 Mb, HDD 160Gb SATA, сеть 1000/100 Mb/s, ОС MS Windows 2000/XP, СУБД – Firebird 1.5;

- Клиент: CPU Intel Celeron 1,3 GHz/128 Mb, HDD 120 Gb, ОС MS Windows 2000/ XP, СУБД – Firebird 1.5.

**В комплект поставки входят:**

1) оптический носитель (компакт-диск), содержащий дистрибутив системы в загрузочных модулях и инсталляционные пакеты, в том числе:

- клиентская часть, представляющая собой программное обеспечение для работы с базой данных АИС;

- серверная часть, представляющая собой файл данных БД АИС «ТиС»;

- инсталляционные пакеты СУБД Firebird 1.5.1. под Linux и Windows, являющейся клоном Interbase и разрабатываемой как продукт Open Source. Данная СУБД может использоваться бесплатно.

Информацию о СУБД Firebird 1.5.1. можно получить на сайте производителя <http://www.ibphoenix.com>. АИС «ТиС» на стадии комплексных испытаний и опытной эксплуатации протестирована в работе с данной СУБД и является полностью совместимой с ней.

2) техническая документация в составе:

- АИС «ТиС». Техническое задание.

- АИС «ТиС». Общее описание системы.

- АИС «ТиС». Руководство пользователя.

- АИС «ТиС». Руководство программиста.

Документация выполнена в соответствии с Государственный стандарт РФ ГОСТ Р ИСО/МЭК 15910—2002. Информационная технология. Процесс создания документации пользователя программного средства.

АИС «ТиС» снабжена программой инсталляции, обеспечивающей установку и настройку программного обеспечения в локальной сети (сервер) и на отдельные компьютеры (рабочие станции).

Для установки системы необходимо вставить установочный компакт-диск в привод CD-ROM, самостоятельно запустить файл setup.exe и следовать в дальнейшем руководству по инсталляции (входит в «Руководство пользователя»), включающем также инструкцию по установке инсталляционных пакетов СУБД Firebird 1.5.1.

Предусмотрена техническая поддержка и сопровождение системы на основании соответствующего соглашения Поставщика и Покупателя.

#### **Функциональные возможности системы.**

АИС «ТиС» реализована как распределенная система управления процессами закупки и реализации продукции в составе комплекса взаимосвязанных АРМ («Администратор», «Прием продукции», «Прием заказов», «Прием платежей», «Исполнение заказов»), объединенных единым форматом представления данных, идеологией обработки информации и ориентированных на использование баз данных общего пользования, единую техническую базу и операционную среду. Каждый АРМ рассчитан на эксплуатацию конкретным подразделением организации.

Для оптимизации процессов сбора и обработки информации, минимизации затрат на ввод и хранение данных, повышения актуальности, достоверности и сопоставимости данных различных информационных систем в АИС «ТиС» реализованы следующие технологические требования:

- централизованная база данных, с возможностью подключения к ней с удаленных терминалов посредством сети Интернет;
- однократный ввод данных в систему с возможностью дальнейшего их использования в функционально связанных подсистемах;
- включение в состав хранимой БД информации только тех данных, для которых существуют надежные тракты актуализации;
- своевременная актуализация данных в базе в зависимости от вида хранимой информации.

АИС «ТиС» выполняет следующие функции:

- внесение, поиск и обработка информации о поставщиках и поставляемых продуктах;
- внесение, поиск и обработка информации о заказчиках и заказах;
- внесение, поиск и обработка информации о платежах;
- проведение товарно-финансовых операций при исполнении заказов;
- администрирование АИС, в том числе ведение словарей, справочников, классификаторов, списков пользователей.
- разграничение доступа к данным и функциям АИС на основе системы ролей и привилегий.

АИС «ТиС» обеспечивает круглосуточную бесперебойную работу при отсутствии помех со стороны аппаратного обеспечения и операционной системы.

Предусмотрены механизмы автоматического резервного копирования данных, возможна настройка периодичности выполнения резервного копирования.

Предусмотрена возможность восстановления системы по команде администратора на основе последних результатов резервного копирования.

В случае возникновения системного сбоя (отключение питания, неполадки в аппаратном обеспечении, в работе операционной системы) АИС «ТиС» выполняет самостоятельное восстановление работоспособности после перезагрузки системы.

Взаимодействие с пользователем организовано посредством графического пользовательского интерфейса в общепринятой форме. Предусмотрены возможности настройки пользовательского интерфейса под требования пользователя: возможна настройка системы меню, внешнего вида приложения. Настройки хранятся на сервере БД и действуют для конкретного пользователя из любой точки доступа к АИС.

При работы с системой предполагается, что пользователи знакомы в общих чертах с операционной системой MS Windows 9x/XP и владеют базовыми навыками работы в ней, при этом они должны обладать простейшим опытом работы с окнами (Windows), приемами работы с меню и уметь пользоваться

стандартными диалогами, а также обладать минимальными навыками работы в сети Интернет.

Для облегчения работы пользователя в АИС «ТиС» предусмотрена контекстная справочная система, содержащая информацию о порядке работы с конкретными функциями АИС.

Эффективность использования системы заключается в:

- сокращении времени выполнения расчетов по текущим торговым операциям;
- сокращении времени построения оперативных и регламентных отчетов;
- повышении качества и контроля исполнительской дисциплины сотрудников организации.

Время отклика системы на запрос пользователя при условии, что аппаратная конфигурация удовлетворяет рекомендуемым системным требованиям, не превышает:

- 1 сек. – при выполнении оперативных внесений в БД;
- 10 сек. – при построении оперативных отчетов;
- 5 мин. – при построении регламентных отчетов.

Построение регламентных отчетов происходит в фоновом режиме (пользователь может продолжать работу с системой в процессе построения регламентного отчета).

Время восстановления работоспособности системы после перезагрузки не превышает 15 минут, при условии, что аппаратная конфигурация удовлетворяет рекомендуемым системным требованиям.

В АИС «ТиС» предусмотрена возможность уведомления администратора по электронной почте о возникновении сбоев в работе системы. Специальные возможности изменения режимов функционирования системы не предусмотрены. Специальные возможности модификации системы не предусмотрены.

Возможность перенесения АИС «ТиС» на другую аппаратно-программную платформу не предусмотрена.

Надежность, практичность, эффективность, сопровождаемость и мобильность системы соответствует основным положениям ГОСТ 28195-89 Оценка качества программных средств. Общие положения.

# I. Прямой метод определения технико-экономических показателей проекта (метод экспертных оценок)

## Исходные данные.

Тип системы: программно-информационная

Сложность системы: простая

Язык программирования – Delphi

Плановый срок разработки системы, установленный заказчиком – 12 месяцев.

## Анализ объекта автоматизации

Декомпозируем программную систему до уровня элементарных компонент и для оценки размеров каждой из них используем либо внешних экспертов, либо специалистов разработчика и заказчика (пример декомпозиции – рис. 1.).

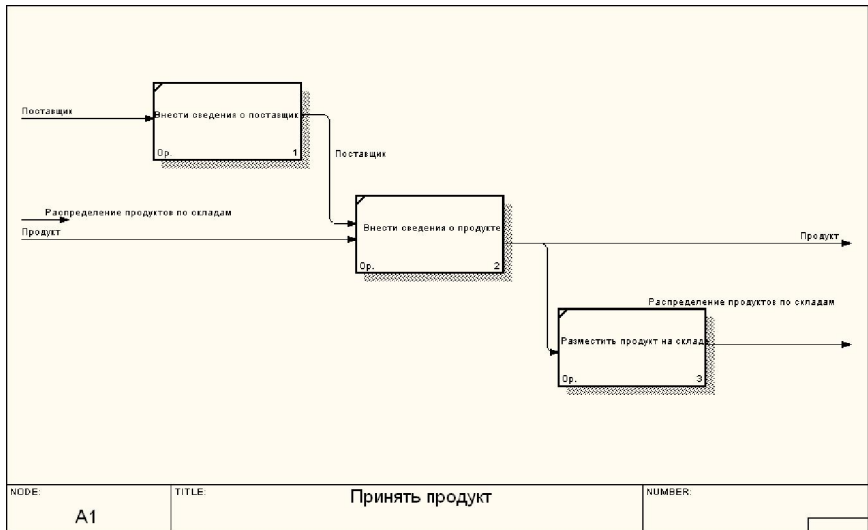


Рис. 1. Декомпозиция бизнес-процесса «Прием продукта на склад»

В результате проведенного анализа получаем структуру программных комплексов и компонент АИС «ТиС» (рис. 2.).

Для упрощения расчетов примем допущение, что работу по оцениванию системы проводит только один эксперт – представитель разработчика.

Эксперт должен дать оптимистическую ( $o$ ), пессимистическую ( $p$ ) и реалистическую ( $b$ ) оценки строк исходного кода для каждой программной компоненты. Средняя оценка по бета-распределению определяется по формуле 2.1. (раздел 2.3.1.):

$$r^k_{ij} = (o_{ij} + 4b_{ij} + p_{ij}) / 6$$

где  $o$  – оптимистическая оценка;  
 $b$  – реалистическая оценка;  
 $p$  – пессимистическая оценка;  
 $k$  – номер участвующего в оценке эксперта;  
 $i$  – номер уровня;  
 $j$  – номер программной компоненты.

Эксперт оценивает размерность программной системы, заполняя таблицу оценивания размерности ПС (табл. 1).

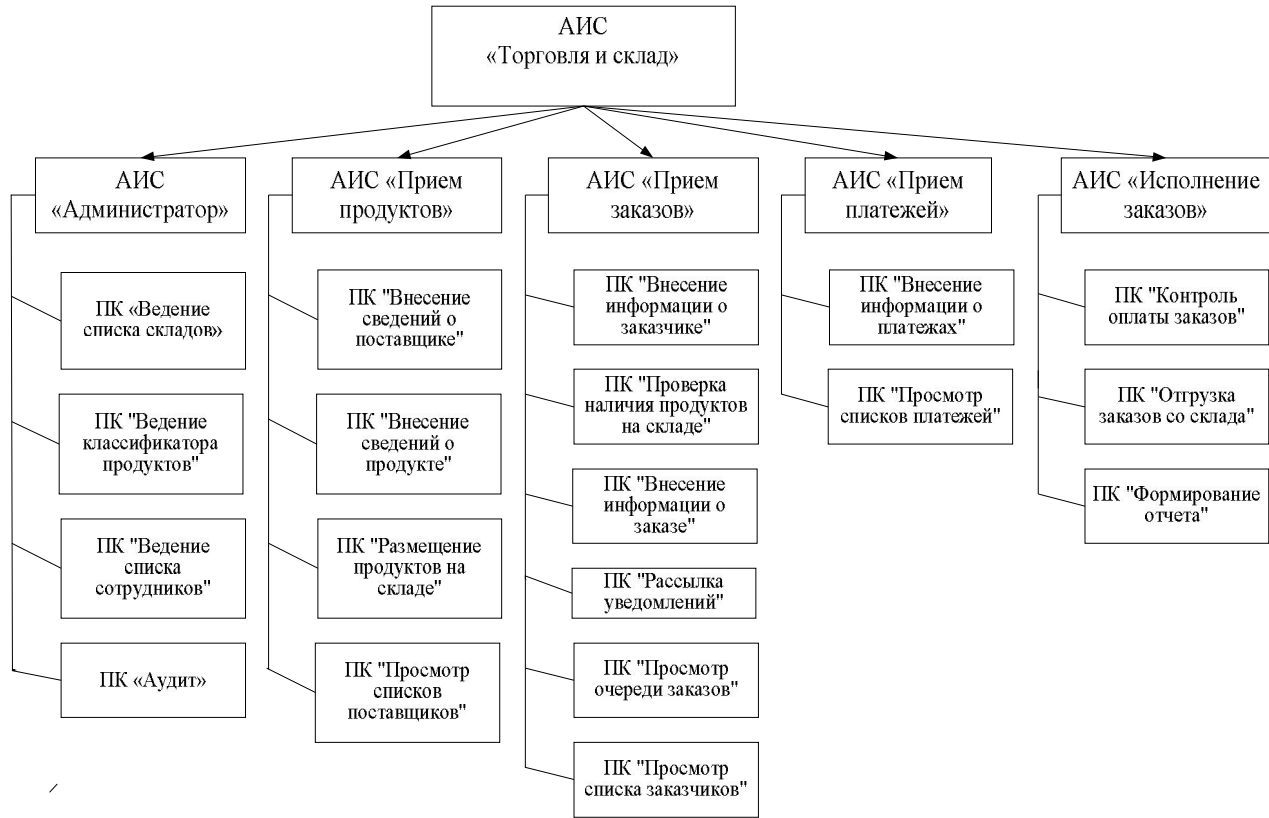


Рис. 2. Структура программной системы «Торговля и склад»

Таблица 1

Бланк экспертного  
оценивания размерности программной системы

Программные комплексы и компоненты АИС «ТиС»	Оценка			
	П	Р	О	Средняя оценка
<b>АИС «Администратор»</b>				
ПК "Ведение списка складов"	96	64	32	64
ПК "Ведение классификатора продук- тов"	32	32	32	32
ПК "Ведение списка сотрудников"	32	32	32	32
ПК "Аудит"	320	160	120	180
<b>АИС «Прием продуктов»</b>				
ПК "Внесение сведений о поставщике"	640	320	128	341
ПК "Внесение сведений о продукте"	640	320	128	341
ПК "Размещение продуктов на складе"	128	128	128	128
ПК "Просмотр списков поставщиков"	128	128	128	128
<b>АИС «Прием заказов»</b>				
ПК "Внесение информации о заказчике"	640	320	128	341
ПК "Проверка наличия продуктов на складе"	640	320	128	341
ПК "Внесение информации о заказе"	640	320	128	341
ПК "Рассылка уведомлений"	320	320	128	288
ПК "Просмотр очереди заказов"	32	320	128	240
ПК "Просмотр списка заказчиков"	32	320	128	240
<b>АИС «Прием платежей»</b>				
ПК "Внесение информации о платежах"	512	320	64	309
ПК "Просмотр списков платежей"	640	512	64	459
<b>АИС «Исполнение заказов»</b>				
ПК "Контроль оплаты заказов"	128	64	64	75
ПК "Отгрузка заказов со склада"	128	64	64	75
ПК "Формирование отчета"	320	512	128	416
<b>ИТОГО</b>				<b>4372</b>

Сокращения: П – пессимистическая оценка; Р – реалистиче-  
ская оценка; О – оптимистическая оценка.



После оценивания всех компонент на каждом уровне, суммируются результаты измерения по принципу «снизу-вверх» (формула 2.2.) и определяется размерность системы:

$$R = \sum_{k=1}^q \sum_{i=1}^n \sum_{j=1}^{m_i} r^k_{ij} / q,$$

где  $q$  - количество экспертов,

$m_i$  - количество программных компонент на  $i$ -ом уровне.

$$R = 4372 \text{ строк кода}$$

Следующим этапом определения технико-экономических показателей (ТЭП) проекта является оценка трудозатрат, длительности и средней численности разработчиков при реализации проекта. Она основана на согласовании между разработчиком и заказчиком производительности труда программиста -  $P$ .

Используя нормативы трудоемкости (табл. 2.2.) разработки программ в базовой модели СОСОМО, примем  $P = 220$  строк/чел.-месяц (простая информационно-справочная система, количество строк – до 30 тыс.). Трудозатраты на разработку определяются по формуле 2.3:

$$T = 4372 / 220 = 19,87 \text{ чел.- месяцев}$$

И, наконец, при заданной длительности разработки (12 месяцев), получаем искомую среднюю численность персонала, необходимого для ее разработки по формуле 2.4:

$$Z = 19,87 / 12 = 1,66 \text{ чел.}$$

Итак, с помощью прямого метода мы определили основные технико-экономические показатели разработки:

1) трудозатраты на разработку системы составят 19,87 человеко-месяцев;

2) необходимые людские ресурсы при реализации системы = 1,66 чел.

## **II. Определение технико-экономических показателей проекта методом функциональных точек**

### **Исходные данные.**

Тип системы: программно-информационная

Сложность системы: простая

Язык программирования – Delphi

Плановый срок на разработку системы, установленный заказчиком – 12 месяцев.

Размеры программной системы оцениваются в терминах количества и сложности бизнес-процессов (функций), реализуемых в данном программном коде. Система с использованием методологии структурного анализа и проектирования описывается в виде многоуровневой графической модели, представленной в виде совокупности пользовательских бизнес-процессов, каждый из которых включает в себя входные и выходные данные, преобразования, внешние интерфейсы.

Процедура оценивания размеров программной системы соотносится с одним из пользовательских бизнес-процессов и состоит из следующей последовательности этапов:

- выделение множества бизнес-процессов;
- подсчет количества функциональных точек бизнес-процесса в разрезе каждой категории;
- определение весовых коэффициентов сложности каждой функции;
- учет факторов и требований среды разработки программной системы;
- вычисление интегральных показателей сложности;
- вычисление итогового количества функциональных точек;
- определение размеров программной системы в показателях LOC;
- определение размеров программной системы в целом.

На основании методики, изложенной в разделе 2.3.2. рассчитаем количество функциональных точек по каждому бизнес-процессу и заполним соответствующие таблицы по аналогии с таблицей 2.3.

При этом используем весовые коэффициенты сложности выводов, вводов, опросов ввода, опросов вывода, сложности структурных данных (файлов), сложности интерфейсов (табл. 2.4. – 2.9).

Для разрабатываемой АИС получаем рабочие таблицы определения количества функциональных точек по каждому бизнес-процессу (табл. 2-6).

Таблица 2.

Рабочая таблица определения количества  
функциональных точек по бизнес-процессу  
«Администрирование системы»

<b>АИС «Администратор»</b>				
Категория функций	Простые	Средние	Сложные	Кол-во точек
Количество выводов	0	5*8	0	40
Количество вводов	0	5*8	0	40
Количество опросов вывода	0	0	7*8	56
Количество опросов ввода	0	0	6*8	48
Количество файлов	0	10*3	0	30
Количество интерфейсов	0	0	5*18	90
<b>Количество функциональных точек</b>				<b>304</b>

Поясним, что в данном случае, для бизнес-процесса «Администрирование системы»:

1) категория функций определена как «Средние» и «Сложные»;

2) количество «средних» выводов = 40 (5\*8), это означает, что количество элементов данных составляет = 8 (в интервале от 6 до 19), количество файлов = 3 (интервал 2-3), при этом коэффициент сложности  $\alpha_{12} = 5$  (табл. 2.4.);

3) количество вводов = 40 (5\*8), соответственно количество элементов данных составляет = **8** (в интервале от 6 до 19), количество файлов = 2 (интервал 2-3), при этом коэффициент сложности  $\alpha_{22} = 5$  (табл. 2.5);

4) количество «сложных» опросов вывода = 56 (7\*8), то есть количество элементов данных = **8** (интервал от 6-19), соответственно  $\alpha_{32} = 7$  при количестве файлов = **8** (табл. 2.6.);

5) количество «сложных» опросов ввода = 48 (6\*8), то есть количество элементов данных = **8**, при количестве файлов = **8**, при этом  $\alpha_{42} = 6$  (табл. 2.7)

6) количество файлов = 30 (10\*3), соответственно количество элементов данных составляет = **3** (в интервале от 1 до 19), количество логических взаимосвязей – более 6 записей, при этом коэффициент сложности  $\alpha_{51} = 10$  (табл. 2.8).

7) количество интерфейсов = 90 (5\*18), соответственно - количество элементов данных составляет 18 (интервал от 1 до 19),  $\alpha_{61} = 5$  (табл. 2.9) при одной логической записи типа формат/взаимосвязь.

Если категории функций – несущественны, то параметры матриц по ним равны нулю.

Общее количество функциональных точек определяется по формуле:

$$F = \sum_{i=1}^6 \sum_{j=1}^3 \alpha_{ij} \cdot x_{ij} = 304 \text{ точки}$$

Аналогичным образом заполняются рабочие таблицы по остальным бизнес-процессам разрабатываемой автоматизированной системы.

Таблица 3.

Рабочая таблица определения количества функциональных точек по бизнес-процессу «Прием продуктов»

<b>АИС «Прием продуктов»</b>				
Категория функций	Простые	Средние	Сложные	Кол-во точек
Количество выводов	0	5*4	7*4	48
Количество вводов	4*1	5*1	0	9
Количество опросов вывода	4*1	0	0	4
Количество опросов ввода	0	0	10*6	60
Количество файлов	7*1	10*4	0	47
Количество интерфейсов	0	0	10*3	30
<b>Количество функциональных точек</b>				<b>198</b>

Таблица 4.

Рабочая таблица определения количества функциональных точек по бизнес-процессу «Прием заказов»

<b>АИС «Прием заказов»</b>				
Категория функций	Простые	Средние	Сложные	Кол-во точек
Количество выводов	0	5*2	7*2	24
Количество вводов	4*1	5*1	7*1	16
Количество опросов вывода	0	0	7*1	7
Количество опросов ввода	0	0	6*10	60
Количество файлов	0	15*4	0	60
Количество интерфейсов	0	0	10*3	30
<b>Количество функциональных точек</b>				<b>197</b>

Таблица 5.

Рабочая таблица определения количества функциональных точек по бизнес-процессу «Прием платежей»

<b>АИС «Прием платежей»</b>				
Категория функций	Простые	Средние	Сложные	Кол-во точек
Количество выводов	0	5*4	0	20
Количество вводов	4*1	5*1	0	9
Количество опросов вывода	0	0	7*1	7
Количество опросов ввода	0	0	15*6	90
Количество файлов	0	10*5	0	50
Количество интерфейсов	0	0	10*12	120
<b>Количество функциональных точек</b>				<b>296</b>

Таблица 6.

Рабочая таблица определения количества функциональных точек по бизнес-процессу «Исполнение заказов»

<b>АИС «Исполнение заказов»</b>				
Категория функций	Простые	Средние	Сложные	Кол-во точек
Количество выводов	0	4*5	0	20
Количество вводов	0	0	8*6	48
Количество опросов вывода	0	0	7*1	7
Количество опросов ввода	0	0	7*2	14
Количество файлов	0	10*8	0	80
Количество интерфейсов	0	0	10*10	100
<b>Количество функциональных точек</b>				<b>269</b>

Общее количество функциональных точек по всем бизнес-процессам составит:

$$F = 304 + 198 + 197 + 296 + 269 = 1264 \text{ точки}$$

Следующим этапом определения размерности программной системы является учет факторов и требований среды разработки (конечных пользователей системы), так как от этих факторов зависит сложность предметной области и качество создаваемого программного обеспечения.

Влияние этих факторов на размеры программного обеспечения оценивается по ряду показателей, согласно таблице 2.10. При этом каждый из показателей, в свою очередь, оценивается по пятибалльной шкале измерения, которая приведена в табл. 2.11. (оценка существенности влияния факторов среды).

Учитывая вышеизложенное, проводим оценку влияния данных факторов. Результаты заносим в таблицу 7.

Таблица 7.

## Факторы и требования среды разработки

№ пп	Факторы среды	Значение
1	Каналы передачи данных	4
2	Распределенные вычисления	1
3	Производительность системы	5
4	Конфигурирование	2
5	Частота транзакций	2
6	Интерактивная разработка	2
7	Пользовательский интерфейс	5
8	Интерактивное обновление БД	3
9	Сложность обработки запросов	4
10	Сложность установки ПО	5
11	Сложность эксплуатации системы	5
12	Степень распределенности системы	3
13	Гибкость изменения функций	4
<b>Суммарное значение коэффициентов (N)</b>		<b>45</b>

Рассчитаем влияние факторов внешней среды по формуле 2.6:

$$Z = 0.65 + (0.01 \cdot N) = 1.1$$

где  $N$  – суммарное значение весовых коэффициентов факторов внешней среды = 45.

Уточненное количество функциональных точек с учетом факторов внешней среды определим по формуле 2.7:

$$R(F) = F \cdot Z = 1264 \cdot 1.1 = 1390 \text{ точек}$$

Размеры программной системы определяем в виде количества строк исходного кода в терминах Lines of code-LOC.

В качестве базового показателя количества строк исходного кода используется число операторов языка ассемблер.

Варианты преобразования размеров программы, оцененной по этому измерителю в размеры программы кода, написанного на других языках программирования и наоборот, используем из таблицы 2.12. Размерность программного обеспечения для конкретного языка программирования определим по формуле 2.8.:

$$R(LOC) = R(F) \cdot LOC,$$

где  $LOC$  – среднее количество операторов конкретного языка программирования, требующегося для реализации одной функциональной точки.

В нашем случае используется язык Delphi. Преобразовав размеры программной системы, написанной на языке Delphi, получаем соответствие 11 строк кода ассемблер и 1 строки кода Delphi, при этом показатель LOC на 1 функциональную точку равен 29.

$$R(LOC) = R(F) \cdot LOC = 1390 \cdot 29 = 40310 \text{ строк кода}$$

Оценка трудозатрат проводится с помощью степенной функции (формула 2.9):

$$T = A \cdot R^E(KLOC)/12,$$

где  $T$  – трудозатраты, выраженные в человеко-месяцах;

$R(KLOC)$  размерность программной системы, выраженная в тысячах строк кода.



Значения параметров  $A$  и  $E$  получим из таблицы коэффициентов математической модели оценки трудозатрат на основе базовой модели СОСОМО в зависимости от типа программной системы (табл. 2.13)  $A = 3$ ,  $E = 1.12$ .

$$T = 3 \cdot 40.310^{1.12} / 12 = 15.7 \text{ чел.-месяцев}$$

Средняя численность сотрудников, занятых в проекте, составляет:

$$Z = T / Д = 15.7 / 12 = 1,31 \text{ чел.}$$

Таким образом, метод функциональных точек определил следующие основные технико-экономические показатели:

- 1) трудозатраты на разработку системы составят 15.7 человеко-месяцев;
- 2) необходимые людские ресурсы при реализации системы = 1,31 чел.

### III. Метод определения ТЭП проекта на основе размерности базы данных программной системы

В результате анализа объекта автоматизации с помощью ER-моделирования строим концептуальную модель базы данных программной системы для определения количества таблиц (объектов) предметной области, связей и атрибутов (рис. 3).

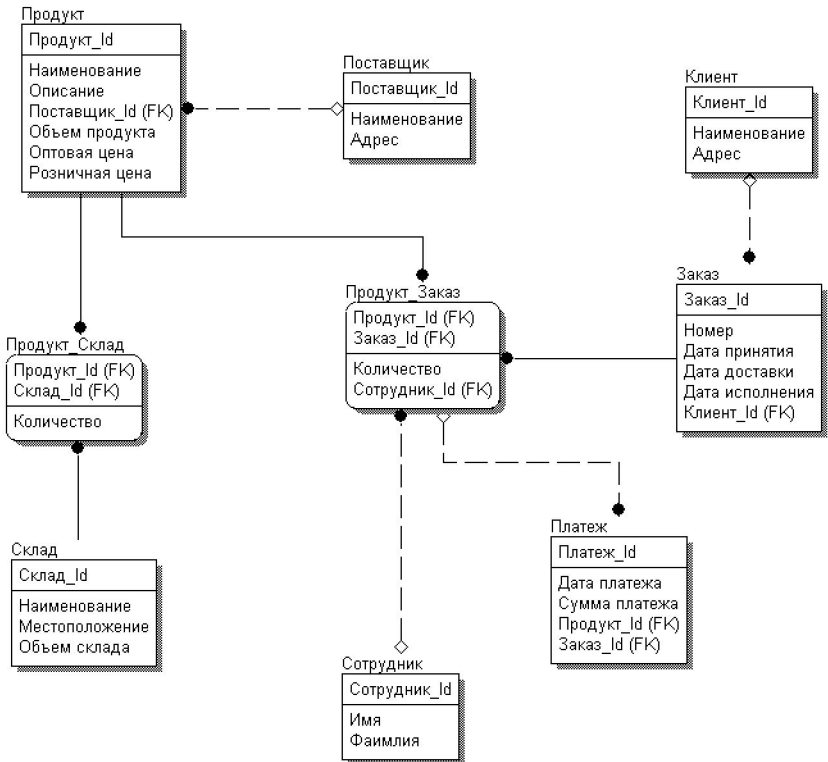


Рис. 3. Концептуальная модель базы данных

Анализируя построенную модель БД получаем:

$N$  – количество таблиц (объектов) = 9;

$K_1$  – количество взаимосвязей между объектами = 8;

$M$  – количество атрибутов на один объект = 27/9 = 3.

Размерность программного обеспечения (в данном случае – базы данных) определяется по формуле 2.14:

$$R = 2N \cdot 5K_1 \cdot 10M,$$

Подставляя в формулу результаты анализа, получаем размерность базы данных:

$$R = 2 \cdot 9 \cdot 5 \cdot 8 \cdot 10 \cdot 3 = 21600 \text{ полей БД}$$

Далее переходим к расчету ТЭП проекта и определим понятие «нормализованной величины» при создании программной системы. Этой величиной является количество формируемых атрибутов, входящих в электронные таблицы посредством установленных связей. При значениях  $N$ ,  $K_1$  и  $M$ , равных единице, величина, выражающая их количество равна 100.

Трудозатраты определяются по формуле 2.15 на основе статистических нормативов трудоемкости, приведенных в табл. 2.15:

$$T = 0.01 \cdot R \cdot \theta,$$

где  $\theta$  - норматив трудоемкости разработки программной системы, который характеризует собой категорию сложности разрабатываемой системы, в т.ч. размерность базы данных.

В нашем случае размерность базы данных (21600) находится в нормативном промежутке до 90 тыс. полей, что соответствует значению норматива  $\theta = 0,00566$ . Таким образом, трудоемкость будет равна:

$$T = 0.01 \cdot 21600 \cdot 0.00566 = 1.22 \text{ чел.-месяцев}$$

Длительность разработки, установленная заказчиком  $D = 12$  месяцев, тогда средняя численность специалистов, которые должны быть привлечены к реализации программной системы составит:

$$Z = T / D = 1.22 / 12 = 0.1 \text{ чел.}$$

Таким образом, применяя метод определения ТЭП на основе размерности базы данных ПС, мы определили следующие основные технико-экономические показатели разработки:

- 1) трудозатраты на разработку системы составят 1,22 человеко-месяцев;
- 2) необходимые людские ресурсы при реализации системы = 0,1 чел.

## ВЫВОДЫ

При расчете технико-экономических показателей по трем методам, при заданной заказчиком длительности разработки 12 месяцев, трудозатраты и численность исполнителей приведены в табл. 8.

Таблица 8

Выводы. Оценка методов определения трудозатрат

Метод	Трудозатраты, чел.-месяц.	Длительность, месяцев	Исполнителей, чел.
Прямой метод (экспертных оценок)	19,87	12	1,66
На основе размерности базы данных программной системы	1,22	12	0,10
Метод функциональных точек	15,7	12	1,31

Очевидно, что трудозатраты и численность разработчиков, полученные с помощью метода определения ТЭП на основе размерности баз данных слишком занижены, т.к. расчеты велись по концептуальной (упрощенной) модели БД.

После расчета технико-экономических показателей проекта выбираем исходные данные (трудозатраты/длительность) для определения стоимости (договорной цены) на создание программной системы.

#### IV. Определение стоимости (договорной цены) на создание программной системы

##### Определение фонда оплаты труда на разработку и комплексные испытания программной системы

В основу определения фонда оплаты труда положены:

- длительность реализации каждого этапа жизненного цикла проекта;
- количество и качественный состав специалистов, привлекаемых на каждом этапе проекта;
- базовая месячная ставка специалиста-программиста.

Выбираем исходные данные, полученные с помощью метода функциональных точек:

- трудоемкость (Т) = 15,7 чел.-месяцев;
- длительность (Д) = 12 месяцев.

Заполняем таблицу средней численности сотрудников, занятых на каждом из этапов создания ПС, используя статистические данные таблицы 2.16 и получаем расчетную таблицу 9.

По формуле 2.17 выполняем расчет средней численности сотрудников, занятых на каждом из этапов создания ПС.

$$Z_i = \alpha_i T / \beta_i D \quad \text{где } i=1,4.$$

Таблица 9

Средняя численность сотрудников, занятых на каждом из этапов создания программной системы и длительности каждого этапа

Этапы жизненного цикла	Численность сотрудников, чел.	Длительность, месяцев
Анализ предметной области и разработка требований	1.31	1.2
Проектирование	0.96	3.6
Программирование	1.52	4.2
Тестирование и комплексные испытания	1.44	3.0

Следующий шаг – распределение специалистов по этапам жизненного цикла (ЖЦ) создания ПС, при этом численность каждого типа специалистов на каждом из этапов ЖЦ определяется по выражению 2.18 с использованием статистического распределения таблицы 2.17:

$$Z_{ij} = P_{ij} \cdot Z_i, i=1,4, j=1,3,$$

где  $P_{ij}$  – относительная доля (%) специалистов  $J$ -го типа, привлекаемых для реализации проекта на  $i$ -ом этапе.

Данные представлены в таблице 10.

Таблица 10

Численность каждого типа специалистов на каждом из этапов жизненного цикла создания программной системы

Этапы жизненного цикла	Типы специалистов, чел. ( $Z_{ij}$ )		
	Аналитики	Программисты	Технические специалисты
Анализ предметной области и разработка требований	0.52	0.26	0.52
Проектирование	0.34	0.34	0.29
Программирование	0.15	0.98	0.38
Тестирование и комплексные испытания	0.22	0.86	0.36

Примем размер ставки программиста = 15 тысяч рублей, как рыночную базовую ставку программиста в данном регионе.

Месячная ставка специалиста-программиста соотносится к ставке системного аналитика как 1:1,3, а к ставке технического специалиста – как 1:0,7, следовательно:

- базовая ставка программиста = 15 000 руб.
- ставка аналитика = 19 500 руб.
- ставка техника = 10 500 руб.

Фонд заработной платы для реализации  $i$ -го этапа проекта определим по формуле 2.19:

$$S_i = \sum_{j=1}^3 Z_{ij} \cdot D_i \cdot S_j, \quad i=1,4,$$

где  $D_i$  — длительность  $i$ -го этапа проекта;

$S_j$  — месячный фонд зарплаты  $j$ -го типа специалиста.

Общий фонд заработной платы определим по формуле 2.20:

$$S = \sum_{i=1}^4 S_i$$

Далее рассчитаем фонд зарплаты для каждого этапа и общий фонд зарплаты (табл. 11).

Таблица 11

Распределение фонда заработной платы по этапам  
жизненного цикла программной системы

Этапы жизненного цикла	Аналитик	Программист	Техник	ФЗП по этапу
Анализ предметной области и разработка требований	12251	4712	6597	23561
Проектирование	23584	18141	10885	52611
Программирование	12404	62023	16698	91126
Тестирование и комплексные испытания	12634	38874	11338	62847
Итого общий фонд заработной платы				230 145

Таким образом, фонд оплаты труда на разработку и комплексные испытания системы составляет 230 145 рублей.

### Определение фонда оплаты труда на проведение опытной эксплуатации программной системы

Численность сотрудников, привлекаемых к опытной эксплуатации определяется по формуле 2.21:

$$Z_{on} = t_{on} \cdot N,$$

где  $t_{on}$  — срок опытной эксплуатации.

Установим срок опытной эксплуатации, согласованный с Заказчиком – 6 месяцев (при заданной длительности 12 месяцев).

Норматив трудоемкости при проведении опытной эксплуатации  $N$  определяется из таблицы 2.18 (категория сложности) – примем его равным 0,0095 чел.-месяцев (когда количество сеансов работы с системой в течение года (от 650 до 6000)).

Численность сотрудников, привлекаемых для опытной эксплуатации составит:

$$Z_{on} = 6 \cdot 0.0095 = 0.057 \text{ (чел.)}$$

Фонд зарплаты сотрудников, привлекаемых для опытной эксплуатации, определяется по формуле 2.22:

$$S_{on} = Z_{on} \cdot t_{on} \cdot S_n \cdot 0.85,$$

где  $S_n$  — месячная базовая ставка программиста.

В нашем случае вышеуказанный фонд составит:

$$S_{on} = 0.057 \cdot 6 \cdot 15000 \cdot 0.85 = 4360 \text{ руб.}$$

Общий фонд заработной платы на разработку и внедрение программной системы = 230145 + 4360 = 234505 руб.



## **Структура договорной цены на программное обеспечение**

Договорная цена на разработку и внедрение программной системы имеет, в основном, типовую структуру, которая включает в себя соответствующие статьи расходов, приведенные и подробно изложенные в разделе 2.4.3.

Основополагающим элементом, из которого будет произведен расчет стоимости проекта, является рассчитанный выше общий фонд заработной платы (234505 руб.).

Дальнейшие разделы сметы затрат зависят от формы организации разработчика (государственное предприятие, коммерческое) и соответствующих форм налогообложения ее деятельности. В приведенном примере сметы предполагается, что система разработана в коммерческой организации, реализующей продукцию и услуги с обычной системой налогообложения, предусматривающей налог на добавленную стоимость (18%).

Стоимость персонального компьютера, приобретенного для выполнения проекта, составляет 50000 рублей, при этом амортизационные отчисления для средств ВТ стоимостью свыше 40000 рублей, согласно действующему законодательству будут производиться в течение установленного в организации срока полезного использования техники (в нашем случае – 36 месяцев), а за 12 месяцев, определенных Заказчиком на разработку системы, составят  $50000/36*12 = 16667$  рублей.

Кроме того, должны быть предусмотрены расходы на лицензионное программное обеспечение, если его приобретение необходимо для разработки и эксплуатации программного продукта.

Далее определяем необходимые виды основных расходов, из которых и складывается окончательная смета затрат (коммунальные услуги, прочие расходы, накладные расходы и т.д.). Процент накладных расходов не имеет жестких нормативов и зависит от затрат на содержание АУП, бухгалтерии и т.д. в организации.

Итак, с учетом нормативов, приведенных в таблице 2.19. составим смету затрат и определим общую стоимость проекта (таблица 12).

Смета затрат  
на разработку и внедрение системы

Наименование статей расходов	Сумма (руб.)
Фонд оплаты труда	234 505
Страховые взносы ПФР, ФСС и ФОМС (30%)	70 352
Увеличение стоимости основных средств (1 компьютер по цене 50 тыс. рублей)	50 000
Амортизация средств вычислительной техники	16 667
Лицензионное программное обеспечение (Windows 7 Professional – 2 шт.)	18 280
Коммунальные услуги, услуги связи (телефон, Интернет) (3000р. * 12 мес.)	36 000
Прочие расходы (1100р. * 12 мес.)	13 200
<b>Итого прямые расходы</b>	<b>439 003</b>
Фонд развития производства (10% от прямых затрат)	43 900
Накладные расходы (12% от прямых затрат)	52 680
<b>Всего расходов</b>	<b>535 584</b>
Налог на добавленную стоимость (18%)	96 405
<b>Итого договорная цена</b>	<b>631 989</b>

**Окончательный результат.**

Договорная цена на разработку и внедрение программной системы составляет порядка 631,9 тысяч рублей.

## Приложение 2

### Пример анализа рыночной стоимости программного продукта

#### Основные задачи

1. Определить точку безубыточности, т.е. минимальное количество (объем продаж) в месяц по заданной договорной цене, при котором выручка покрывает затраты предприятия (фирмы) на реализацию программного продукта. Построить график точки безубыточности (согласно рис. 3.1).
2. Рассчитать договорную цену тиражируемого программного продукта при заданном объеме рынка продаж.
3. Рассчитать необходимое количество продаж копий программного продукта для полного возмещения затрат и определить срок окупаемости проекта.

#### Исходные данные (постановка задачи)

Небольшая фирма (7-8 сотрудников), специализирующаяся в области разработки и продвижения коробочного программного обеспечения, изучив предварительно рынок прикладных программных средств и определив необходимость создания нового программного продукта, выступила в качестве заказчика АИС «Торговля и склад» и взяла с этой целью банковской кредит в размере 632 тыс. рублей (т.е. договорная цена системы – приложение 1) сроком на 1 год под 20% годовых.

Исследовав рынок ПО подобного типа, экспертами отдела маркетинга установлено, что рекомендуемая стоимость одной копии АИС будет составлять 26280 рублей (стоимость аналогичной АИС – 1С «Склад и Торговля» составляет 28800 руб.).

Основная зарплата специалистов отдела маркетинга (зав. отделом, программист, маркетолог, экономист) составляет 33% от стоимости тиражируемого продукта = 8672 руб.

Накладные расходы на содержание административно-управленческого персонала по данному проекту (директор – оклад – 50000 руб., главный бухгалтер – 30000 руб., секретарь-референт – 20000 руб.), составляют 10% от фонда заработной платы АУП = 10000 руб. в месяц (с налогами).

Прочие постоянные расходы по проекту в месяц – 500 руб.

Амортизируемые основные средства отдела маркетинга – находящийся в эксплуатации ПК стоимостью 45000 руб. при установленном сроке полезного использования – 30 месяцев (2,5 года). Комплектующие и расходные материалы отдела маркетинга на одну продажу – 729 руб. Накладные расходы отдела маркетинга на одну продажу – 747 руб.

**Первая задача** - определение точки безубыточности.

Чистая прибыль фирмы определяется как разница между выручкой и переменными и постоянными издержками (формула 3.1.):

$$P = s \cdot x - (a + b \cdot x) = (s - b) \cdot x - a,$$

где  $P$  - прибыль фирмы;

$x$  – объем выпуска продукции;

$S$  – договорная цена продажи единицы продукции;

$a$  – величина фиксированных расходов;

$b$  – величина переменных издержек на единицу реализованной продукции.

Фиксированные издержки за 1 месяц приведены в табл. 1.

Таблица 1

Постоянные (фиксированные) расходы в месяц

Наименование расходов	Сумма (руб.)
Накладные расходы по проекту АУП (10%)	10 000
Плановое ежемесячное гашение кредита (632000 руб./12 мес.)	52 667
Выплата среднего банковского процента – 20% годовых (632000 руб*20% /12)	10 533
Амортизация средств ВТ отдела маркетинга (1 ПК по цене 45000 руб.) при сроке полезного использования 30 месяцев (45000/30)	1 500
Прочие расходы	500
<b>ИТОГО</b>	<b>75 200</b>

Переменные издержки отдела маркетинга, занимающегося непосредственно тиражированием программного продукта, рассчитываются *на единицу продукции*, т.е. на одну продажу программного продукта (таблица 2).

Таблица 2

## Переменные издержки (отдел маркетинга)

Наименование расходов	Сумма (руб.)
Основная зарплата специалистов (33 % от стоимости тиражируемого продукта)	8 672
Страховые взносы (30%) от фонда зарплаты	2 602
Комплекующие и расходные материалы (картриджи, тонер, бумага, диски CD-DVD и т.д.)	729
Накладные расходы отдела маркетинга (транспорт, услуги связи, Интернет, телефоны и т.д.)	747
<b>ИТОГО</b>	<b>12 750</b>

Объем выпуска, при котором достигается точка безубыточности (нулевой уровень прибыли) определяется по формуле 3.2:

$$x_0 = \frac{a}{(s - b)}$$

$$x_0 = \frac{75200}{(26280 - 12750)} = 5,56 \sim 6 \text{ копий.}$$

Период, за который был реализован данный объем программной продукции, определяется сроком, за который мы определили величину постоянных издержек – 1 месяц.

Таким образом, в течение месяца фирме необходимо подготовить и продать минимум 6 копий программного обеспечения по цене 26280 руб., чтобы покрыть её постоянные и переменные расходы в рамках проекта АИС «ТиС».

**Вторая задача.** Необходимо рассчитать договорную цену тиражируемой системы при заданном объеме рынка продаж – 25 копий программного продукта.

Чтобы узнать рыночную цену, которую можно установить при заданном объеме продаж (при нулевом уровне прибыли), используем выражение 3.3.

$$s_0 = \frac{a + bx_0}{(x_0)}$$

Принимая условие, что в заданном периоде (1 месяц) постоянные и переменные издержки неизменны, определяем искомую договорную цену:

$$s_0 = \frac{75200 + 12750 \cdot 25}{25} = 15758 \text{ руб.}$$

Таким образом, при гарантированном объеме рынка продаж равном 25, договорная цена может быть снижена относительно установленной цены до 15758 руб. за копию, что позволяет установить скидку покупателям в размере 40%.

**Третья задача.** Необходимо рассчитать общее число продаж копий программного продукта для полного возмещения затрат и определить срок окупаемости проекта.

Количество продаж программного продукта для полного возмещения затрат найдем по формуле 3.6:

$$x_n = \frac{K * a}{(s - b)} = \frac{12 * 75200}{(26280 - 12750)} = 66,7 \sim 67 \text{ копий}$$

Срок окупаемости проекта определим по выражению 3.7.

$$C_{ok} = \frac{x_n}{x_0} = \frac{67}{6} = 11,2 \text{ месяца.}$$

**Выводы.** Срок окупаемости проекта при продаже не менее 6 копий продукта в месяц (точка безубыточности) и рыночной стоимости 26280 руб. за копию составляет 11,2 месяца, то есть для того, чтобы окупить все расходы на реализацию проекта (полное возмещение затрат) необходимо продать 67 копий программного продукта.