

РЕКОМЕНДАЦИИ
к Методике оценки трудоемкости разработки, сопровождения и
эксплуатации программных средств

СОДЕРЖАНИЕ

1. ОБЛАСТЬ ПРИМЕНЕНИЯ	3
2. НОРМАТИВНЫЕ ССЫЛКИ	4
3. ОБЩИЕ ПОЛОЖЕНИЯ.....	5
4. РАСЧЕТ ТРУДОЕМКОСТИ РАЗРАБОТКИ ПРОГРАММНЫХ СРЕДСТВ С ИСПОЛЬЗОВАНИЕМ КАТАЛОГА ФУНКЦИЙ	7
4.1. Оценка объема ПС.....	7
4.2. Оценка трудоемкости разработки программного средства	9
4.3. Правила применения коэффициентов	12
4.4. Особенности расчета трудоемкости разработки макета программного средства.....	15
5. РАСЧЕТ ТРУДОЕМКОСТИ РАЗРАБОТКИ ПРОГРАММНЫХ СРЕДСТВ С ИСПОЛЬЗОВАНИЕМ МЕТОДА ФУНКЦИОНАЛЬНОЙ БАЛЛЬНОЙ ОЦЕНКИ	17
5.1. Оценка объема ПС.....	17
5.2. Оценка трудоемкости разработки программного средства	19
6. РАСЧЕТ ТРУДОЕМКОСТИ РАЗРАБОТКИ ПРОГРАММНЫХ СРЕДСТВ С ИСПОЛЬЗОВАНИЕМ МЕТОДА РАСЧЕТА РАЗМЕРА ПС ПО ЭЛЕМЕНТАМ ГРАФИЧЕСКОГО ИНТЕРФЕЙСА ПОЛЬЗОВАТЕЛЯ	20
6.1. Оценка объема ПС.....	20
6.2. Оценка трудоемкости разработки программного средства	34
7. РАСЧЕТ ТРУДОЕМКОСТИ СОПРОВОЖДЕНИЯ ПРОГРАММНЫХ СРЕДСТВ.....	35
7.1. Общие положения.....	35
7.2. Правила применения коэффициентов	38
8. РАСЧЕТ ТРУДОЕМКОСТИ ЭКСПЛУАТАЦИИ ПРОГРАММНЫХ СРЕДСТВ	44
8.1. Общие положения.....	44
8.2. Правила применения коэффициентов	45
9. КОММЕНТАРИИ К ОЦЕНКЕ ЗАТРАТ ВРЕМЕНИ И ЧИСЛЕННОСТИ СПЕЦИАЛИСТОВ	47
10. АНАЛИЗ ПОЛУЧАЕМЫХ РЕЗУЛЬТАТОВ	49
10.1. Общие сведения	49
10.2. Порядок расчета	50
ПРИЛОЖЕНИЕ 1 КАТАЛОГ ФУНКЦИЙ ПРОГРАММНЫХ СРЕДСТВ, СОДЕРЖАЩИЙ РЕКОМЕНДУЕМЫЕ ОБЪЕМЫ ФУНКЦИЙ.....	52
ПРИЛОЖЕНИЕ 2 ПРИМЕР ТЕХНИЧЕСКОГО ЗАДАНИЯ НА РАЗРАБОТКУ ПС	69
ПРИЛОЖЕНИЕ 3 РАСЧЕТ ТРУДОЕМКОСТИ РАЗРАБОТКИ МАКЕТА ПС	73
ПРИЛОЖЕНИЕ 4 ПРИМЕР РАСЧЕТА ТРУДОЕМКОСТИ РАЗРАБОТКИ ПС С ИСПОЛЬЗОВАНИЕМ КАТАЛОГА ФУНКЦИЙ	80
ПРИЛОЖЕНИЕ 5 ПРИМЕР РАСЧЕТА ТРУДОЕМКОСТИ РАЗРАБОТКИ ПС С ИСПОЛЬЗОВАНИЕМ МЕТОДА ФУНКЦИОНАЛЬНОЙ БАЛЛЬНОЙ ОЦЕНКИ.....	86
ПРИЛОЖЕНИЕ 6 ПРИМЕР РАСЧЕТА ТРУДОЕМКОСТИ РАЗРАБОТКИ ПС С ИСПОЛЬЗОВАНИЕМ МЕТОДА РАСЧЕТА РАЗМЕРА ПС ПО ЭЛЕМЕНТАМ ГРАФИЧЕСКОГО ИНТЕРФЕЙСА ПОЛЬЗОВАТЕЛЯ	117
ПРИЛОЖЕНИЕ 7 ПРИМЕР РАСЧЕТА ТРУДОЕМКОСТИ СОПРОВОЖДЕНИЯ ПС	137
ПРИЛОЖЕНИЕ 8 ПРИМЕР РАСЧЕТА ТРУДОЕМКОСТИ ЭКСПЛУАТАЦИИ ПС.....	141
ПРИЛОЖЕНИЕ 9 ШКАЛЫ ДЛЯ ОЦЕНКИ ТОЧНОСТИ ВЫЧИСЛЕНИЙ.....	146
ПРИЛОЖЕНИЕ 10 ВЫДЕЛЕНИЕ КОМПОНЕНТОВ АВТОМАТИЗИРОВАННОЙ СИСТЕМЫ	148

1. ОБЛАСТЬ ПРИМЕНЕНИЯ

1.1. Документ содержит общие рекомендации для подразделений Кредитных организаций, выполняющих расчет трудоемкости разработки, сопровождения и эксплуатации программных средств (ПС) в соответствии с Методикой оценки трудоемкости разработки, сопровождения и эксплуатации программных средств (далее Методика).

1.2. Документ может использоваться Кредитными организациями для оценки трудоемкости разработки, сопровождения и эксплуатации ПС, а также сторонними организациями, выполняющими работы по разработке и сопровождению ПС для Кредитных организаций.

2. НОРМАТИВНЫЕ ССЫЛКИ

При разработке настоящего документа использованы:

ГОСТ Р ИСО/МЭК 12207-99	Информационная технология. Процессы жизненного цикла программных средств.
ГОСТ Р ИСО/МЭК 14764-2002	Информационная технология. Сопровождение программных средств.
ГОСТ 19.101-77	Виды программ и программных документов.
ГОСТ 19781-90	Обеспечение систем обработки информации программное. Термины и определения.
ГОСТ 28806-90	Качество программных средств. Термины и определения.
ГОСТ 34.003-90	Информационная технология. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Термины и определения.
ГОСТ 34.601-90	Информационная технология. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Стадии создания.
ГОСТ 20886-85	Организация данных в системах обработки данных. Термины и определения.

3. ОБЩИЕ ПОЛОЖЕНИЯ

3.1. Методика применяется для расчета трудоемкости разработки, сопровождения и эксплуатации **программных средств**.

Программное средство (по ГОСТ 28806-90) – объект, состоящий из программ, процедур, правил, а также, если предусмотрено, сопутствующих им документации и данных, относящихся к функционированию системы обработки информации.

Допускается использование Методики для оценки трудоемкости разработки, сопровождения и эксплуатации отдельных компонентов (программных средств) автоматизированной системы, если такие компоненты можно классифицировать как программные средства. Точность расчета трудоемкости при этом зависит от того, насколько глубоко интегрирована компонента в систему: чем меньше связей – тем выше точность.

Декомпозиция АС должна производиться с учетом функциональных особенностей составных частей АС, возможности их независимой разработки, ввода в действие (см. Приложение 10 Рекомендаций).

3.2. Не допускается оценка трудоемкости разработки, сопровождения и эксплуатации АС как единого программного средства. Такой подход приведет к занижению трудоемкости за счет того, что не будут учитываться элементы, являющиеся общими для нескольких компонентов АС.

3.3. На трудоемкость разработки, сопровождения и эксплуатации ПС оказывает влияние ряд объективных и субъективных факторов. Для учета влияния этих факторов в Методике введены соответствующие поправочные коэффициенты. Если определить значение какого-либо поправочного коэффициента не представляется возможным, то его значение следует принять равным 1.00.

3.4. В новой редакции Методики изменился состав средств разработки с соответствующими поправочными коэффициентами, используемыми при расчете трудоемкости разработки и сопровождения ПС. Для обеспечения совместимости с прежней версией Методики используется таблица 3.1 соответствия средств разработки.

Таблица 3.1. Соответствие средств разработки

Средство разработки в соответствии с прежней версией Методики	Средство разработки в соответствии с Методикой 2010 года
Языки низкого уровня (Assembler)	Языки низкого уровня (Assembler)
Процедурные языки высокого уровня (Pascal, C)	Pascal
C#	C#
C++	C++
Microsoft Visual C++	C++
Borland C++	C++
PowerBuilder	VBA
Языки 4GL (Visual Basic, VBA, Delphi), 1C, Uniface	VBA
Visual FoxPro	Clarion
Системы программирования на основе СУБД типа FoxPro (Clipper, Clarion)	Clarion
Встроенные в ПО макроязыки	Встроенные в ПО макроязыки (1C)
Системы программирования на основе СУБД типа Oracle, SqlServer, Informix, Paradox	PL/SQL
Командные скриптовые языки	Командные скриптовые языки
HTML/CSS (без использования редакторов)	HTML/CSS
HTML/CSS (с использованием редакторов)	HTML/CSS
XML/XSLT	XML/XSLT
Java	Java
JavaScript	JavaScript
Скриптовые языки для WEB (ASP, PHP, Perl)	ASP
SQL	SQL

4. РАСЧЕТ ТРУДОЕМКОСТИ РАЗРАБОТКИ ПРОГРАММНЫХ СРЕДСТВ С ИСПОЛЬЗОВАНИЕМ КАТАЛОГА ФУНКЦИЙ

4.1. Оценка объема ПС

4.1.1. Оценка объема разрабатываемого ПС выполняется с использованием Каталога функций программных средств (далее Каталог), приведенного в таблице 1.1 Приложения 1 Методики.

4.1.2. Состав функций разрабатываемого ПС формируется на основе Технического задания (ТЗ) на разработку ПС. Программному средству устанавливается в соответствие набор функций из Каталога. Данный набор функций должен соответствовать следующим основным требованиям:

- охватывать полностью все возможности ПС, описанные в ТЗ и пояснительной записке;
- не иметь пересечений: разные функции не должны выполнять одни и те же действия для решения одной задачи;
- состоять из функций с наибольшей детализацией.

Правила применения Каталога представлены в пункте 3.2 Методики.

4.1.3. Каталог содержит функции трех типов:

1 тип. Атомарные функции (обозначены в Каталоге символом *). Например, функция 1.5.4 «Формирование запросов к аналитическим БД», объем которой задан исходя из выполнения **одного** запроса. Количество реализаций этой функции будет определяться числом различных выполняемых запросов.

2 тип. Композитные функции и функции, реализующие классы задач (обозначены в Каталоге символом **). Например, функция 1.5.11 «Прикладное администрирование аналитических БД». Функции этого типа составлены из других функций Каталога в рамках того же класса функций или представляют собой реализацию значительной части программного средства. Например, приведенная функция включает в себя контроль целостности и восстановление БД, функции настройки и ведение журнала операций с БД (функции 1.5.5, 1.5.8, 1.5.10 Каталога). Объем таких функций примерно соответствует объему входящих в нее функций с учетом доли участия каждой из них в составе композитной функции (см. таблицу 2 Приложения 1 Рекомендаций).

Функция не может применяться одновременно с соответствующей композитной функцией, в состав которой она входит, для решения одной задачи. Например, Если для работы с базой данных выбрана функция «Прикладное администрирование аналитических

БД», то недопустимо для работы с этой же БД использовать функцию «Контроль целостности и восстановление БД».

В этой связи применение композитных функций и функций, реализующих классы задач, допускается в следующих случаях:

- если для ПС затруднительно далее провести детализацию функций (например, недостаточно детально изложено техническое задание);
- для крупных ПС, требующих представления набора функций на более высоком уровне.

Допускается использование не более одной функции данного типа в рамках расчета. Для повышения точности рекомендуется использовать более детальные функции из Каталога. Детальными по отношению к композитным функциям являются функции первого и третьего типа.

3 тип. Функции, реализующие отдельную задачу. К ним относятся все остальные функции Каталога, не относящиеся к двум первым типам. Для таких функций допускается повторное использование.

4.1.4. Для определения объема разрабатываемого ПС необходимо определить объем каждой отдельной функции, входящей в набор функций данного ПС. Объем функции определяется одним из способов, приведенных в Методике в пункте 3.2.1.

4.1.5. Объемы функций в Каталоге представлены в виде диапазона значений. Нижняя граница объема функции из Каталога соответствует предоставлению минимально возможного набора операций для реализации задачи функции, что вполне достаточно для реализации ПС среднего уровня.

Если значение объема функции на языке разработки (например, в случае расчета по написанному коду ПС) находится ниже или выше рекомендуемого диапазона, то в качестве значения объема функции следует принимать нижнюю или верхнюю границу допустимого диапазона значений соответственно.

Выбор объема функции с верхней границей объема должен обязательно обосновываться разработчиком.

В случае невозможности оценки объема функции следует выбирать величину объема в соответствии с рекомендованным значением в Приложении 1 Рекомендаций.

4.1.6. Если функция входит в состав готового программного модуля или библиотеки (например, разработчик использует готовый исходный код или реализацию функции из заимствованного программного обеспечения), то на трудозатраты она никак не влияет и степень ее повторного использования при этом равна 0 (полностью

используется повторно). Например, степень повторного использования для любой функции из заимствованного программного модуля, включаемого в разрабатываемое программное средство, равна 0.

Если в рамках программного средства или компонента программного средства некоторая функция используется повторно (исходный код или реализация функции дописывается разработчиком, или исходный код готовой функции разработчиком частично модифицируется), то величина коэффициента повторного использования будет равна доле изменений объема функции, используемой повторно. Например, в расчете необходимо использовать чтение данных XML формата. При этом используется готовая библиотека, встроенная в операционную систему. Для полного достижения необходимой функциональности (если, например, после чтения XML необходимо определенным образом обработать данные) необходимо написать еще 500 строк кода. Объем функции «Загрузка файлов», с помощью которой моделируется чтение (загрузка) XML формата, составляет по каталогу 1250 строк кода. Таким образом, степень повторного использования функции составит $k_i = 500 / 1250 = 0.4$.

В случае, когда функция реализуется неоднократно (число реализаций функции задается больше 1), величина коэффициента ее повторного использования (для заданного числа реализаций) определяется как доля изменений объема всех реализаций. Например, для функции «Ведение журналов» с объемом 2750 строк кода задано 4 реализации ($r_i = 4$) по одной на тип журнала. Объем всех реализаций функции без учета прочих факторов составит $2750 * 4 = 11000$ строк кода. При этом 40% кода каждой функции ($2750 * 0.4 = 1100$ строк кода) одинаковы (используется повторно). Объем повторно используемого кода составит $1100 * 4 = 4400$ строк кода. Таким образом, степень повторного использования составит $k_i = (11000 - 4400) / 11000 = 0,6$ (60%).

4.1.7. Объем программного средства в строках исходного кода с учетом его структуры рассчитывается в соответствии с пунктом 3.2.5 Методики.

4.2. Оценка трудоемкости разработки программного средства

4.2.1. Расчет трудоемкости разработки программных средств выполняется Разработчиком ПС (опорные объекты информатизации Кредитных организаций или сторонние организации).

4.2.2. Оценка трудоемкости разработки программного средства выполняется в два этапа:

- расчет базовой трудоемкости разработки ПС;
- расчет трудоемкости отдельных подпроцессов разработки;

Расчет базовой трудоемкости разработки выполняется в соответствии с пунктом 3.2.6 Методики.

Расчет трудоемкости отдельных подпроцессов разработки ПС выполняется строго в соответствии с пунктами 3.2.8–3.2.14 Методики. Поскольку модель расчета трудоемкости разработки предполагает разбиение базового значения трудоемкости по этапам, то итоговая величина трудоемкости разработки считается путем суммирования ее по всем этапам (пункт 3.2.14 Методики).

Расчет трудоемкости разработки с учетом влияния сроков работ выполняется в соответствии с пунктом 3.2.15 Методики.

4.2.3. Если при разработке ПС используются несколько средств разработки, то это программное средство разбивается на функции таким образом, чтобы каждая функция была реализована с использованием одного средства разработки. Различные функции одного ПС могут быть реализованы с использованием различных средств разработки.

4.2.4. Если в процессе разработки программного средства появляется необходимость реализовать дополнительно одну или несколько функций (задач), то следует оценивать величину трудоемкости доработок следующим образом:

- рассчитать величину трудоемкости T_2 с учетом реализации дополнительных функций (задач) для всего программного средства;
- найти величину трудоемкости доработок, которая будет составлять $\Delta T = T_2 - T_1$, где T_1 – величина трудоемкости разрабатываемого ПС без дополнительных функций (задач).

4.2.5. В случае, когда программное средство уже было разработано, и возникает необходимость реализовать дополнительно одну или несколько функций (задач), или в случае, когда изменение требований нормативных документов Кредитных организаций предполагает существенного изменения функциональных возможностей ПС, трудоемкость доработок (развития) рекомендуется рассчитывать одним из следующих способов:

- 1) Рассчитать величину трудоемкости разработки T_1 для исходного ПС так, как если бы оно разрабатывалось заново, или в качестве T_1 использовать ранее рассчитанную величину трудоемкости, если для данного ПС такое вычисление проводилось. Затем выполнить все шаги п. 4.2.4 Рекомендаций.
- 2) Выполнить следующие шаги:

а) оценить трудоемкость необходимых доработок в соответствии с разделом 3.2 Методики, получив величину T_3 . При этом в расчет включаются только те функции (задачи), которые необходимо реализовать дополнительно;

б) так как Методика предполагает оценку трудоемкости самостоятельного и полнофункционального ПС, а включаемые в расчет функции (задачи) образуют дорабатываемый функционал ПС, полученный результат необходимо скорректировать по формуле: $\Delta T = K \cdot T_3$. Значение коэффициента K выбирается по следующей таблице:

Отношение объема доработок к объему ПС, r , %	Значение K
$r < 10$	0,2
$10 \leq r < 30$	0,4
$30 \leq r < 60$	0,5
$60 \leq r < 90$	0,6
$R \geq 90$	0,7

4.2.6. Для примерной оценки величины трудоемкости разработки АС необходимо:

- 1) Представить АС в виде компонентов (см. Приложение 10 Рекомендаций);
- 2) Оценить величину трудоемкости каждого компонента АС;
- 3) Рассчитать трудоемкость T^* для АС суммированием трудоемкости разработки ее каждого компонента;
- 4) Уменьшить величину T^* экспертным образом, вычитая из нее трудоемкости многократно учитываемых факторов, с учетом того, что:
 - подпроцесс «анализ требований» должен выполняться один раз для всей АС;
 - подпроцесс «ввод в действие» выполняется для всей АС, а не для каждого компонента;
 - вспомогательные и организационные процессы, величина трудоемкости для которых уже входит в трудоемкость отдельных подпроцессов, должны быть учтены один раз для всей АС;
 - разработка документации выполняется для всей АС, а не для каждого компонента.

4.3. Правила применения коэффициентов

4.3.1. Настоящий раздел регламентирует правила выбора уровней поправочных коэффициентов для модели расчета трудоемкости разработки ПС согласно порядку расчета, представленному в Методике.

4.3.2. Значение коэффициента сложности функции ($K_{сложн}$) при расчете объема ПС (пункт 3.2.4 Методики) выбирается согласно уровню сложности функции соответствующего типа операций.

Каждая функция из Каталога соответствует одному из пяти типов операций:

- управляющие операции;
- вычислительные операции;
- операции, зависящие от аппаратуры;
- операции управления данными;
- операции управления пользовательского интерфейса.

Для определения уровня сложности функции необходимо выполнить следующие шаги:

- 1 шаг.** Определить, к какому типу операции относится функция. Тип операции определяется по Каталогу функций (таблица 1.1 Приложения 1 Методики).
- 2 шаг.** Определить уровень сложности функции по одной из таблиц 2.1–2.5 Приложения 2 Методики в зависимости от типа операции.
- 3 шаг.** Найденному номеру уровня L_v сопоставить значение из таблицы 2.6 Приложения 2 Методики.

4.3.3. Коэффициент, учитывающий конкретные условия и средства разработки ПС ($K_{ср.разр}$) при расчете объема ПС (пункт 3.2.4 Методики) выбирается в соответствии с используемым для реализации функции средством разработки.

В случае если необходимое средство разработки в приведенной таблице отсутствует, то, согласно Методике, необходимо самостоятельно отнести его к какому либо средству, наиболее близкому по уровню используемого языка. Аналогично следует поступать при пересчете объема кода согласно пунктам 3.2.1 и 3.2.2 Методики для различных средств разработки с применением переводных коэффициентов (таблица 1.2 Приложения 1 Методики).

4.3.4. Значения коэффициентов, учитывающих уровень требований к показателям качества ПС (требования к надежности ($K_{над}$), производительности ПС ($K_{произв}$) и уровню информативности документации ($K_{докум}$) на этапах жизненного цикла) при расчете

величины базовой трудоемкости разработки ПС (пункт 3.2.6 Методики) выбирается с **учетом требований заказчика** к ожидаемому качеству программного средства.

Допустим, если выполняется разработка АРМ, то уровень требования к надежности можно установить средним, поскольку сбои в работе подобного ПС не приведут к серьезным последствиям. Требования к надежности выше среднего устанавливаются для ПС, результатом неправильной работы которых может являться возможная потеря важной информации и т.п.

4.3.5. Уровень коэффициента, учитывающего степень новизны ПС (K_n) при расчете базовой трудоемкости разработки ПС (пункт 3.2.6 Методики), выбирается в соответствии с таблицей 3.1 Приложения 3 Методики.

Программное средство является принципиально новым в случае, если не имеется доступных аналогов данного ПС в ФАП Кредитных организаций.

Программное средство является развитием параметрического ряда ПС в следующих случаях:

- если выполняется разработка новой версии уже существующего программного средства;
- если осуществляется доработка комплекса задач, задачи, функций и создаются новые компоненты, программные модули ПС;

Учет новизны типа ТС и ОС осуществляется следующим образом: если тип ТС/ОС, на котором предполагается функционирование ПС, не использовался в подразделениях Кредитных организаций, то он считается новым.

Тип ТС и ОС устанавливается в ТЗ. Выбор значения коэффициента должен соответствовать требованиям ТЗ.

4.3.6. Значение коэффициента, учитывающего использование технологий разработки (K_{tex}) при расчете величины базовой трудоемкости разработки ПС (пункты 3.2.6–3.2.7 Методики) устанавливается в соответствии со структурой ПС и используемыми технологиями взаимодействия компонентов ПС.

Структура ПС и набор технологий взаимодействия компонентов устанавливаются в соответствии с техническим заданием на разработку программного средства. Технологии взаимодействия компонентов могут выбираться для любой структуры программного средства, кроме структуры «Монолитное ПС». Одна или несколько технологий взаимодействия компонентов выбираются, если они реализуются в программном средстве.

Уровни значений коэффициента, учитывающего используемые средства проектирования ($K_{cp.np}$) при расчете трудоемкости подпроцесса «Проектирование» (пункт

3.2.10 Методики), соответствуют классам применяемых средств проектирования. Разработчик должен отнести используемые им средства проектирования к одной из предлагаемых групп, наиболее близких по своим возможностям для применяемых средств.

Аналогично устанавливается значение коэффициента, учитывающего используемые среды разработки (K_{cp}) при расчете трудоемкости подпроцесса «Программирование» (пункт 3.2.11 Методики).

4.3.7. Коэффициент, учитывающий используемые средства тестирования ($K_{cp.mc}$) при расчете трудоемкости подпроцесса «Тестирование» (пункт 3.2.12 Методики), отражает влияние на значение трудоемкости наличия средств автоматизированного тестирования. Значение коэффициента устанавливается в соответствии с описаниями его уровней.

4.3.8. Поправочный коэффициент, учитывающий размер тестовой БД ($K_{cp.mc}$) при расчете трудоемкости подпроцесса «Тестирование» (пункт 3.2.12 Методики), отражает дополнительные трудозатраты на сбор данных для завершения процесса тестирования. Значение этого коэффициента определяется отношением размера (объема) БД в байтах к количеству строк кода отдельных ПС или комплекса ПС.

Объем данных в БД рассчитывается исходя из размера одной записи и количества записей. Размер одной записи рассчитывается как сумма размеров полей, входящих в запись. Например, тестовая БД состоит из одной таблицы; запись таблицы тестовой БД состоит из двух полей. Первое поле типа INT (размер поля 4 байта), второе – типа VARCHAR (размер поля 256 байтов). В таком случае размер одной записи составит 260 байтов. Допустим, таблица тестовой БД содержит 100 записей, тогда размер тестовой БД составит $(260 \cdot 100) = 26000$.

4.3.9. Значение коэффициента, учитывающего опыт работы программистов со средством разработки ($K_{опыт}$) при расчете объема ПС (пункт 3.2.4 Методики), выбирается с учетом числа разработанных ПС (компонентов АС) коллективом разработчиков с использованием данного средства разработки по аналогичной тематике. Оценка проводится для всего коллектива разработчиков, участвующего в разработке ПС.

Аналогично устанавливается значение коэффициента, учитывающего опыт разработки программных средств подобного типа ($K_{оп}$), при расчете базовой трудоемкости разработки ПС (пункт 3.2.6 Методики).

Значение коэффициента, учитывающего опыт работы аналитиков в данной предметной области ($K_{опыт.ан}$), при расчете трудоемкости подпроцесса «Анализ

требований к ПС» (пункт 3.2.9 Методики) и значение коэффициента, учитывающего опыт проектировщиков в данной предметной области ($K_{\text{опыт.пр}}$), при расчете трудоемкости подпроцесса «Проектирование» (пункт 3.2.10 Методики) выбирается с учетом числа разработанных ПС (компонентов АС), относящихся к данной предметной области.

4.3.10. Оценку уровня квалификации персонала, задействованного в выполнении отдельных подпроцессов, следует производить для всего коллектива, занятого в этом подпроцессе. Значение коэффициента, учитывающего уровень квалификации аналитиков ($K_{\text{квал.ан}}$) при расчете трудоемкости подпроцесса «Анализ требований к ПС» (пункт 3.2.9 Методики) устанавливается согласно квалификации персонала.

Аналогично устанавливается значение коэффициента, учитывающего уровень квалификации проектировщиков в данной предметной области ($K_{\text{квал.пр}}$), при расчете трудоемкости подпроцесса «Проектирование» (пункт 3.2.10 Методики), значение коэффициента, учитывающего уровень квалификации программистов ($K_{\text{квал.прог}}$), при расчете трудоемкости подпроцесса «Программирование» (пункт 3.2.11 Методики), значение коэффициента, учитывающего уровень квалификации персонала, осуществляющего тестирование ($K_{\text{квал.тест}}$), при расчете трудоемкости подпроцесса «Тестирование» (пункт 3.2.12 Методики), значение коэффициента, учитывающего уровень квалификации персонала, осуществляющего ввод в действие ПС ($K_{\text{квал.вн}}$), при расчете трудоемкости подпроцесса «Ввод в действие» (пункт 3.2.13 Методики).

Оставляя право формирования коллектива разработчиков, участвующих в разработке ПС для Кредитных организаций, за руководством компании-разработчика, рекомендуется относить коллектив разработчиков к имеющим очень высокий уровень опыта в случаях, когда компания сотрудничает с Банком России более 4-х лет и имеет более 5-ти разработанных ПС (компонентов АС) для Кредитных организаций или по аналогичной тематике. Кроме того, если компания-разработчик разрабатывает (развивает) отдельные компоненты к ранее разработанному ПС (АС), то уровень квалификации коллектива разработчиков считается как минимум высоким.

4.4. Особенности расчета трудоемкости разработки макета программного средства

4.4.1. Данная методика может применяться для расчета макета программного средства. ***Под макетом понимается совокупность программных элементов, отражающая основные функциональные возможности разрабатываемого программного средства.*** Это означает, что принцип расчета этапов, модели и дискретный уровень коэффициентов остаются теми же самыми, что и для расчета ПС, однако выбор

качественных и количественных характеристик для макета отличается следующим образом:

- требования к производительности – коэффициент устанавливается на уровень «Требования к производительности ПС не установлены (однако производительность ПС должна обеспечивать приемлемое время отклика при работе пользователя в интерактивном режиме)»;

- информативность документации – коэффициент устанавливается на низкий уровень – «Не учтены некоторые потребности жизненного цикла»;

4.4.2. Функциональный состав макета состоит из базовых функций, которые позволяют решить поставленные в ТЗ задачи, т.е. макет не обязан реализовывать все функции как полноценное ПС. Большая часть внимания, как правило, уделяется алгоритмическим функциям, нежели интерфейсным и транспортным. Кроме того, для функций, не являющихся основными для данного ПС (реализация интерфейса и т.п.), рекомендуется выбирать минимальное значение объема.

5. РАСЧЕТ ТРУДОЕМКОСТИ РАЗРАБОТКИ ПРОГРАММНЫХ СРЕДСТВ С ИСПОЛЬЗОВАНИЕМ МЕТОДА ФУНКЦИОНАЛЬНОЙ БАЛЛЬНОЙ ОЦЕНКИ

5.1. Оценка объема ПС

5.1.1. Метод ФБО является альтернативным способом оценки объема ПС. Оценка объема ПС с использованием метода ФБО не связана с субъективной оценкой объема каждой отдельной функции, что имеет место при использовании Каталога функций. В отличие от Каталога функций, данный метод оценивает объем исходя как из логической структуры (набор программных модулей, их структура и состав), так и из функционального состава ПС (набор функций и количество используемых ими элементов данных).

5.1.2. Использование метода ФБО не исключает использование Каталога функций: для оценки объема ПС может выбираться любой из методов в зависимости от имеющихся исходных данных и их полноты. Если данных для использования этого метода недостаточно или из имеющихся документов невозможно достоверно установить одновременно логическую структуру и функциональный состав ПС, то используется Каталог функций.

При использовании метода ФБО необходимо, чтобы техническое задание на разработку содержало:

- UML-диаграммы классов, используемых для определения логических файлов;
- UML-диаграммы вариантов использования, определяющих транзакционные функции;
- ER-диаграммы, задающие структуру используемых баз данных.

5.1.3. Определение границы программного средства и границы расчета выполняется в соответствии с правилами пунктов 4.2.2–4.2.3 Методики. Данные процессы строятся на понятии «Логика работы программы». Необходимо считать, что элемент модели (класс, вариант использования) определяет логику работы программы, если его исключение или модификация приводит к нарушению либо изменению алгоритма решения основной задачи (задачи, повлекшей разработку этого программного средства).

Если исключение или модификация элемента модели не влияет на алгоритм решения основной задачи, а влияет на такие свойства как вариант реализации (в виде сервиса, исполняемого модуля, библиотеки и др.), общую структуру, возможности пользовательского интерфейса и т.п., то этот элемент не задает логику работы программного средства, а задает особенности реализации. Следовательно, в границу

расчета такой элемент не входит. Он включается только в границу программного средства.

5.1.4. Необходимо учитывать, что граница программного средства разделяет не физически различные объекты в рамках одного программного средства (например, клиентская программа и используемая ей база данных – 2 объекта), а наборы объектов, относящихся к различным программным средствам. При этом совокупность объектов одного ПС (например, приложения с набором библиотек) полностью входит в границу этого ПС.


5.1.5. В UML-диаграмме классов под элементами данных следует понимать атрибуты классов с учетом требований правил пункта 4.2.6 Методики. Число записей логических файлов определяется в соответствии пунктом 4.2.6 Методики. Для классов, не связанных отношением композиции или обобщения, число записей каждого логического файла всегда равно 1.

5.1.6. В ER диаграмме, описывающей структуру базы данных, под элементами данных следует понимать атрибуты сущности, представляющей таблицу БД во второй или третьей нормальной форме (в реализации базы данных атрибуты – это поля таблицы) с учетом требований правил пункта 4.2.6 Методики. Число записей для любого такого логического файла всегда равно 1.

5.1.7. Элемент данных может быть представлен не только в виде одиночного атрибута, но и совокупностью атрибутов. Например, если дата представлена как набор из трех атрибутов (число, месяц, год), то все три атрибута считаются как один элемент данных. Аналогично и для других подобных групп (время, адрес, ФИО и др.)

5.1.8. Если на диаграмме представлен набор атрибутов для повторного ввода значения одной величины (например, 12 атрибутов для ввода значения ежемесячного дохода по одному на каждый месяц), то этот набор учитывается как один элемент данных. Другими словами, важно наличие самой величины, а не число ее экземпляров.

5.1.9. В UML-диаграмме вариантов использования под транзакционной функцией следует понимать элемент структуры, называемый «Вариант использования». При этом вариант использования считается транзакционной функцией, если он реализует одну задачу, а не комплекс задач (в этой ситуации его необходимо разбивать на более мелкие единицы).

5.1.10. Если вариант использования на UML-диаграмме, определенный как транзакционная функция, объединен отношением обобщения () либо расширения

`<<extend>>`
 (->) с другими вариантами использования, то обобщаемый либо расширяемый вариант использования не считается транзакционной функцией.

5.1.11. Если вариант использования на UML-диаграмме, определенный как транзакционная функция, связан отношением включения (`<<include>>` (->)) с другими вариантами использования, отражающими на этой диаграмме более детальные действия, то включаемый вариант использования не считается транзакционной функцией.

5.1.12. При выделении транзакционных функций необходимо обязательно учитывать правила, приведенные в пункте 4.2.8 Методики.

5.1.13. Варианты использования, учитываемые как транзакционные функции, имеют аналогичные (или подобные) представленным в техническом задании на разработку названиям выполняемых программой функций (см. пример расчета). Этим же определяется уровень детализации, на котором необходимо рассматривать диаграмму вариантов использования.

Если вариант использования не определен как транзакционная функция, то далее в расчете он не учитывается.

5.2. Оценка трудоемкости разработки программного средства

5.2.1. Согласно Методике, расчет трудоемкости разработки с использованием метода функциональной балльной оценки выполняется аналогично тому, как это делается при использовании Каталога функций. Для этого предварительно необходимо пересчитать объем ПС в функциональных точках в строки исходного кода. Пересчет выполняется в соответствии с пунктом 4.2.15 Методики.

5.2.2. Правила выбора значений коэффициентов аналогичны тем, которые указаны для расчета разработки ПС с использованием Каталога функций.

6. РАСЧЕТ ТРУДОЕМКОСТИ РАЗРАБОТКИ ПРОГРАММНЫХ СРЕДСТВ С ИСПОЛЬЗОВАНИЕМ МЕТОДА РАСЧЕТА РАЗМЕРА ПС ПО ЭЛЕМЕНТАМ ГРАФИЧЕСКОГО ИНТЕРФЕЙСА ПОЛЬЗОВАТЕЛЯ

6.1. Оценка объема ПС

6.1.1. Процесс определения объема (размера) ПС с помощью метода расчета размера ПС по элементам графического интерфейса пользователя состоит из следующей последовательности шагов:

- 1) Выделение экземпляров программных объектов, которые будут разработаны в создаваемом ПС.
- 2) Определение уровня сложности выделенных программных объектов.
- 3) Получение частных оценок размеров программных объектов разрабатываемого ПС.
- 4) Получение итоговой оценки размера ПС, путем аккумуляирования значений частных оценок.

6.1.2. Порядок выделения программных объектов «Клиентские и диалоговые окна». Основное назначение окна – обеспечение визуального взаимодействия пользователя с программой. На рисунке 6.1 представлен пример клиентского окна приложения «Адресная книга».

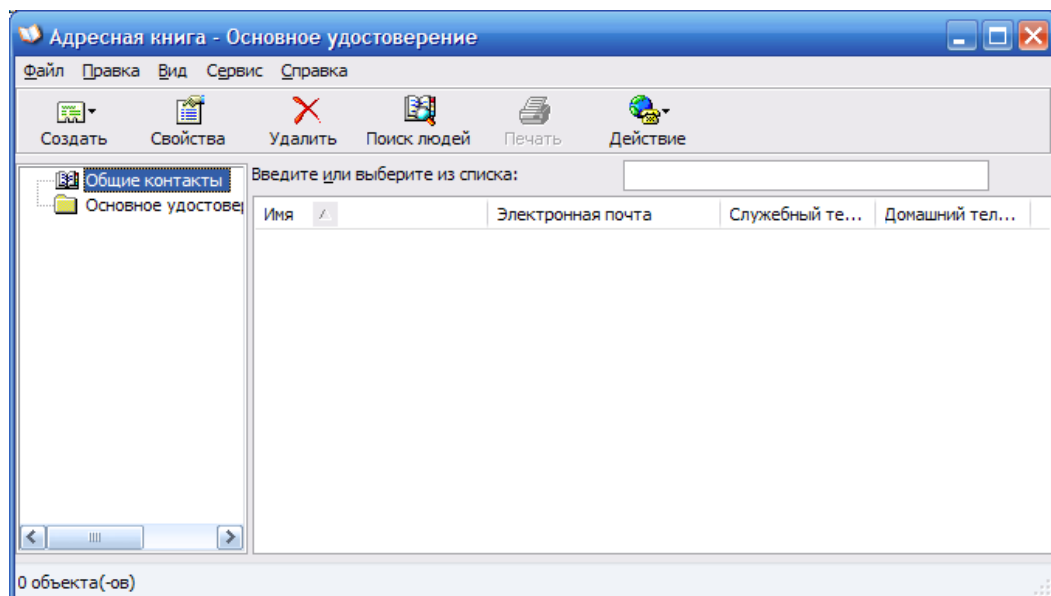


Рис. 6.1 Клиентское окно приложения

Отдельно выделяют подкласс клиентских окон – диалоговые окна. На рисунках 6.2 и 6.3 представлены примеры диалоговых окон.

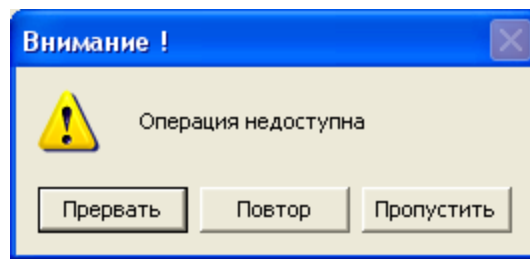


Рис. 6.2 Пример диалогового окна – сообщения

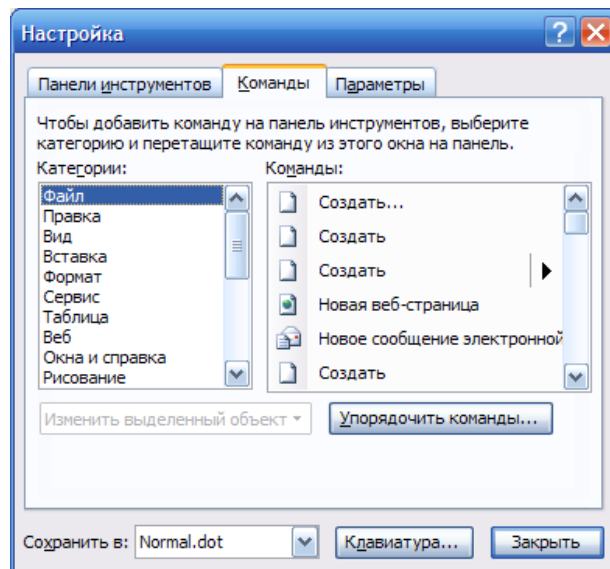


Рис. 6.3 Пример диалогового окна с элементами управления

При выделении диалоговых окон модальные диалоги сообщений и предупреждений не учитываются. Примеры модальных диалогов приведены на рис. 6.4.

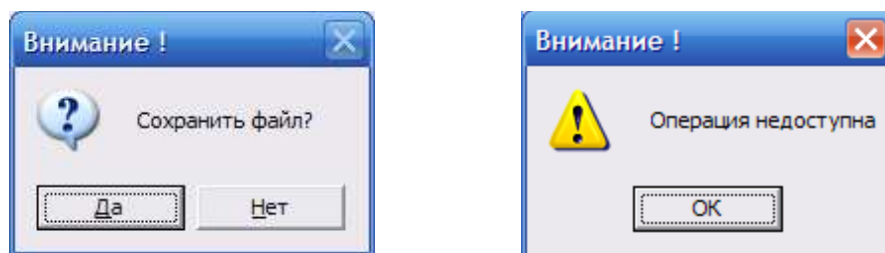


Рис. 6.4 Пример модальных диалоговых окон

Многостраничные диалоги могут учитываться как несколько диалогов – по количеству закладок. Многостраничный диалог с рис. 6.5 должен быть посчитан как 11 диалоговых окон (11 закладок). Свойства такого диалога считаются отдельно для каждой закладки.

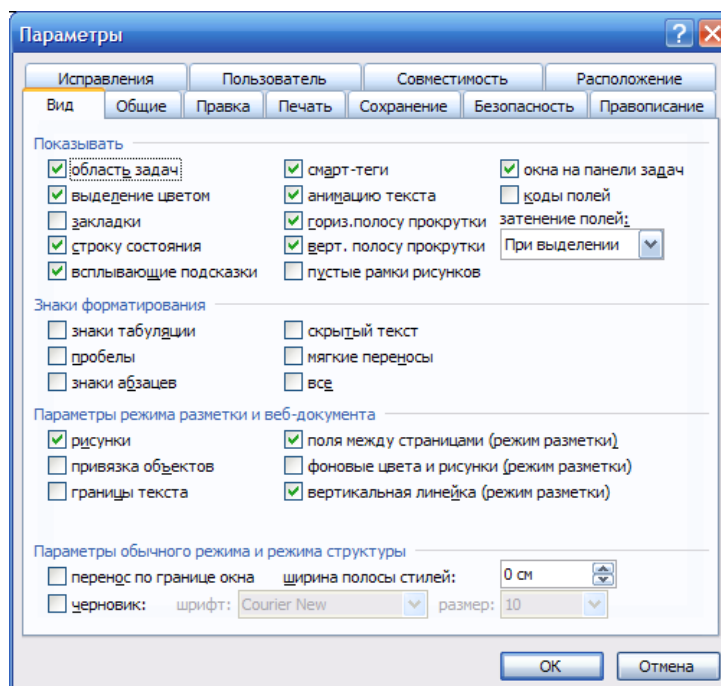


Рис. 6.5 Пример многостраничного диалога

Для каждого окна ПС должен быть создан эскиз (макет) его представления на экране. Макеты окон могут быть созданы с помощью специальных инструментальных средств (дизайнеров форм), так и вручную в виде графических рисунков. На макетах должны быть отражены все элементы интерфейса пользователя, которые будет содержать окно.

На рис. 6.6 и 6.7 изображены примеры макетов клиентского окна ПС.

Рисунок 6.6 Макет окна ПС в виде рисунка

Рисунок 6.7 Макет окна ПС выполненного в дизайнера форм

6.1.3. Для определения функциональной сложности клиентских и диалоговых окон ПС необходимо подсчитать количество элементов интерфейса для ввода/редактирования/просмотра данных, расположенных в окне, учитывая перечисленные ниже правила.

Каждое поле однострочного текстового редактирования (EDIT FIELD) учитывается как один элемент интерфейса для ввода/редактирования/просмотра данных. Однострочное редактируемое текстовое поле – простейший редактор, позволяющий пользователю просматривать и редактировать текст (рис. 6.8).

Рис. 6.8 Пример однострочного редактируемого текстового поля

Каждое поле многострочного текстового редактирования (МЕМО) учитывается как один элемент интерфейса для ввода/редактирования/просмотра данных. Многострочное редактируемое текстовое поле используется для ввода и/или отображения многострочного текста (рис. 6.9).

Рис. 6.9 Пример многострочного редактируемого текстового поля

Каждый список (LIST BOX) учитывается как один элемент интерфейса для ввода/редактирования/просмотра данных. Список – отображает список элементов, позволяет пользователю выбрать один или несколько его элементов. Прокликиваемые списки могут позволять пользователям совершать как единственный, так и множественный выбор (рис. 6.10 и рис. 6.11).

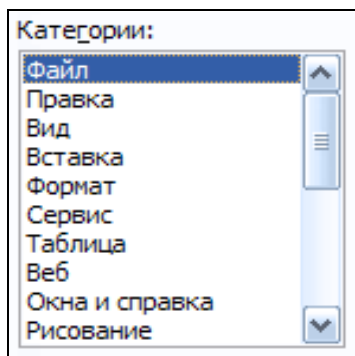


Рис. 6.10 Список единственного выбора

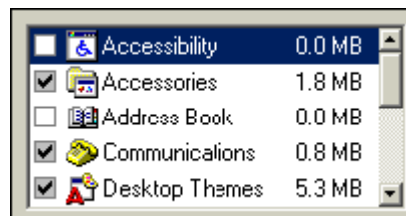


Рис. 6.11 Список множественного выбора с флажками

Каждый комбинированный список (COMBO BOX) учитывается как один элемент интерфейса для ввода/редактирования/просмотра данных. Комбинированный (выпадающий) список – комбинация списка и однострочного элемента редактирования (рис. 6.12). Позволяет пользователю выбрать существующий элемент, либо ввести свой.

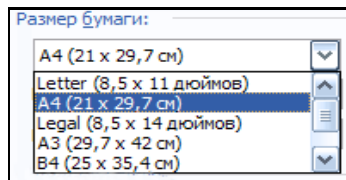
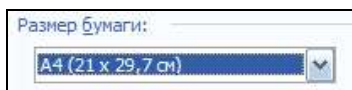


Рис. 6.12 Раскрывающийся комбинированный список

Группа переключателей (RADIO BUTTON) – позволяет пользователю выбрать только одно из возможных значений (рис. 6.13). Вся группа учитывается как один элемент интерфейса для ввода/редактирования/просмотра данных.

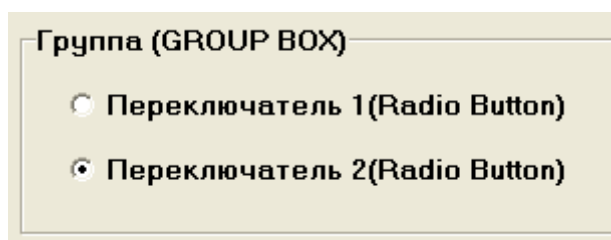


Рис. 6.13 Группа переключателей

Каждый флажок (CHECK BOX) учитывается как один элемент интерфейса для ввода/редактирования/просмотра данных. Флажки – используются, если элемент данных для ввода/вывода может принимать только два возможных значения (рис. 6.14). Например, область окна с рис. 6.14 содержит три элемента интерфейса.

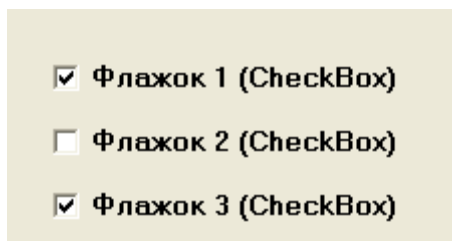


Рис. 6.14 Флажки

Таблицы (GRID VIEW) – предназначены для отображения информации в табличном виде в окне приложения (рис. 6.15).

	Номер	Дата регистр.	Тип	Корреспондент	Содержание	Статус
	1227	28.04.2005	Письмо	ФГУП Почта Росс	О перечислении средств	Исполнен
	1228	28.04.2005	Указание	ФГУП Почта Росс	О предоставлении свед	Исполнен
	1229	28.04.2005	Информационное пись	ФГУП Почта Росс	О выпуске в обращени	Архивный
▶	1230	28.04.2005	Письмо	Краснохолмский г	О получении образова	На исполнении
	1231	28.04.2005	Информационное пись	Центртелеком	Приказ Об изменении	Исполнен
	1232	28.04.2005	Информационное пись	Администрация Ц	Положение о районном	Архивный

Рис. 6.15 Пример таблицы

Каждой таблице соответствует такое количество элементов интерфейса для ввода/редактирования/просмотра данных, сколько столбцов для отображения/редактирования данных в ней присутствует. Например, таблице представленной на рис. 6.15 соответствует шесть элементов интерфейса (шесть столбцов).

Древовидная структура текстовых элементов (TreeView) – используется для отображения иерархий (рис. 6.16). Каждому объекту TreeView соответствует такое количество элементов интерфейса для ввода/редактирования/просмотра данных, сколько уровней иерархии он отображает. Например, объекту TreeView, изображенному на рис. 6.15, соответствует три элемента интерфейса (максимум отображается три уровня иерархии).

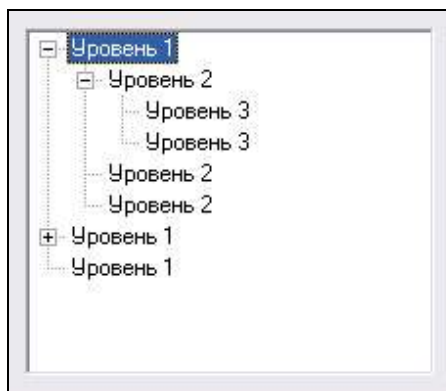


Рис. 6.16 Пример древовидной структуры

Многостраничная панель с закладками (TAB PANEL) предназначена для рационального размещения объектов в окне приложения (рис. 6.17). Многостраничная панель состоит из страниц, каждая из которых может содержать различные компоненты. В каждый момент видна только одна активная страница, переключение происходит с помощью закладок.

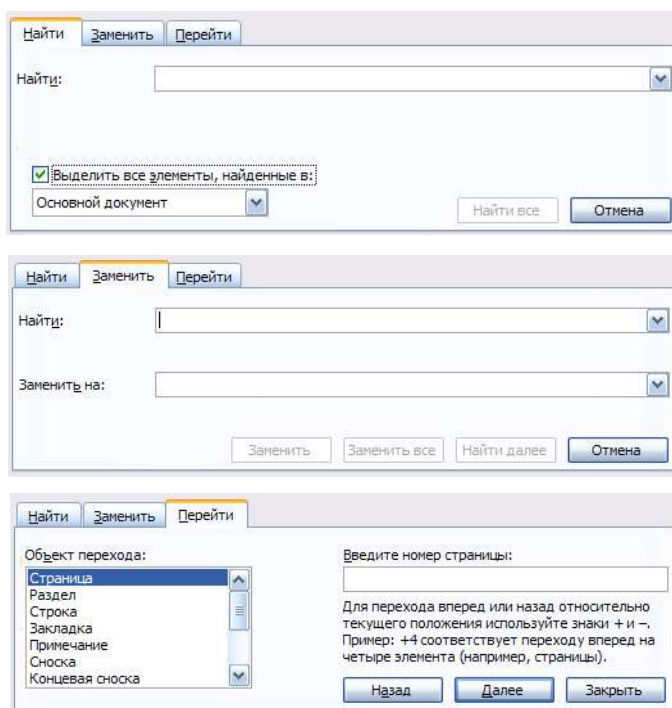


Рис. 6.17 Пример многостраничной панели с закладками

Многостраничная панель с закладками при подсчете количества элементов интерфейса не учитывается. Необходимо учитывать все элементы интерфейса из приведенного в Методике списка, расположенные на всех страницах многостраничной панели, а не только те, которые видны на отображаемой закладке. Например, многостраничная панель, изображенная на рис. 6.17, содержит семь элементов интерфейса для ввода/редактирования/просмотра данных (три элемента на первой странице, два на второй странице и два на третьей).

На рис. 6.18, 6.19, 6.20 приведены примеры подсчета количества элементов интерфейса для ввода/редактирования/просмотра данных для клиентских окон различной конфигурации.

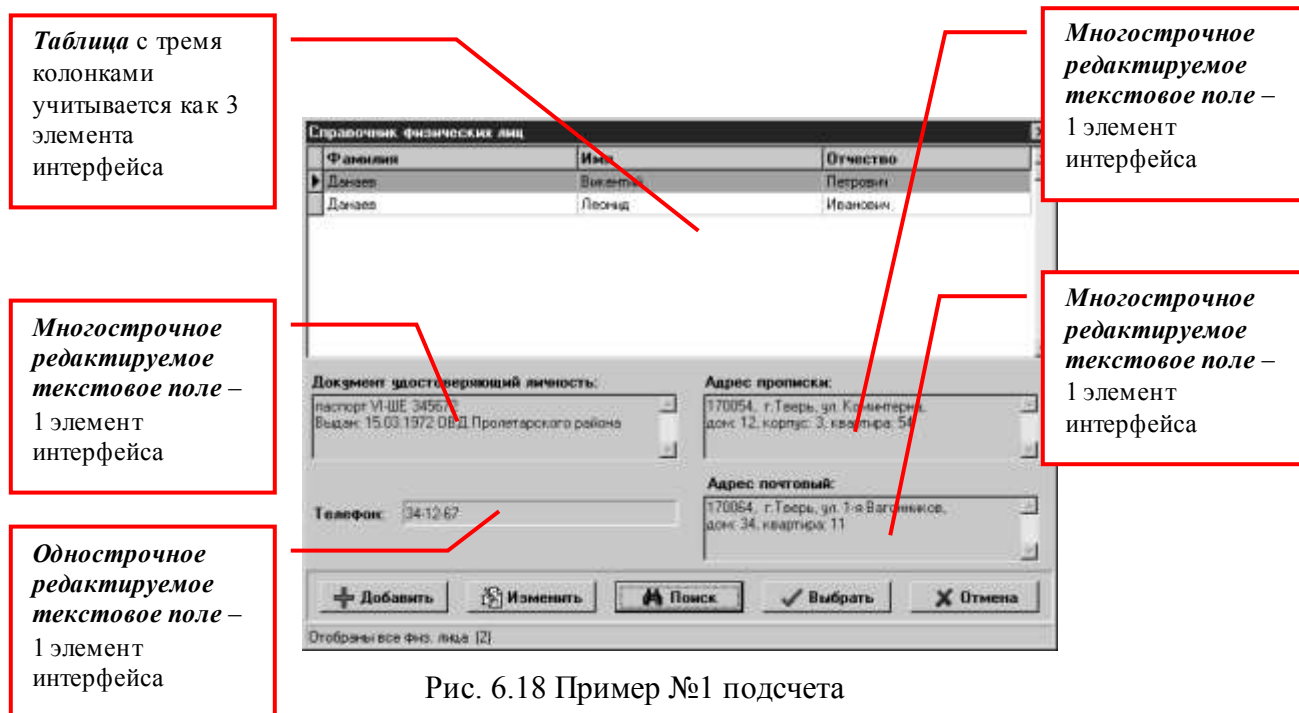
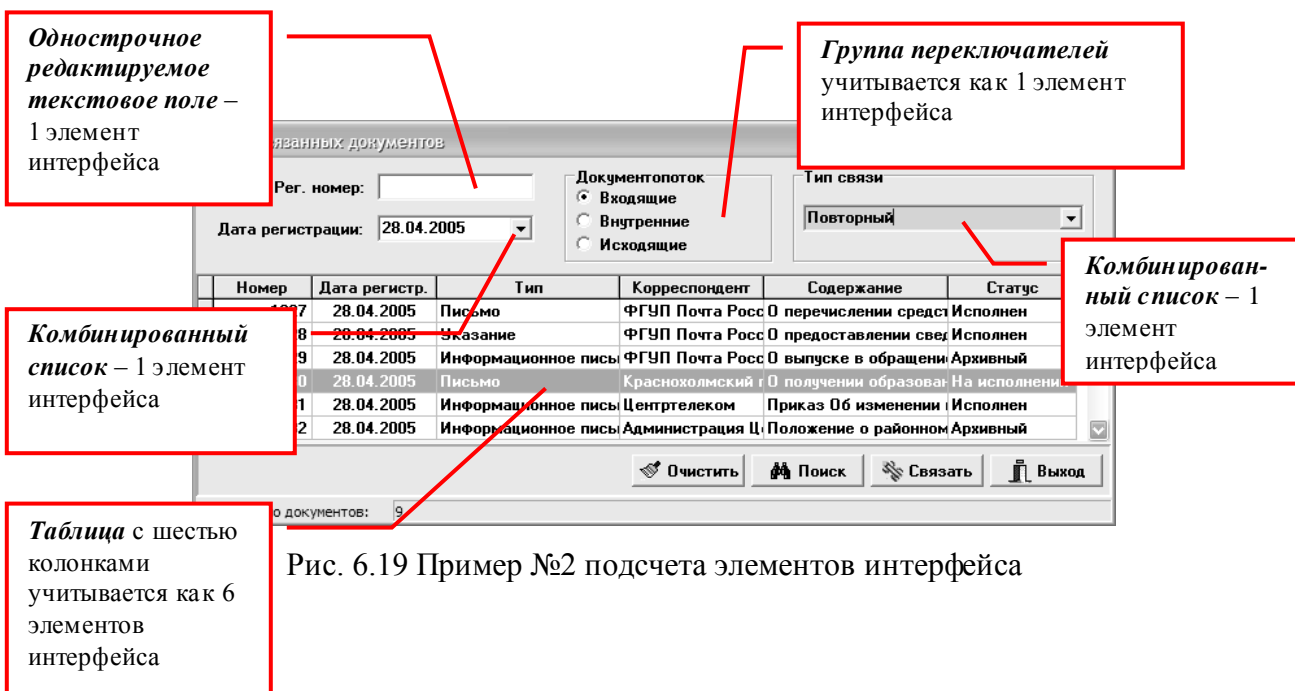
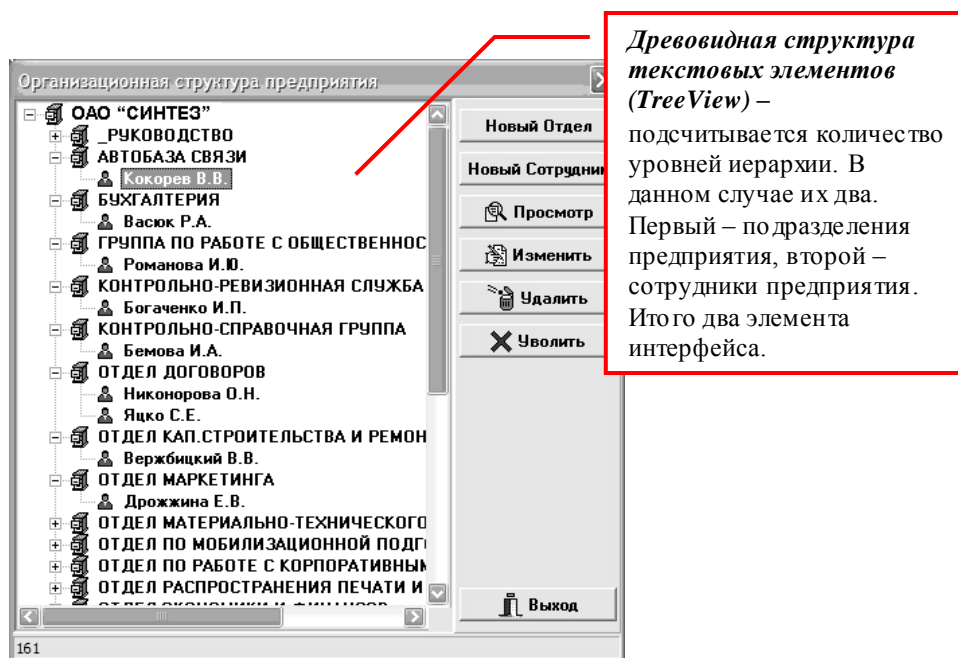


Рис. 6.18 Пример №1 подсчета элементов интерфейса

Клиентскому окну, изображенному на рис. 6.18, соответствует 7 элементов интерфейса для ввода/редактирования/просмотра данных.



Клиентскому окну, изображенному на рис. 6.19, соответствует 10 элементов интерфейса для ввода/редактирования/просмотра данных.



Клиентскому окну, изображенному на рис. 6.20 соответствует 2 элемента интерфейса для ввода/редактирования/просмотра данных.

6.1.4. Для определения функциональной сложности отчетов и других выходных документов ПС необходимо подсчитать количество их реквизитов.

Отчет «бланк» является наиболее простой формой отчета и предназначен для вывода одного набора данных в виде одного документа. При необходимости вывода нескольких наборов данных выводится один документ на каждый такой набор (рис. 6.21).

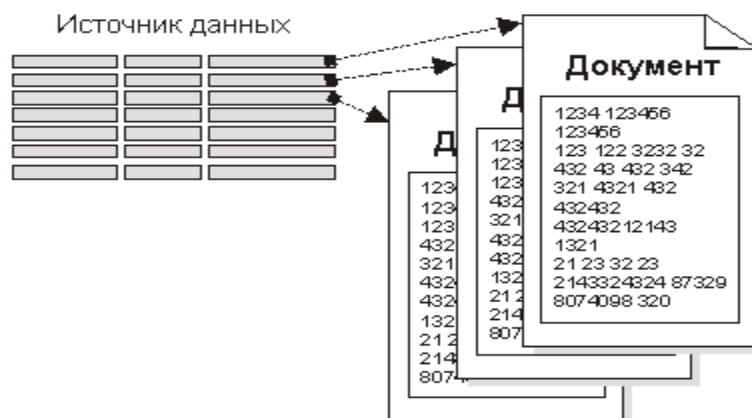


Рис. 6.21 Схема взаимодействия отчета «бланк» с внешними источниками данных

В отчете подобного типа нет повторяющихся групп данных. Пример такого отчета приведен на рис. 6.22.

Проект по разработке ПС (Каталог) № 14

Параметры проекта

Параметр	Значение
Имя проекта	Проект для примера
Тип проекта	Проект по разработке ПС (Каталог)
Версия модели	2,31
Фонд времени разработчика	21
Тип ограничения	По продолжительности этапа [мес.]
Тип структуры программного средства	ПС с набором библиотек(=1.00)
Технологии взаимодействия модулей	Объектно-ориентированные технологии (COM/DCOM, ODBC, OLEDB, ADO, ActiveX)(=0.35)

Рис. 6.22 Отчет типа «бланк»

При подсчете количества реквизитов учитываются только те, которые были извлечены из источников данных или были вычислены (преобразованы) на основе извлеченных данных. На рис. 6.22 такие реквизиты выделены цветом и таким образом, этот отчет содержит семь реквизитов.

Отчет «список» предназначен для организации сложных отчетов, позволяющих отражать различное количество однотипных блоков информации. Обычно подобные отчеты содержат повторяющиеся группы данных и группировки. Порядок формирования подобных отчетов изображен на рис. 6.23.

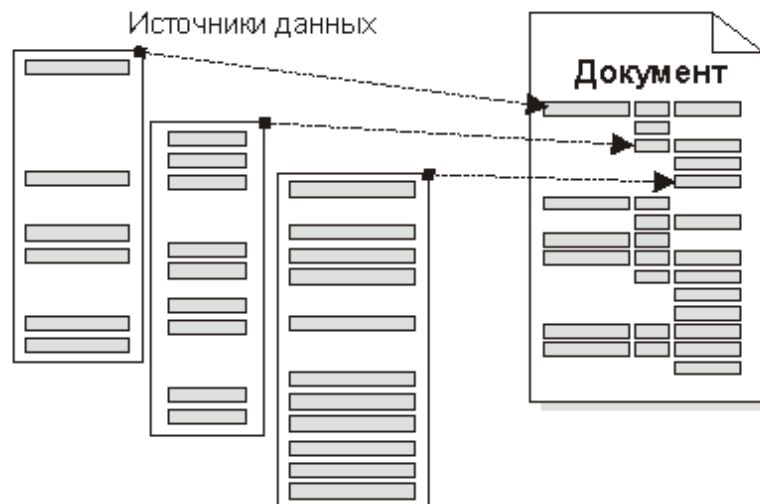


Рис. 6.23 Схема взаимодействия отчета «список» с внешними источниками данных

Пример отчета типа «список» с повторяющейся группой данных приведен на рис. 6.24.

Дерево проекта

№	Элемент	V_i (на условном языке)	r_i	$k_i [0;1]$	$K_{\text{сложн}}$	$K_{\text{ф.разр}}$	$K_{\text{опыт}}$
1.5.12	Поиск, представление и вывод информации	3500 из [1000-6000]	1	1	Высокий(=1.10)	PowerBuilder(=0.84)	Средний(=1.00)
1.5.13	Обработка записей базы данных	1750 из [500-3000]	1	1	Высокий(=1.10)	PowerBuilder(=0.84)	Средний(=1.00)
3.3.2	Преобразование формата данных	2750 из [1000-4500]	1	1	Высокий(=1.10)	PowerBuilder(=0.84)	Средний(=1.00)
3.3.3	Формирование файлов	1750 из [500-3000]	1	1	Средний(=1.00)	PowerBuilder(=0.84)	Средний(=1.00)
2.1.5	Расчет алгебраических выражений	550 из [100-1000]	1	1	Средний(=1.00)	PowerBuilder(=0.84)	Средний(=1.00)

Рис. 6.24. Отчет типа «список» с повторяющейся группой данных

В данном случае каждая строка выделенных цветом реквизитов является повторяющейся группой. Каждый такой реквизит подсчитывается только один раз. При этом учитываются только те реквизиты, которые были извлечены из источников данных или были вычислены (преобразованы) на основе извлеченных данных. Таким образом, отчет, изображенный на рис. 6.24, содержит восемь реквизитов.

Отчеты могут иметь более сложную структуру, чем те, которые приведены в примерах. Также отчеты могут содержать группировки и итоги. Однако любой отчет

можно разбить на составляющие части, соответствующие одному из типов отчетов, указанных выше. Состав реквизитов для каждой части отчета подсчитывается отдельно, затем подсчитывается их общее количество.

Вид такого отчета приведен на рис. 6.25. Аналогично предыдущему примеру повторяющиеся реквизиты учитываются только один раз. Также при подсчете необходимо учесть вычисляемые реквизиты с промежуточными и общими итогами. Все реквизиты отчета, участвующие в подсчете, выделены. Отчет, изображенный на рис. 6.25, содержит 35 реквизитов.

Оценка суммарной трудоемкости

Параметры проекта

Параметр	Значение
Имя проекта	Тестовый проект оценки суммарной трудоемкости
Описание проекта	Тест
Тип проекта	Оценка суммарной трудоемкости
Общие трудозатраты [чел.дней]	1702,12

Тип проектов: Проект по разработке ПС (Каталог)

Наименование	Версия модели	Анализ требований	Проектирование	Программирование	Тестирование	Ввод в действие ПС	Всего
development_1	2.4.1.0	3,59	45,59	283,10	25,08	3,59	360,95
development_2	2.4.1.0	2,87	36,47	226,48	20,06	2,87	288,75
development_3	2.4.1.0	4,22	53,57	332,65	29,47	4,22	424,13
Всего		10,68	135,63	842,23	74,61	10,68	1073,83

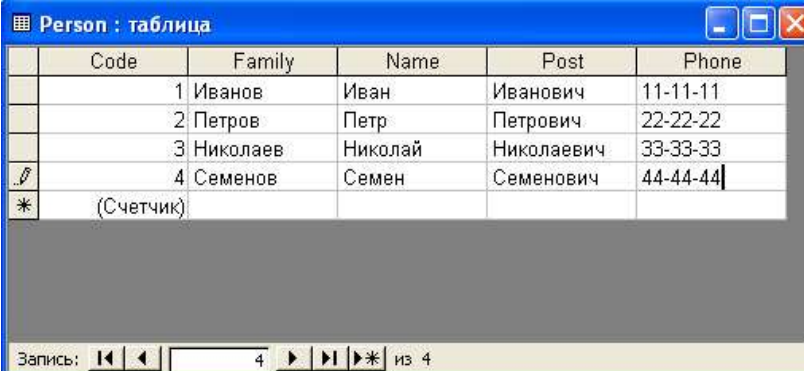
Тип проектов: Проект по сопровождению ПС

Наименование	Версия модели	Подготовка процесса	Анализ проблем и изменений	Внесение изменений в программное средство	Внесение изменений в документацию	Проверка и приемка при сопровождении	Поддержка пользователя	Всего
maintenance_1	2.4.1.0	8,47	10,93	128,81	19,10	23,22	20,65	211,18
maintenance_2	2.4.1.0	6,78	8,75	103,05	15,28	18,58	16,52	168,96
maintenance_3	2.4.1.0	9,95	12,85	151,35	22,45	27,28	24,27	248,15
Всего		25,20	32,53	383,21	56,83	69,08	61,44	628,29

Рис. 6.25 Пример сложного отчета

6.1.5. Для определения сложности файлов данных и таблиц РСУБД информационной базы ПС необходимо подсчитать количество элементов данных, которые они содержат.

В ER диаграмме, описывающей структуру базы данных, под элементами данных следует понимать атрибуты сущности, представляющей таблицу БД. Например, таблица «Person» на рис. 6.26 состоит из пяти элементов данных (содержит пять полей-столбцов: «Code», «Family», «Name», «Post», «Phone»). Поле «Code» в данном примере является первичным ключом.

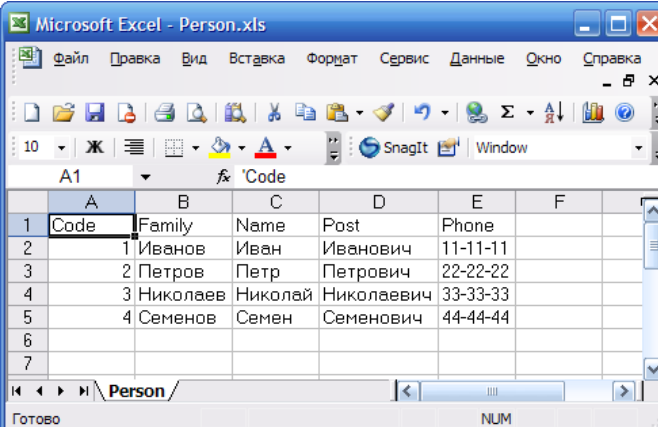


	Code	Family	Name	Post	Phone
	1	Иванов	Иван	Иванович	11-11-11
	2	Петров	Петр	Петрович	22-22-22
	3	Николаев	Николай	Николаевич	33-33-33
	4	Семенов	Семен	Семенович	44-44-44
*	(Счетчик)				

Рис. 6.26 Пример таблицы РСУБД

6.1.6. Для определения уровня функциональной сложности процедур импорта/экспорта данных необходимо подсчитать количество элементов данных в одной передаваемой записи и количество файлов данных (таблиц РСУБД), участвующих в информационном обмене.

На рис. 6.27 представлен пример экспорта данных из таблицы «Person» БД приложения с рис. 6.26 в табличный редактор Excel. Результат подсчета для данного примера: передавалось пять элементов данных (по количеству полей таблицы) из одной таблицы.



	A	B	C	D	E	F
1	Code	Family	Name	Post	Phone	
2		1 Иванов	Иван	Иванович	11-11-11	
3		2 Петров	Петр	Петрович	22-22-22	
4		3 Николаев	Николай	Николаевич	33-33-33	
5		4 Семенов	Семен	Семенович	44-44-44	
6						
7						

Рис. 6.27 Пример экспорта данных в Excel

6.2. Оценка трудоемкости разработки программного средства

Расчет трудоемкости разработки с использованием метода расчета размера ПС по элементам графического интерфейса пользователя выполняется аналогично тому, как это делается при использовании Каталога функций. При этом необходимо предварительно пересчитать объем ПС в функциональных точках в строки исходного кода. Пересчет выполняется в соответствии с пунктом 4.2.15 Методики.

Правила выбора значений коэффициентов, определяющих трудоемкость разработки ПС, аналогичны тем, которые указаны для расчета разработки ПС с использованием Каталога функций.

7. РАСЧЕТ ТРУДОЕМКОСТИ СОПРОВОЖДЕНИЯ ПРОГРАММНЫХ СРЕДСТВ

7.1. Общие положения

7.1.1. Сопровождение программных средств может выполняться опорными центрами по сопровождению, разработчиками ПС, отдельными организациями, сопровождающими ПС.

7.1.2. Расчет трудоемкости сопровождения ПС выполняется исходя из срока выполнения работ **один** год. Для расчета необходимо наличие следующих параметров:

- полный объем программного средства, передаваемого на сопровождение;
- объем доработок кода (объем кода, написанного разработчиком вручную и (или) модифицированного им);
- объем документации ПС (документация сопровождения);
- объем доработок документации (строк);
- количество установленных экземпляров ПС;
- число пользователей ПС.

7.1.3. В случае, когда рассчитать полный объем исходных текстов ПС не представляется возможным (например, отсутствуют исходные коды), то рекомендуется следующий порядок применения методов пересчета с использованием переводных коэффициентов, описанных в Методике:

1 метод. Оценка объема сопровождаемого ПС через Каталог функций с использованием переводных коэффициентов (пункт 6.1.7 Методики).

2 метод. Оценка объема сопровождаемого ПС через объем сопровождаемых бинарных модулей (пункт 6.1.8 Методики).

7.1.4. В качестве величины объема исходного кода, передаваемого на сопровождение, рекомендуется принять значение, рассчитанное для программного средства с использованием Каталога функций. Расчет объема при этом можно выполнять в соответствии с пунктами 3.2.1–3.2.5 Методики. В случае затруднения в выборе корректирующих коэффициентов их значения следует принять равными 1.00. По возможности необходимо избегать применения методов пересчета, поскольку они могут давать большую ошибку при расчете.

7.1.5. Объем кода в Методике измеряется числом строк кода на исходном языке разработки. Если ПС написано с использованием нескольких языков разработки, то для каждого языка объем оценивается отдельно, а затем суммируется. Например, если код

программы состоит на 10000 строки на ассемблере и 5000 строк на C++, то общий объем составляет 15000 строк. Использование различных языков разработки учитывается поправочным коэффициентом этапа внесения изменений в программное средство, учитывающего язык программирования и средства разработки ПС ($K_{раз}$).

Для каждого средства разработки указывается доля его участия в расчете (для данного примера соотношение составляет 0,67 для ассемблера и 0,33 для C++). При этом суммарная доля участия для всех средств разработки должна быть равна 1.

При подсчете строк кода комментарии не учитываются. Объявления считаются как одна строка кода. При этом число строк исходного кода с достаточной точностью можно определить простым подсчетом числа строк.

7.1.6. Полный объем программного средства (строк кода) включает в себя:

- объем кода написанный разработчиком;
- объем кода, сгенерированный автоматически при разработке ПС средой разработки;
- объем кода, заимствованный из библиотек;
- объем кода, используемый повторно из библиотек, покупного и коммерческого программного обеспечения и т.п.

7.1.7. Объем документации ПС выражается количеством строк. С учетом того, что помимо текста документы могут содержать схемы, диаграммы, картинки и т.п., объем документации может быть подсчитан следующим образом. Определяется число строк на странице документа (для шрифта Times New Roman размером 14, межстрочным интервалом 1,5 и форматом бумаги А4 эта величина равна 30 строк/стр.) Объем документации получается перемножением числа строк на странице и количества страниц документации (учитываются все необходимые сопроводительные и эксплуатационные документы).

В расчет объема документации не включают:

- титульный лист;
- содержание;
- список литературы.

7.1.8. Состав документации сопровождения аналогичен составу программной документации для программных средств и определен в ГОСТ 19.101-77. Документация сопровождения включает в себя:

- спецификация (содержит состав ПС и документации на нее);

- описание программного средства (содержит сведения о логической структуре и функционировании ПС);
- программа и методика испытаний (содержит требования, подлежащие проверке при испытании ПС, а также порядок и методы их контроля);
- техническое задание (содержит назначение и область применения ПС, технические, технико-экономические и специальные требования, предъявляемые к ПС, необходимые стадии и сроки разработки, виды испытаний);
- пояснительная записка (содержит схемы алгоритма, общее описание алгоритма и/или функционирования ПС, а также обоснование принятых технических и технико-экономических решений);
- эксплуатационные документы в составе:
 - а) формуляр (содержит основные характеристики ПС, комплектность и сведения об эксплуатации программы);
 - б) описание применения (содержит сведения о назначении ПС, области применения, применяемых методах, классе решаемых задач, ограничениях для применения, минимальной конфигурации технических средств);
 - в) руководство системного программиста (содержит сведения для проверки, обеспечения функционирования и настройки ПС на условия применения);
 - г) руководство программиста (содержит сведения для эксплуатации программы);
 - д) руководство пользователя (содержит сведения для обеспечения диалога пользователя с ПС в процессе работы с данным ПС);
 - е) руководство по техническому обслуживанию (содержит сведения для применения тестовых и диагностических ПС при обслуживании технических средств).

7.1.9. Объем доработок кода (строк кода) включает в себя:

- объем кода, написанный разработчиком в процессе сопровождения;
- объем кода, модифицированный разработчиком при сопровождении.

Объем доработок документации включает в себя:

- объем документации, написанный в процессе сопровождения;
- объем документации, модифицированный при сопровождении.

В случае невозможности определения объема доработок (как кода, так и документации), этот объем может быть определен экспертным методом исходя из опыта прошлых лет (если сопровождение выполняется несколько лет).

Максимальный объем доработок кода при сопровождении не должен превышать 20% от объема программного средства. Большой объем доработок связан, как правило, со значительными изменениями функциональных возможностей ПС.

7.1.10. Согласно пункту 6.2.11 Методики, общая трудоемкость сопровождения равна сумме трудоемкостей всех работ сопровождения. В процессе сопровождения могут выполняться не все работы по сопровождению ПС, а только их часть. Поэтому при оценке трудоемкости сопровождения ПС следует рассчитывать трудоемкости только тех работ, которые выполняются службой сопровождения (разработчиком ПС).

Если в процессе сопровождения программное средство не дорабатывается, а исходный код модифицируется исключительно с целью функционирования данного ПС на другой платформе (перенос на другую платформу, ОС), то расчет работы «Внесение изменений в программное средство» не выполняется. Все необходимые для переноса изменения исходного кода учитываются в стадии «Перенос».

7.2. Правила применения коэффициентов

7.2.1. При оценке трудоемкости сопровождения ПС следует обращать внимание на задаваемые значения поправочных коэффициентов. Поправочные коэффициенты могут оказывать влияние на трудоемкость нескольких работ сопровождения. На всех работах для одного и того же коэффициента устанавливается один уровень. Значение уровня при этом для разных работ различно.

7.2.2. Значение уровня коэффициентов, учитывающих механизм обновления ПС для оценки трудоемкостей работ «Подготовка процесса» ($K_{мо.пн}$), «Анализ проблем и изменений» ($K_{мо.ан}$), «Внесение изменений в документацию» ($K_{мо.вн.док}$) процесса сопровождения (пункты 6.2.2, 6.2.3 и 6.2.5 Методики соответственно) устанавливается в соответствии со следующими правилами:

- значение «Автоматическое обновление» используется, если операции установки (копирование файлов на ТС, настройка) выполняются автоматически;
- значение «Ручное обновление одного ПС без настройки» используется, если копирование файлов на ТС обновляемого ПС выполняется вручную. Настройка или перенастройка ПС после этого не требуется;
- значение «Ручное обновление и настройка одного ПС» используется, если операции установки ПС выполняются вручную, после чего выполняется настройка или перенастройка ПС;

– значение «Ручное обновление комплекса ПС без настройки» используется, если копирование файлов на ТС обновляемого комплекса ПС выполняется вручную. Настройка или перенастройка комплекса ПС после этого не требуется;

– значение «Ручное обновление и настройка комплекса ПС» используется, если все операции установки комплекса ПС выполняются вручную, после чего выполняется настройка или перенастройка ПС.

В случае, когда могут использоваться различные механизмы обновления (например, зачастую обновление выполняется автоматически, а в ряде случаев необходимо вручную выполнять какие-либо настройки), следует задавать наиболее трудоемкий вариант.

7.2.3. Значение уровня коэффициента, учитывающего наличие в фонде программ аналогов данного ПС ($K_{ан.пм}$), для оценки трудоемкости работы «Подготовка процесса» (пункт 6.2.2 Методики) устанавливается на уровень «Есть» в том случае, когда в фонде алгоритмов программ имеются ПС, выполняющие аналогичные функции, а также служба сопровождения (разработчик ПС, соисполнитель) при этом имеет опыт сопровождения ПС по аналогичной тематике или сопровождала ранее разработанные ПС. В противном случае выбирается уровень «Нет».

7.2.4. Значение уровня коэффициентов, учитывающих характер внедрения ПС для оценки трудоемкостей работ «Анализ проблем и изменений» ($K_{хв.ан}$), «Проверка и приемка при сопровождении» ($K_{хв.пс}$) процесса сопровождения (пункты 6.2.3, и 6.2.6 Методики соответственно) выбирается согласно Методики.

Уровень «Локальное внедрение ПС» используется для ПС, способного полноценно функционировать независимо от других ПС. К категории таких программных средств могут относиться, например, АРМы для различных областей применения.

Уровень «Внедрение ПС в составе комплекса невзаимосвязанных ПС» используется для ПС, поставляемого как независимый элемент какого-либо программного средства. Программное средство при этом должно функционировать независимо от других ПС, т.е. не должно использовать их функциональные возможности. Но при этом оно может иметь с ним общие ресурсы, библиотеки и т.д.

Уровень «Внедрение ПС в составе комплекса взаимосвязанных ПС» используется для ПС, поставляемого как составной элемент (компонент) какого-либо программного средства. Программное средство при этом функционирует совместно с другими ПС (использует их функциональные возможности).

Уровень «Внедрение ПС как компонентов разрабатываемой или функционирующей системы обработки информации (СОИ), связанной с другими компонентами» используется для ПС, используемых совместно или разрабатываемых для работы существующих систем. К таким ПС можно отнести программы, используемые для контроля, диагностики работы систем, расширения их функциональных возможностей и т.д.

7.2.5. Значение уровня коэффициента, учитывающего язык программирования и средства разработки ПС ($K_{раз}$) для оценки трудоемкости работы «Внесение изменений в программное средство» сопровождения ПС (пункт 6.2.4 Методики) устанавливается согласно требованиям Методики. Если необходимое средство разработки отсутствует, то порядок действий такой же, как в пункте 4.3.3 настоящих Рекомендаций.

7.2.6. Значение уровня сложности ПС для оценки трудоемкостей работ «Внесение изменений в программное средство» ($K_{сл.вн.пр}$), «Внесение изменений в документацию» ($K_{сл.вн.док}$), «Проверка и приемка при сопровождении» ($K_{сл.пс}$) процесса сопровождения (пункты 6.2.4, 6.2.5, и 6.2.6 Методики соответственно) определяется в соответствии с таблицей 5.1 Приложения 5 Методики. Данная таблица содержит описания характеристик программных средств, по которым следует выбирать уровень сложности. При этом функциональные характеристики ПС устанавливаются в соответствии с документацией сопровождения. Выбирается значение L_v в соответствии с которым по таблицам 5.13, 5.16 и 5.18 Приложения 5 Методики определяется значение коэффициента уровня сложности для работ сопровождения.

Необходимо учесть, что если возможностям ПС соответствуют несколько уровней сложности, то выбирается наиболее высокий уровень. Например, если в соответствии с документацией программное средство выполняет сложное преобразование и реструктурирование информации в базе данных (высокий уровень сложности), но при этом его интерфейс строится на базе нескольких стандартных элементов управления (низкий уровень сложности), то уровень сложности программного средства в целом необходимо принять как высокий.

7.2.7. Значение уровня коэффициента, учитывающего полноту тестирования отдельных ПС или комплекса ПС ($K_{тест.пс}$) для оценки трудоемкости работы «Проверка и приемка при сопровождении» сопровождения ПС (пункт 6.2.6 Методики) устанавливается в соответствии с Методикой. Для этого учитывается характеристика полноты тестирования сопровождаемого программного средства или комплекса программных средств.

7.2.8. Значение уровня коэффициентов, учитывающих характер поставки ПС для оценки трудоемкостей работ «Перенос» ($K_{хп.п}$), «Снятие с эксплуатации» ($K_{хп.сн}$) процесса сопровождения (пункты 6.2.7 и 6.2.8 Методики соответственно) выбирается согласно Методики.

Уровень «Локальная поставка стандартного комплекта ПС или поставка в комплекте с несвязанными ПС» следует установить:

- если выполняется поставка ПС в базовом варианте (ориентированном на потребности большинства пользователей данного класса ПО);
- если выполняется поставка комплекта ПС, в котором каждое отдельное ПС может функционировать независимо от других, входящих в состав данного комплекта.

Уровень «Локальная поставка нестандартного комплекта ПС» следует установить в случае, если выполняется поставка ПС в расширенном (по сравнению с базовым) варианте, ориентированном на потребности конкретного пользователя. В данном случае в комплекте ПС дополнительно могут поставляться различные тестовые примеры, средства отладки и настройки, предназначенные для оптимизации комплекта ПС под нужды конкретного пользователя либо небольшой группы пользователей.

Уровень «Поставка стандартного комплекта ПС в составе комплекса взаимосвязанных ПС» следует установить в случае, если выполняется поставка ПС в расширенном (по сравнению с базовым) варианте, ориентированном на потребности конкретного пользователя, в котором отдельные ПС могут функционировать только совместно друг с другом (не могут использоваться самостоятельно).

Уровень «Поставка нестандартного комплекта ПС в составе комплекса взаимосвязанных ПС» следует установить в случае, если выполняется поставка ПС в расширенном (по сравнению с базовым) варианте, ориентированным на потребности конкретного пользователя, в котором отдельные ПС могут функционировать только совместно друг с другом (не могут использоваться самостоятельно).

Уровень «Поставка комплекта ПС как компонента системы обработки информации (СОИ), не связанного с другими компонентами» следует установить в случае, если выполняется поставка комплекта ПС как части функционирующей сложной системы, в котором каждое отдельное ПС может функционировать независимо от других, входящих в состав данного комплекта.

Уровень «Поставка комплекта ПС как компонента СОИ, связанного с другими компонентами» следует установить в случае если выполняется поставка комплекта ПС как части функционирующей сложной системы, в котором отдельные ПС могут функционировать только совместно друг с другом и не могут использоваться самостоятельно.

Уровень «Поставка комплекта ПС как компонента СОИ, связанного с другими компонентами поставки и с уже функционирующими компонентами СОИ» следует установить в случае, если выполняется поставка комплекта ПС как части функционирующей сложной системы, в котором отдельные ПС могут функционировать только совместно друг с другом и не могут использоваться самостоятельно. А также тех, работа, которых оказывает влияние на другие компоненты СОИ, и на которые влияет работа других компонентов СОИ.

7.2.9. Значение уровня коэффициентов, учитывающих количество модификаций программного средства, для оценки трудоемкостей работ «Подготовка процесса» ($K_{\text{мод.пт}}$), «Анализ проблем и изменений» ($K_{\text{мод.ан}}$), «Проверка и приемка при сопровождении» ($K_{\text{мод.пс}}$), «Перенос» ($K_{\text{мод.л}}$), «Формирование и ведение ФАП» ($K_{\text{мод.вф}}$), «Поддержка пользователя» ($K_{\text{мод.по}}$) процесса сопровождения устанавливается исходя из среднего количества модификаций рабочей версии данного ПС за год. Если количество модификаций ПС в планируемом году определить не представляется возможным, то используются данные за прошлый год (годы).

7.2.10. Значение уровня коэффициентов, учитывающих периодичность использования программного средства, для оценки трудоемкостей работ «Анализ проблем и изменений» ($K_{\text{пер.ан}}$), «Внесение изменений в программное средство» ($K_{\text{пер.вн.пр}}$), «Внесение изменений в документацию» ($K_{\text{пер.вн.док}}$), «Перенос» ($K_{\text{пер.л}}$), «Поддержка пользователя» ($K_{\text{пер.по}}$) процесса сопровождения устанавливается согласно значениям, приведенным в таблицах 5.10, 5.14, 5.17, 5.25, 5.29 Приложения 5 Методики.

7.2.11. Значение уровня коэффициентов, характеризующих степень участия службы сопровождения в разработке ПС для оценки трудоемкостей работ «Подготовка процесса» ($K_{\text{уч.пт}}$), «Анализ проблем и изменений» ($K_{\text{уч.ан}}$), «Внесение изменений в программное средство» ($K_{\text{уч.вн.пр}}$), «Проверка и приемка при сопровождении» ($K_{\text{уч.пс}}$) процесса сопровождения (пункты 6.2.2, 6.2.3, 6.2.4 и 6.2.6 Методики соответственно) выбирается согласно Методике.

При выборе уровня значения коэффициента следует помнить, что если в сопровождении принимает участие организация-разработчик данного ПС, то она становится службой сопровождения или ее частью, т.е. в этой ситуации выбирается уровень «ССо разрабатывала ПС или значительную его часть» (минимальное значение коэффициента).

Если в сопровождении принимает участие организация, являвшаяся соисполнителем работ по разработке (т.е. также становится службой сопровождения или ее частью), то для нее выбирается уровень «ССо участвовала в разработке ПС на правах соисполнителя».

Уровень «ССо в разработке ПС не участвовала, но имела информацию о ходе разработки и принимала участие в испытаниях ПС» выбирается тогда, когда сопровождение выполняется организацией-разработчиком, обладающей информацией о ходе разработки программного средства и принимавшим участие в его испытании.

Значение уровня «ССо в разработке ПС не участвовала. Информации о разработке до момента сдачи в фонд не имелось» выбирается в том случае, если сопровождение выполняется организацией, не имевшей отношение к разработке ПС, передаваемого ей на сопровождение.

8. РАСЧЕТ ТРУДОЕМКОСТИ ЭКСПЛУАТАЦИИ ПРОГРАММНЫХ СРЕДСТВ

8.1. Общие положения

8.1.1. Эксплуатация программных средств выполняется специалистами подразделения информатизации территориальных учреждений Кредитных организаций с целью поддержки функционирования ПС в эксплуатационной среде без внесения изменений в программный код.

8.1.2. Расчет трудоемкости эксплуатации ПС выполняется исходя из срока выполнения работ **один** год. Для расчета необходимо наличие следующих параметров:

- объем документации ПС (только эксплуатационная документация);
- количество установленных экземпляров ПС;
- число пользователей ПС.

8.1.3. Правила подсчета объема документации представлены в пункте 7.1.7 Рекомендаций.

8.1.4. Эксплуатационная документация содержит сведения для обеспечения функционирования и эксплуатации программы. Состав эксплуатационной документации для программных средств определен в ГОСТ 19.101-77. Эксплуатационная документация включает в себя:

- формуляр (содержит основные характеристики ПС, комплектность и сведения об эксплуатации ПС);
- описание применения (содержит сведения о назначении ПС, области применения, применяемых методах, классе решаемых задач, ограничениях для применения, минимальной конфигурации технических средств);
- руководство системного программиста (содержит сведения для проверки, обеспечения функционирования и настройки ПС на условия применения);
- руководство программиста (содержит сведения для эксплуатации ПС);
- руководство пользователя (содержит сведения для обеспечения диалога пользователя с ПС в процессе работы с данным ПС);
- руководство по техническому обслуживанию (содержит сведения для применения тестовых и диагностических ПС при обслуживании технических средств).

8.1.5. Согласно пункту 7.2.8 Методики, общая трудоемкость эксплуатации равна сумме трудоемкостей всех работ эксплуатации. В том случае, если выполняются не все

работы по эксплуатации ПС, следует рассчитывать и суммировать трудоемкости только тех работ, которые выполняются.

8.1.6. При оценке трудоемкости эксплуатации ПС следует обращать внимание на задаваемые значения поправочных коэффициентов (см. п. 8.2 Рекомендаций). Поправочные коэффициенты могут оказывать влияние на трудоемкость нескольких работ эксплуатации. На всех работах для одного и того же коэффициента устанавливается один уровень. Значение уровня при этом для разных работ различно.

8.1.7. Трудоемкость эксплуатации общесистемных программных средств (операционные системы ТС, офисные ПС, сетевое программное обеспечение и прочие приобретаемые ПС), которые не тестируются и не модифицируются службами Кредитных организаций, выполняется экспертным путем. В качестве величины этой трудоемкости рекомендуется брать 10% от общего фонда рабочего времени специалистов подразделения информатизации ТУ Кредитных организаций.

8.1.8. При оценке суммарной трудоемкости эксплуатации программных средств в ТУ Кредитных организаций необходимо включать в расчет все необходимые работы по эксплуатации всех ПС. Величина общей трудоемкости определяется как сумма трудоемкостей работ по эксплуатации всех ПС, включаемых в расчет. АРМ обеспечивает итоговый расчет по всем ПС.

8.1.9. Под количеством установленных экземпляров ПС понимается число установленных копий данного программного средства. Для ПС, функционирующего в рамках одной рабочей станции (такowymi, например, являются многие АРМ), под одним экземпляром следует понимать копию, установленную на одной рабочей станции. Если ПС является распределенным и(или) функционирует в рамках нескольких рабочих станций, объединенных локальной сетью, то одним экземпляром такого программного обеспечения является совокупность серверной части (возможно, состоящей в свою очередь из нескольких модулей, реализующих различные сервисы) и одной или нескольких клиентских частей, взаимодействующих с данной серверной.

8.2. Правила применения коэффициентов

8.2.1. Уровни коэффициентов, учитывающих механизм обновления ($K_{мо.пп}$, $K_{мо.эп}$), наличие в фонде программ аналогов данного ПС ($K_{ан.пп}$), сложность ($K_{сл.пт}$, $K_{сл.эп}$) и полноту тестирования отдельных ПС или комплекса ПС ($K_{те.эп}$) на работах «Подготовка процесса» и «Эксплуатационные испытания» процесса эксплуатации (пункты 7.2.2 и 7.2.3 Методики

соответственно) выбираются с учетом тех же требований, что и при сопровождении ПС (см. пункты 7.2.1, 7.2.3, 7.2.6 и 7.2.7 Рекомендаций).

Для коэффициента уровня сложности ПС при эксплуатации используются таблицы 6.1, 6.4, 6.8 Приложения 6 Методики.

Уровни коэффициентов, учитывающих количество модификаций программного средства ($K_{\text{мод.лп}}$, $K_{\text{мод.эп}}$, $K_{\text{мод.э}}$, $K_{\text{мод.вф}}$, $K_{\text{мод.ло}}$) и периодичность использования программного средства ($K_{\text{пер.э}}$, $K_{\text{пер.ло}}$) на работах «Подготовка процесса», «Эксплуатационные испытания», «Эксплуатация ПС», «Формирование и ведение ФАП» и «Поддержка пользователя» процесса эксплуатации (пункты 7.2.2, 7.2.3, 7.2.4, 7.2.5 и 7.2.6 Методики соответственно) выбираются с учетом тех же требований, что и при сопровождении ПС (см. пункты 7.2.9, 7.2.10 Рекомендаций).

8.2.2. Уровень коэффициента, учитывающего тип используемой базы данных ($K_{\text{БД}}$) работ «Подготовка процесса» и «Эксплуатационные испытания» процесса эксплуатации выбирается в зависимости от используемого типа БД (пункты 7.2.2 и 7.2.3 Методики).

9. КОММЕНТАРИИ К ОЦЕНКЕ ЗАТРАТ ВРЕМЕНИ И ЧИСЛЕННОСТИ СПЕЦИАЛИСТОВ

9.1. Трудоемкость разработки ПС рассчитывается для всех подпроцессов разработки. Трудоемкость сопровождения и эксплуатации может рассчитываться для всех или некоторых (выполняемых) работ процесса.

Трудоемкость сопровождения и эксплуатации ПС рассчитывается исходя из сроков выполнения работ **один** год, включая все возможные повторения работ в рамках этого срока.

9.2. Методика предусматривает возможность оценки трудоемкости сопровождения и эксплуатации ПС по отдельным кварталам в течение года (пункт 7.3 Методики). При необходимости, аналогичным образом допускается пересчет на любые другие сроки, суммарная продолжительность которых составляет один год.

9.3. Пересчет трудоемкости сопровождения и эксплуатации на сроки, превышающие один год, данной Методикой не предусматривается.

9.4. В качестве значения фонда времени (число рабочих дней в месяце) в расчетах и при переводе трудоемкости из чел-дней в чел-месяцы рекомендуется величина 21 (день/мес.). Для получения более точной величины необходимо выполнить следующие шаги:

1 шаг. Подсчитать количество рабочих дней n_1 за период выполнения работ по разработке, сопровождению и эксплуатации ПС.

2 шаг. Подсчитать количество рабочих дней n в соответствующем году.

3 шаг. Определить уточненный срок выполнения работ в месяцах по формуле:

$$t = 12 \cdot \frac{n_1}{n} \text{ [мес.]} \quad (8.1)$$

4 шаг. Определить среднее количество рабочих дней в месяце D для расчета трудоемкости в рассматриваемом периоде (с округлением до целого числа):

$$D = \frac{n_1}{t} \text{ [дней]} \quad (8.2)$$

9.5. Методика предусматривает расчет трудоемкости сопровождения и эксплуатации ПС для случая, когда работы выполняются 8 часов в сутки, 5 дней в неделю (односменный режим работы в течение рабочей недели). Многосменный режим работы не учитывается. Методика также не учитывает случаи, когда одним специалистом подразделения информатизации ТУ Кредитных организаций выполняются работы по сопровождению или

эксплуатации нескольких ПС одновременно, а также многосменный режим работы. Данные режимы работы учитываются самостоятельно в каждом территориальном учреждении Кредитных организаций.

10. АНАЛИЗ ПОЛУЧАЕМЫХ РЕЗУЛЬТАТОВ

10.1. Общие сведения

10.1.1. Анализ получаемых результатов позволяет оценить степень приемлемости получаемых результатов при оценке трудоемкости разработки и сопровождения ПС, выявить причины получения неадекватных результатов, а также определить направления совершенствования Методики (дополнительная калибровка модели, пополнение состава используемых факторов, влияющих на трудоемкость и др.).

10.1.2. В основе предлагаемого метода анализа лежит определение численного значения допустимой ошибки вычисления, в пределах которого расчет можно признать достоверным. Это значение определяется исходя из характеристик самой модели и статистической выборки, используемой для построения модели. Поэтому данный метод позволяет оценить достоверность модели расчета относительно фактических данных и определить пути корректировки модели.

10.1.3. Сравнительный анализ значений трудоемкости производится на основе сопоставления планируемых и расчетных величин.

Планируемое значение трудоемкости определяется экспертным путем с учетом имеющегося опыта или накопленной статистики для разработанных ПС (компонентов АС) подобного типа.

Расчетное значение трудоемкости находится для разработки и сопровождения ПС (в соответствии с пунктами 3.2 или 4.2 Методики).

10.1.4. Сопоставление планируемых и расчетных значений производится путем расчета относительной ошибки вычисления трудоемкости. Относительная ошибка находится по формуле:

$$\delta = \frac{|T_{\text{факт}} - T_{\text{расч}}|}{|T_{\text{факт}}|}, \quad (9.1)$$

где $T_{\text{факт}}$ – планируемое значение трудоемкости, определяемое экспертным путем;

$T_{\text{расч}}$ – расчетное значение трудоемкости разработки или сопровождения, рассчитываемое в соответствии с пунктом 3.2 или 4.2 Методики.

Относительная ошибка вычисления трудоемкости рассчитывается для каждого подпроцесса разработки и работы сопровождения.

10.1.5. Полученная величина относительной ошибки оценивается по шкале допустимой величины. Такая шкала построена для каждого подпроцесса разработки и работы сопровождения с учетом информации о расчетах, заложенных в модель расчета. Чем ближе относительная погрешность стремится к нулю, тем более точно расчетное значение трудоемкости совпадает с фактическим.

Шкалы для оценки точности проведенного расчета трудоемкости приведены в Приложении 9 Рекомендаций.

10.1.6. Шкала состоит из трех зон:

- $(0 - \delta_1)$ – допустимая ошибка (зеленая зона); можем считать оценку трудоемкости выполненной с достаточной точностью;
- $(\delta_1 - \delta_2)$ – условно допустимая ошибка (желтая зона); полученная величина трудоемкости объяснима моделью, однако такой расчет все же дает большие отклонения;
- больше δ_2 – недопустимая ошибка (красная зона); означает большую неточность при выполнении расчетов.

При попадании величины ошибки в желтую или красную зоны необходимо выполнить проверку выбранных значений коэффициентов.

10.2. Порядок расчета

10.2.1. Расчет трудоемкости разработки и сопровождения ПС выполняется в соответствии с пунктами 3.2 или 4.2 Методики.

10.2.2. Планируемое значение трудоемкости определяется экспертным путем с учетом имеющегося опыта либо накопленной статистики для разработанных ПС (компонентов АС) подобного типа.

10.2.3. Вычисляется величина относительной ошибки расчета по формуле (9.1) для каждого подпроцесса разработки либо работы сопровождения ПС.

10.2.4. Полученные величины относительных погрешностей оцениваются по шкале (Приложение 9 Рекомендаций), приведенной в виде таблиц. В зависимости от процесса (разработка, сопровождение) выбирается соответствующая таблица. В зависимости от подпроцесса (либо работы) и порядка величины трудоемкости (планируемой) выбирается соответствующая строка таблицы.

Если величины относительных погрешностей для расчетных значений всех подпроцессов разработки или работ сопровождения рассматриваемого программного

средства попадут в зеленую зону шкалы, то полученные расчетные значения трудоемкостей рассматриваемых работ можно считать достоверными.

Для всех значений трудоемкости, попавших в желтую или красную зону шкалы, необходимо:

- 1) Проверить правильность выбора уровней коэффициентов с учетом требований, установленных в Рекомендациях по применению Методики расчета.
- 2) Проверить достоверность параметров модели (величина объема ПС, документации, доработок, количество установленных экземпляров ПС, число пользователей) с учетом требований, установленных в Рекомендациях по применению Методики расчета.

ПРИЛОЖЕНИЕ 1
КАТАЛОГ ФУНКЦИЙ ПРОГРАММНЫХ СРЕДСТВ, СОДЕРЖАЩИЙ
РЕКОМЕНДУЕМЫЕ ОБЪЕМЫ ФУНКЦИЙ

Таблица 1. Каталог функций

№	Наименование типа, класса; наименование (содержание) функции	Комментарий (Обоснование выбора)	Рекомендуемый объем функции ПС (строк исходного текста на условном языке)
1.	Управляющие операции		
1.1.	Информационная безопасность		
1.1.1.	Реализация криптографических алгоритмов (<i>один алгоритм</i>)	Функции, обеспечивающие обратимый способ преобразования информации с целью ее защиты.	550*
1.1.2.	Обеспечение безопасности передачи сообщений и обмена данными (шифрование)	Функции, обеспечивающие обратимый способ преобразования информации с целью ее защиты в канале передачи данных.	850
1.1.3.	Организация криптозащиты с помощью ПТК Криптоцентр		5500
1.1.4.	Контроль целостности ПО	Функции, обеспечивающие контроль и защиту программных файлов от модификации в процессе эксплуатации ПС.	3000
1.1.5.	Реализация базовых функций информационной безопасности в соответствии с нормативными документами Кредитных организаций		10500**
1.1.6.	Поддержка дискреционного контроля доступа	Реализация и обеспечение дискреционного контроля доступа - разграничение доступа между поименованными субъектами и поименованными объектами доступа.	3500

1.1.7.	Поддержка мандатного контроля доступа	Реализация и обеспечение мандатного контроля доступа - разграничение доступа субъектов к объектам, основанное на характеризуемой меткой конфиденциальности информации, содержащейся в объектах, и официальном разрешении (допуске) субъектов обращаться к информации такого уровня конфиденциальности.	3500
1.1.8.	Создание, редактирование, удаление пользователей (группы пользователей)		3250
1.1.9.	Установка прав доступа к объекту	Функции, предоставляющие разграничение прав доступа к объекту.	1500
1.1.10.	Ведение протокола доступа		3000
1.1.11.	Настройка политики безопасности	Настройка параметров процедур и интерфейсов задач, связанных с обеспечением информационной безопасности.	1500
1.1.12.	Ведение журналов отказа		1250
1.1.13.	Взаимодействие с ПО СКЗИ	Взаимодействие с ПО СКЗИ Интеграция СКЗИ в прикладную программу в соответствии с требованиями, указанными в документации на данные СКЗИ.	350
1.2.	<i>Реализация интерфейсов между программными средствами</i>		
1.2.1.	Взаимодействие с другим ПО	Процессы, которые посылают данные или управляющую информацию за пределы приложения.	750
1.2.2.	Взаимодействие с транспортной (почтовой) системой	Функции, обеспечивающие прием и передачу информации через почтовую систему.	2500
1.2.3.	Реализация ПО обмена информацией между ЦОИ и клиентами (WEB-сервер)		9500**
1.2.4.	Организация доступа к динамическим информационным ресурсам через Intranet Кредитных организаций	Организация работы сотрудников Кредитных организаций через Корпоративный портал Интранет Кредитных организаций.	12500**

1.2.5.	Организация регулярно актуализируемого статистического контента для доступа через Intranet Кредитных организаций	Реализация процедур организации доступа пользователей к сформированному регулярно актуализируемому статистическому контенту через Intranet Кредитных организаций.	12500**
1.3.	<i>Выполнение регламентных операций</i>		
1.3.1.	Пусковое решение		750
1.3.2.	Реализация процедур обновления версий ПО		1500
1.4.	<i>Реализация взаимосвязи программных средств и компонентов</i>		
1.4.1.	Сетевая передача команд и сообщений	Передача команд и сообщений между модулями ПС по сети.	550
1.4.2.	Реализация связи между распределенными приложениями с использованием стандартных транспортных средств	Реализация функций, позволяющие использовать для передачи информации стандартные транспортные средства, такие как MSMQ	550
1.4.3.	Реализация связи между распределенными приложениями на основе сетевых интерфейсов низкого уровня	Реализация функций, позволяющие использовать для передачи информации сетевые интерфейсы низкого уровня, такие как sockets	350
1.4.4.	Удаленная доставка информации с подтверждением получения		350
1.4.5.	Вызов удаленных процедур (<i>единичный вызов</i>)	Использование сервиса RPC	750
1.4.6.	Управление файлами, доступом к файлам и передачей файлов между удаленными и разнородными файловыми системами	Реализация функций для разграничения доступа к файлам и их перемещению в рамках сети.	350
1.4.7.	Обработка сообщений	Обработка отдельных сообщений, передаваемых данному компоненту ПС от другого компонента.	105
1.4.8.	Обработка распределенных транзакций	Обработка сообщений компонентом ПС в рамках транзакции.	155
1.4.9.	Реализация средства контроля состояния распределенной сети однородных компонент	Реализация функций, позволяющих описывать и управлять состоянием компонентов распределенного в сети ПС.	150
1.4.10.	Доступ к общей памяти: - в рамках одной машины - в рамках вычислительной сети	Распределение общей памяти (оперативной, виртуальной).	30 105
1.4.11.	Ведение журнала обращений к распределенному ПС		255
1.5.	<i>Создание и поддержка аналитических БД***</i>		

1.5.1.	Формирование физической структуры аналитических БД (<i>на одну БД</i>)	Проектирование структуры таблиц, формирование логической и концептуальной моделей данных, формирование и заполнение справочников.	4500
1.5.2.	Формирование многомерной структуры аналитических БД (<i>на одну БД</i>)	Формирование многомерной модели данных, формирование и заполнение репозитория системы.	4500
1.5.3.	Описание связей между аналитическими базами данных	Описание связи аналитических БД в части исходных данных и аналитических показателей.	1750
1.5.4.	Формирование запросов к аналитическим БД (<i>один запрос</i>)	Формирование многомерных и реляционных запросов к БД.	45*
1.5.5.	Ведение журнала операций с аналитическими БД	Формирование журнала регистрации операций с многомерными структурами БД.	1500
1.5.6.	Формирование ETL-процедур	Формирование процедур преобразования и загрузки информации, приведения информации к общей структуре БД и агрегация данных по всем видам мониторинга предприятий.	8500
1.5.7.	Протоколирование ETL-процедур		3750
1.5.8.	Функции настройки аналитических БД	Настройка окружения работы аналитической БД (настройка соединений, пересоздание представлений, синонимов, процедур, планировщиков задач).	1750
1.5.9.	Модификация аналитических БД (<i>один запрос</i>)	Модификация БД со справочной информацией в связи с добавлением новой информации.	225*
1.5.10.	Контроль целостности и восстановление аналитических БД	Функции поддержки баз данных в актуальном состоянии, резервного копирования, восстановления данных в случае сбоев.	750
1.5.11.	Прикладное администрирование аналитических БД	Организация среды выполнения процедур прикладного администрирования службой эксплуатации для наполнения и ведения аналитической БД.	3000**
1.5.12.	Поиск, представление и вывод информации	Организация поиска информации в базах данных, вывода информации на экранных формах	3500
1.5.13.	Обработка записей базы данных		1750
2.	Вычислительные операции		
2.1.	Ввод и обработка данных		

2.1.1.	Ввод данных первичных документов в интерактивном режиме	Регистрация документов, расчет значений реквизитов, ввод дополнительной информации, обработка входной информации.	5500
2.1.2.	Ввод данных по нарушениям в интерактивном режиме		3000
2.1.3.	Ввод и первичный контроль документов из файлов	Ввод, контроль полноты и корректности введенных данных.	3750
2.1.4.	Логический, синтаксический и номенклатурный контроль данных		3750
2.1.5.	Расчет алгебраических выражений	Выполнение арифметических вычислений	550
2.1.6.	Статистическая обработка данных	Реализация методов статистической обработки информации	3500
2.1.7.	Обработка протоколов		2250
2.1.8.	Обработка документов начального авизования		8500
2.1.9.	Ввод прочих данных	Чтение исходной информации из различных источников (либо ее интерактивный ввод) с целью получения данных, пригодных для дальнейшей обработки.	2500
2.1.10.	Обработка входной информации (прочих данных)	Первоначальная обработка данных с целью преобразования ее к виду, пригодному для дальнейшего использования (статистическая обработка, выполнение вычислений по данным и т.п.)	3750
2.2.	Составление и анализ балансов		
2.2.1.	Составление балансов		750**

2.2.2.	Анализ балансов	Включает в себя проведение проверок: -корсчета банка (сравнение валютного корсчета с нормативом; проверка наличия средств на рублевом корсчете); -кассы (проверка наличия валюты и рублей); - вкладов; - кредитов; - депозитов юридических лиц; - остатков на расчетных счетах клиентов; - собственных акций банков; - уставного капитала; -резервов в ЦБ; - накопительных счетов при выпуске акций; - просроченные кредиты; просроченные проценты по кредитам; - межбанковские кредиты; - прочие активы (средства, вложенные в уставные капиталы предприятия; основные средства; МБП) и т.п.	2250**
2.2.3.	Выполнение баланса и оборотов КО за месяц		3000**
2.2.4.	Выполнение ежедневного баланса		1750**
2.2.5.	Корректировка балансов		350**
2.2.6.	Пересчет балансов		2500**
2.3.	<i>Проведение экономического анализа</i>		
2.3.1.	Анализ безналичных расчетов в платежных системах		3000**
2.3.2.	Анализ денежной массы в обращении по районам	Проведение количественного (объем) и качественного (покупной состав) анализов денежной массы.	5500**
2.3.3.	Анализ курсов валют, устанавливаемых ЦБ	Контроль разницы курсов валют и проведение анализа прибыли.	4000**
2.3.4.	Анализ курсов основной валюты в обменных пунктах КО		7500**
2.3.5.	Анализ событий нефинансового характера		700**
2.4.	<i>Выполнение задач планирования</i>		
2.4.1.	Планирование инспекторских проверок		1750**
2.4.2.	Планирование эмиссии	Организация выдачи собственных акций и векселей.	4500**
2.4.3.	Планирование прочих задач		2400**
2.5.	<i>Мониторинг</i>		

2.5.1.	Мониторинг отчетности КО	Организация наблюдения за показателями, характеризующими деятельность КО, позволяющее оперативно выявлять существенные изменения, прослеживать интенсивность процессов, происходящих в КО.	6500**
2.5.2.	Мониторинг отчетности ТУ	Организация наблюдения за показателями, характеризующими деятельность ТУ, позволяющее оперативно выявлять существенные изменения, прослеживать интенсивность процессов, происходящих в ТУ.	6500**
2.6.	<i>Формирование и обработка аналитических показателей</i>		
2.6.1.	Расчет экономических показателей	Расчет синтетических (производных, интегральных, агрегированных) экономических показателей на основе первичных показателей.	10000**
2.6.2.	Экономический и финансовый анализ	Проведение различных видов исследований финансовых характеристик объекта. Выявление тенденций и основных факторов, влияющих на развитие различных экономических объектов или протекание экономических процессов.	15000**
2.6.3.	Моделирование и прогнозирование	Формирование различных видов экономико-математических моделей объектов для объяснения поведения объекта, проведения прогнозных вариантных расчетов либо поиска оптимального управления объектом.	25000**
2.6.4.	Ведение показателей системы, определяющих ее штатное состояние функционирования		
3.	Операции, зависящие от аппаратуры		
3.1.	<i>Прием документов от различных источников</i>		

3.1.1.	Прием файлов, первичный контроль	Контроль полноты и корректности информации, содержащейся в исходных файлах (функция может использоваться для различных форматов файлов).	900
3.1.2.	Прием информации из ЦОИ		5000
3.1.3.	Прием данных, содержащихся в DBF файлах	Вариант функции 3.1.1 для файлов формата DBF.	700
3.1.4.	Прием текстовых файлов	Вариант функции 3.1.1 для текстовых файлов.	550
3.2.	<i>Регистрация входных документов</i>		
3.2.1.	Регистрация входных документов		750
3.3.	<i>Работа с файлами и форматами</i>		
3.3.1.	Загрузка файлов	Чтение файла во внутренние структуры данных	1250
3.3.2.	Преобразование формата данных		2750
3.3.3.	Формирование файлов	Запись внутренних структур данных в файл	1750
3.4.	<i>Ведение архива и копирование информации</i>		
3.4.1.	Создание архива		1500
3.4.2.	Добавление данных в архив		750
3.4.3.	Извлечение данных из архива		750
3.4.4.	Поиск данных в архиве		1000
3.4.5.	Проверка целостности архива		1000
3.4.6.	Задачи ведения электронного архива	Состоит из всех задач ведения и копирования информации (3.4.1-3.4.5)	5000**
4.	<i>Операции управления данными</i>		
4.1.	<i>Ведение журналов</i>		
4.1.1.	Ведение журналов (на один журнал)		2750*
4.1.2.	Ведение журнала регистрации запросов и подтверждений по начальным (ответным) авизо		3500
4.1.3.	Вывод журналов приема-передачи		650
4.1.4.	Контроль и журнализация доступа к защищенным ресурсам		1750
4.2.	<i>Работа со справочниками</i>		
4.2.1.	Ведение справочников (на один справочник)		1750*
4.2.2.	Ведение книги регистрации открытых лицевых счетов		10400
4.3.	<i>Контроль информации документов</i>		
4.3.1.	Контроль входной информации документов	Контроль правильности документов при вводе	1250

4.3.2.	Контроль информации при вводе данных документа (<i>на форму ввода</i>)	Контроль правильности вводимой пользователем информации	350*
4.4.	<i>Настройка ПС</i>		
4.4.1.	Разработка и ввод метаданных	Организация работ с репозиторием системы, обеспечивающего ведение метаданных, т.е. всех необходимых описаний входной и выходной информации, алгоритмов преобразования; а также другой декларативной и процедурной информации об объектах системы.	4500
4.5.	<i>Ведение и обработка протоколов</i>		
4.5.1.	Протоколирование работы	Протоколирование работы включает в себя: -протоколирование выполняемых действий; -протоколирование возникающих ошибок.	3000
4.5.2.	Ведение протокола выполнения расчетов		600
4.5.3.	Формирование протокола проводок		1100
4.5.4.	Обработка протоколов		1500
4.6.	<i>Формирование отчетов</i>		
4.6.1.	Формирование отчетов (<i>на отчет</i>)	Представление результатов в табличном и графическом виде через систему аналитических отчетов.	2050*
4.6.2.	Просмотр и редактирование отчетных форм	Организация просмотра данных отчетных форм; возможность редактирования данных.	750
4.6.3.	Подготовка форм отчетности (<i>на одну форму</i>)		2000*
4.7.	<i>Распределенная обработка данных</i>		
4.7.1.	Создание одного объекта на базе технологии CORBA		50*
4.7.2.	Создание одного объекта на базе технологии COM		60*
4.7.3.	Вызов метода CORBA-объекта (<i>один вызов</i>)		2*
4.7.4.	Вызов метода COM-объекта (<i>один вызов</i>)		1*
4.7.5.	Реализация сетевого взаимодействия на базе средств Sockets API (серверная сторона, один сервер)		40*
4.7.6.	Реализация сетевого взаимодействия на базе средств Sockets API (клиентская сторона, один клиент)		20*

5.	Операции управления пользовательского интерфейса		
5.1.	Настройка ПС на условия применения		
5.1.1.	Настройка параметров	Настройка параметров процедур и интерфейсов комплекса задач.	750
5.1.2.	Настройка аналитических таблиц	Настройка аналитических и служебных таблиц.	1500
5.2.	Реализация пользовательского интерфейса¹		
5.2.1.	Реализация интерфейса пользователя	Организация диалога с пользователем для обеспечения работ по выполнению функций комплексов задач. Реализует одну из функций 5.2.2-5.2.8	2510
5.2.2.	Реализация стандартного графического пользовательского интерфейса (однооконное приложение)	Функция 5.2.1 для однооконного приложения	1150
5.2.3.	Реализация стандартного графического пользовательского интерфейса (диалоговое приложение)	Функция 5.2.1 для диалогового приложения	510
5.2.4.	Реализация стандартного графического пользовательского интерфейса (многооконное приложение)	Функция 5.2.1 для многооконного приложения	3750
5.2.5.	Реализация стандартного графического пользовательского интерфейса (однооконное приложение) с использованием:	Функция 5.2.2, если известны используемые средства (библиотеки, интерфейсы)	1500
5.2.6.	– API		750
5.2.7.	– MFC/OWL		500
	– VCL		
5.2.8.	Реализация стандартного графического пользовательского интерфейса (диалоговое приложение) с использованием:	Функция 5.2.3, если известны используемые средства (библиотеки, интерфейсы)	750
5.2.9.	– API		70
5.2.10.	– MFC/OWL		60
	– VCL		
5.2.11.	Реализация стандартного графического пользовательского интерфейса (многооконное приложение) с использованием:	Функция 5.2.4, если известны используемые средства (библиотеки, интерфейсы)	3500
5.2.12.	– API		1050
5.2.13.	– MFC/OWL		1000
	– VCL		

5.2.14.	Реализация стандартного графического пользовательского интерфейса (однооконное приложение) с использованием WEB-технологий	Функция 5.2.1 при использовании WEB-технологий	6150
5.3.	<i>Реализация машинной графики</i>		
5.3.1.	Реализация машинной графики для отображения состояния ПС в статике	Графическое представление статических моделей	1000
5.3.2.	Реализация машинной графики для отображения состояния ПС в динамике	Графическое представление динамических моделей	1750

* Функции выполняют единичную операцию и могут иметь несколько реализаций как на уровне ПС в целом, так и на уровне компонентов ПС. Каждая реализация должна характеризоваться коэффициентом степени повторного использования.

** Функции являются композитными (либо реализующими классы задач) и включают в себя другие функции из Каталога. Их использование допускается только для крупных ПС, а также для прочих ПС, в которых трудно детализировать набор функций. Такие функции не могут реализовываться более одного раза в расчете. Список функций, которые могут включаться в композитную функцию, приведен в таблице 2 данного приложения.

Все остальные функции являются структурными единицами ПС. Могут иметь несколько реализаций на уровне компонентов ПС и одну реализацию на уровне ПС. Каждая реализация должна характеризоваться коэффициентом степени повторного использования.

*** Допускается применение функций данной группы к базам данных, не являющихся аналитическими

Каталог содержит рекомендуемые значения объема функций программных средств Кредитных организаций, используемые в случае, когда не представляется возможным определить истинное значение объема функции.

¹ Под одним пользовательским интерфейсом следует понимать всю совокупность окон, элементов управления и прочих элементов, реализующих интерфейс одного ПС. Таким образом, одно ПС может реализовывать только одну функцию из группы не более одного раза.

Таблица 2. Состав композитных функций.

Композитная функция	Функциональный состав
1.1.5 Реализация базовых функций информационной безопасности в соответствии с нормативными документами Кредитных организаций [6000-15000]	<ul style="list-style-type: none"> – 1.1.2 Обеспечение безопасности передачи сообщений и обмена данными [200-500] – 1.1.3 Организация криптозащиты с помощью ПТК Криптоцентр авизо[5000-6000] – 1.1.6 Поддержка дискреционного контроля доступа [2000-5000] – 1.1.7 Поддержка мандатного контроля доступа [2000-5000] – 1.1.8 Создание, редактирование, удаление пользователей (группы пользователей) [1500-5000] – 1.1.11 Настройка политики безопасности[1000-2000] – 1.1.12 Ведение журналов отказа [1000-1500]
1.2.3 Реализация ПО обмена информацией между ЦОИ и клиентами (WEB-сервер) [9000-10000]	<ul style="list-style-type: none"> – 1.1.1 Реализация криптографических алгоритмов [100-1000] – 1.1.2 Обеспечение безопасности передачи сообщений и обмена данными [200-500] – 1.1.12 Ведение журналов отказа [1000-1500] – 1.2.1 Взаимодействие с другим ПО [500-1000] – 1.2.2 Взаимодействие с транспортной (почтовой) системой [2000-3000] – 1.4.1 Сетевая передача команд и сообщений [90] – 1.4.4 Удаленная доставка информации с подтверждением получения [200-500] – 1.4.6 Управление файлами, доступом к файлам и передачей файлов между удаленными и разнородными файловыми системами [200-500] – 1.4.9 Реализация средства контроля состояния распределенной сети однородных компонент [100-200]

<p>1.2.4 Организация доступа к динамическим информационным ресурсам через Intranet Кредитных организаций [10000-15000]</p>	<ul style="list-style-type: none"> – 1.1.1 Реализация криптографических алгоритмов [100-1000] – 1.1.2 Обеспечение безопасности передачи сообщений и обмена данными [200-500] – 1.1.12 Ведение журналов отказа [1000-1500] – 1.2.1 Взаимодействие с другим ПО [500-1000] – 1.2.2 Взаимодействие с транспортной (почтовой) системой [2000-3000] – 1.4.1 Сетевая передача команд и сообщений [90] – 1.4.4 Удаленная доставка информации с подтверждением получения [200-500] – 1.4.6 Управление файлами, доступом к файлам и передачей файлов между удаленными и разнородными файловыми системами [200-500] – 1.4.9 Реализация средства контроля состояния распределенной сети однородных компонент [100-200]
<p>1.2.5 Организация регулярно актуализируемого статистического контента для доступа через Intranet Кредитных организаций [10000-15000]</p>	<ul style="list-style-type: none"> – 1.1.1 Реализация криптографических алгоритмов [100-1000] – 1.1.2 Обеспечение безопасности передачи сообщений и обмена данными [200-500] – 1.1.12 Ведение журналов отказа [1000-1500] – 1.2.1 Взаимодействие с другим ПО [500-1000] – 1.2.2 Взаимодействие с транспортной (почтовой) системой [2000-3000] – 1.4.1 Сетевая передача команд и сообщений [90] – 1.4.4 Удаленная доставка информации с подтверждением получения [200-500] – 1.4.6 Управление файлами, доступом к файлам и передачей файлов между удаленными и разнородными файловыми системами [200-500] – 1.4.9 Реализация средства контроля состояния распределенной сети однородных компонент [100-200]

1.5.11 Прикладное администрирование аналитических БД [1000-5000]	<ul style="list-style-type: none"> – 1.5.5 Ведение журнала операций с аналитическими БД [1000-2000] – 1.5.8 Функции настройки аналитических БД [1500-2000] – 1.5.10 Контроль целостности и восстановление аналитических БД [500-1000]
2.2.1 Составление балансов [500-1000]	<ul style="list-style-type: none"> – 2.1.5 Расчет алгебраических выражений [100-1000] – 4.6.1 Формирование отчетов [100-4000]
2.2.2 Анализ балансов [1000-3500]	<ul style="list-style-type: none"> – 2.1.5 Расчет алгебраических выражений [100-1000] – 2.1.6 Статистическая обработка данных [2000-5000] – 4.6.1 Формирование отчетов [100-4000]
2.2.3 Выполнение баланса и оборотов КО за месяц [2500-3500]	<ul style="list-style-type: none"> – 2.1.5 Расчет алгебраических выражений [100-1000] – 4.6.1 Формирование отчетов [100-4000]
2.2.4 Выполнение ежедневного баланса [1500-2000]	<ul style="list-style-type: none"> – 2.1.5 Расчет алгебраических выражений [100-1000] – 4.6.1 Формирование отчетов [100-4000]
2.2.5 Корректировка балансов [200-500]	<ul style="list-style-type: none"> – 2.1.5 Расчет алгебраических выражений [100-1000] – 4.6.2 Просмотр и редактирование отчетных форм [500-1000]
2.2.6 Пересчет балансов [2000-3000]	<ul style="list-style-type: none"> – 2.1.5 Расчет алгебраических выражений [100-1000] – 4.6.2 Просмотр и редактирование отчетных форм [500-1000]
2.3.1 Анализ безналичных расчетов в платежных системах [2500-3500]	<ul style="list-style-type: none"> – 1.5.4 Формирование запросов к аналитическим БД [10-80] – 1.5.12 Поиск, предоставление и вывод информации [1000-6000] – 2.1.5 Расчет алгебраических выражений [100-1000] – 2.1.6 Статистическая обработка данных [2000-5000] – 4.6.1 Формирование отчетов [100-4000]
2.3.2 Анализ денежной массы в обращении по районам [5000-6000]	<ul style="list-style-type: none"> – 1.5.4 Формирование запросов к аналитическим БД [10-80] – 1.5.12 Поиск, предоставление и вывод информации [1000-6000] – 2.1.5 Расчет алгебраических выражений [100-1000] – 2.1.6 Статистическая обработка данных [2000-5000] – 4.6.1 Формирование отчетов [100-4000]

2.3.3 Анализ курсов валют, устанавливаемых ЦБ [3500-4500]	<ul style="list-style-type: none"> – 1.5.4 Формирование запросов к аналитическим БД [10-80] – 1.5.12 Поиск, предоставление и вывод информации [1000-6000] – 2.1.5 Расчет алгебраических выражений [100-1000] – 2.1.6 Статистическая обработка данных [2000-5000] – 4.6.1 Формирование отчетов [100-4000]
2.3.4 Анализ курсов основной валюты в обменных пунктах КО [7000-8000]	<ul style="list-style-type: none"> – 1.5.4 Формирование запросов к аналитическим БД [10-80] – 1.5.12 Поиск, предоставление и вывод информации [1000-6000] – 2.1.5 Расчет алгебраических выражений [100-1000] – 2.1.6 Статистическая обработка данных [2000-5000] – 4.6.1 Формирование отчетов [100-4000]
2.3.5 Анализ событий нефинансового характера [300-1100]	<ul style="list-style-type: none"> – 2.1.5 Расчет алгебраических выражений [100-1000] – 2.1.6 Статистическая обработка данных [2000-5000] – 4.6.1 Формирование отчетов [100-4000]
2.4.1 Планирование инспекторских проверок [1500-2000]	<ul style="list-style-type: none"> – 2.1.5 Расчет алгебраических выражений [100-1000] – 2.1.6 Статистическая обработка данных [2000-5000] – 4.6.1 Формирование отчетов [100-4000]
2.4.2 Планирование эмиссии [4000-5000]	<ul style="list-style-type: none"> – 2.1.5 Расчет алгебраических выражений [100-1000] – 2.1.6 Статистическая обработка данных [2000-5000] – 4.6.1 Формирование отчетов [100-4000]
2.4.3 Планирование прочих задач [800-4000]	<ul style="list-style-type: none"> – 2.1.5 Расчет алгебраических выражений [100-1000] – 2.1.6 Статистическая обработка данных [2000-5000] – 4.6.1 Формирование отчетов [100-4000]
2.5.1 Мониторинг отчетности КО [6000-7000]	<ul style="list-style-type: none"> – 1.5.12 Поиск, предоставление и вывод информации [1000-6000] – 1.5.13 Обработка записей базы данных [500-3000] – 4.6.3 Подготовка форм отчетности [1000-3000] – 5.3.1 Реализация машинной графики для отображения состояния ПС в статике [1000]

2.5.2 Мониторинг отчетности ТУ [6000-7000]	<ul style="list-style-type: none"> – 1.5.12 Поиск, предоставление и вывод информации [1000-6000] – 1.5.13 Обработка записей базы данных [500-3000] – 4.6.3 Подготовка форм отчетности [1000-3000] – 5.3.1 Реализация машинной графики для отображения состояния ПС в статике [1000]
2.6.1 Расчет экономических показателей [5000 – 15000]	<ul style="list-style-type: none"> – 2.1.1 Ввод данных первичных документов в интерактивном режиме [2000-9000] – 2.1.4 Логический, синтаксический и номенклатурный контроль данных [1500-6000] – 2.1.5 Расчет алгебраических выражений [100-1000] – 2.1.6 Статистическая обработка данных [2000-5000] – 4.6.1 Формирование отчетов [100-4000]
2.6.2 Экономический и финансовый анализ [10000 – 20000]	<ul style="list-style-type: none"> – 2.1.1 Ввод данных первичных документов в интерактивном режиме [2000-9000] – 2.1.4 Логический, синтаксический и номенклатурный контроль данных [1500-6000] – 2.1.5 Расчет алгебраических выражений [100-1000] – 2.1.6 Статистическая обработка данных [2000-5000] – 4.6.1 Формирование отчетов [100-4000]
2.6.3 Моделирование и прогнозирование [20000-30000]	<ul style="list-style-type: none"> – 2.1.1 Ввод данных первичных документов в интерактивном режиме [2000-9000] – 2.1.4 Логический, синтаксический и номенклатурный контроль данных [1500-6000] – 2.1.5 Расчет алгебраических выражений [100-1000] – 2.1.6 Статистическая обработка данных [2000-5000] – 4.6.1 Формирование отчетов [100-4000]

3.4.1 Задачи ведения электронного архива [2500-7500]	<ul style="list-style-type: none"> – 3.4.1 Создание архива [500-2500] – 3.4.2 Добавление данных в архив [500-1000] – 3.4.3 Извлечение данных из архива [500-1000] – 3.4.4 Поиск данных в архиве [500-1500] – 3.4.5 Проверка целостности архива [500-1500]
4.9.4 Ведение картотек расчетных документов	<ul style="list-style-type: none"> – 4.9.1 Ведение картотеки расчетных документов, ожидающих акцепта для оплаты – 4.9.2 Ведение картотеки расчетных документов, не оплаченных в срок – 4.9.3 Прием ЭСИС – запросов УБР для получения информации об актуальном состоянии картотек

ПРИЛОЖЕНИЕ 2**ПРИМЕР ТЕХНИЧЕСКОГО ЗАДАНИЯ НА РАЗРАБОТКУ ПС****1. НАЗНАЧЕНИЕ И ОБЛАСТЬ ПРИМЕНЕНИЯ****1.1. Назначение работы**

Программное средство мониторинга финансовых операций, выполняемых через банкомат, предназначено для автоматизации:

- учета выданных карт клиентам банка;
- учета операций по снятию наличных в банкоматах Банка;
- учета операций по оплате услуг оператора сотовой связи через банкоматы;
- составления отчета о выполненных операциях;
- анализа выполненных операций для прогнозирования необходимого количества наличных денег для банкоматов и сроков их пополнения.

Программное средство должно обеспечить возможность:

- автоматизированной обработки информации о проведенных операциях и состоянии личного счета клиентов Банка, использующих возможность выполнения финансовых операций по снятию наличных и оплате услуг оператора сотовой связи через банкоматы Банка;
- составления отчета по выполненным финансовым операциям и состоянию счета клиентов;
- анализа информации, полученной от банкоматов и планирования их обслуживания.

1.2. Область применения

Программное средство, разрабатываемое в соответствии с данным техническим заданием, предназначено для применения в подразделениях Кредитных организаций, осуществляющих контроль финансовых операций. ПС создается с целью автоматизации учета финансовых операций, выполняемых клиентами Банка с использованием кредитных карт.

2. ОСНОВНЫЕ ТЕХНИЧЕСКИЕ ТРЕБОВАНИЯ**2.1. Требования к структуре ПС**

Необходимая для функционирования программного средства информация поступает на вход ПС из следующих источников:

- банкомат (передает информацию о выполнении финансовых операций);

- оператор ПС (выполняет ввод, получение и удаление информации о клиентах Банка, которым были выданы кредитные карты);

- администратор ПС (выполняет ввод, получение и удаление информации о новых банкоматах и операторах сотовой связи, вводе данных о пополнении денежных средств банкоматов).

В результате работы программное средство формирует:

- отчет о выполненных операциях;
- результаты анализа наличия денежных средств в банкоматах и примерные сроки их пополнения.

Каждая функция программного средства предполагает возврат сообщения об ошибках.

Программное средство должно состоять из следующих компонентов:

- «АРМ Оператора», предназначенного для работы оператора с программным средством;

- «АРМ Администратора», предназначенного для работы администратора с программным средством;

- «База Данных», предназначенного для хранения информации о выданных кредитных картах, операций по снятию наличных, оплаты услуг сотовой связи;

- «Отчет», предназначенного для составления отчета о выполняемых операциях.

- «Анализ», предназначенного для прогнозирования сроков пополнения и количества необходимых наличных денег для банкоматов.

Основными источниками данных для программного средства являются:

- база данных Банка, содержащая информацию о выданных клиентам кредитных картах и проводимых по ним операциям;

- данные, получаемые от банкоматов Банка.

2.2. Требования к функциональным возможностям ПС

Программное средство должно обеспечивать выполнение следующих функций:

- добавление оператором ПС информации о клиенте Банка в базу данных при получении клиентом кредитной карты;

- получение/удаление оператором ПС информации о клиенте Банка из базы данных;

- добавление администратором ПС информации о новом банкомате;

- получение/удаление администратором ПС информации о банкомате;

- добавление администратором ПС информации об операторе сотовой связи;

- получение/удаление администратором ПС информации об операторе сотовой связи;

- выполнении администратором ПС операции ввода информации о пополнении денежных средств банкоматов;
- выполнение оператором ПС или клиентом Банка операции получения состояния счета Клиента Банка;
- выполнение оператором ПС или клиентом Банка операции снятия наличных средств со счета клиента Банка;
- выполнение оператором ПС или клиентом Банка операции перевода денежных средств клиента со счета в Банке при оплате услуг;
- выполнение оператором ПС операции перевода денежных средств на счет клиента Банка;
- формирование оператором ПС отчета о выполненных операциях;
- получение администратором ПС анализа о выполненных операциях для прогнозирования необходимого количества наличных денег для банкоматов и сроков их пополнения.

2.3. Технические и системные требования к ПС

Программное средство мониторинга финансовых операций, выполняемых через банкомат, должно функционировать на ТС со следующими минимальными требованиями по аппаратному и программному обеспечению:

- CPU P3, RAM 128 MB;
- 50 MB дискового пространства;
- ОС Windows XP;
- СУБД SQL Server 6.0.

2.4. Требования к надежности

Программное средство мониторинга финансовых операций, выполняемых через банкомат, должно обладать надежностью, обеспечивающей работу ПС в рабочем режиме и оперативное восстановление работоспособности при сбоях. ПС должно иметь необходимый высокий уровень защиты информации в БД и поступающей в программное средство от банкоматов Банка.

В ПС должны быть предусмотрены:

- контроль за целостностью данных на уровне СУБД;
- сохранение целостности данных в БД при нештатном завершении программы;
- сохранение работоспособности ПС при некорректных действиях администратора или оператора ПС.

2.5. Требования к эргономике

Программное средство должно использовать средства построения графического интерфейса пользователя, обеспечиваемых операционной системой Windows.

Сообщения, выдаваемые программным средством, должны выдаваться на русском языке за исключением стандартных системных сообщений.

3. ТРЕБОВАНИЯ К ДОКУМЕНТИРОВАНИЮ

3.1. Состав и содержание документации должны соответствовать требованиям ГОСТ 19.101–77.

3.2. Перечень документов должен включать:

- 1) Техническое задание на разработку ПС.
- 2) Пояснительная записка;
- 3) Описание информационного обеспечения программы;
- 4) Спецификация;
- 5) Формуляр;
- 6) Программа и методика предварительных испытаний;
- 7) Программа и методика приемочных испытаний;
- 8) Руководство пользователя.

ПРИЛОЖЕНИЕ 3**РАСЧЕТ ТРУДОЕМКОСТИ РАЗРАБОТКИ МАКЕТА ПС****1. Функциональный состав ПС**

Определяем функциональный состав макета программного средства исходя из технического задания на разработку ПС. При этом необходимо отметить:

– Компоненты «АРМ оператора» и «АРМ администратора» реализуют минимально необходимый интерфейс. Компонент «АРМ администратора» не реализует функций по получению и удалению информации о банкоматах и операторах сотовой связи, а реализует только ввод этих данных. Для макета также не обязательна полноценная реализация функций обеспечения безопасности информации, хотя в минимальном варианте данная возможность должна присутствовать. Таким образом, программное средство реализует следующие функции:

1) Компонент «АРМ оператора»:

– добавление, получение и удаление информации о клиенте Банка в базе данных;
– выполнение запросов к базе данных для получения состояния счета, снятия наличных средств со счета, перевода денежных средств клиента со счета в Банке при оплате услуг или перевода денежных средств на счет клиента Банка;

2) Компонент «АРМ администратора»:

– добавление информации о банкомате, операторе сотовой связи;
– выполнение запросов к базе данных для ввода информации о пополнении денежных средств банкоматов.

3) Компонент «База данных»

– формирование структуры базы данных;
– обеспечение безопасности передачи данных при работе с БД.

4) Компонент «Анализ»:

– проведение анализа о выполненных операциях для прогнозирования необходимого количества наличных денег для банкоматов и сроков их пополнения;
– формирование отчета по анализу.

3) Компонент «Отчет»:

– формирование отчета о выполненных операциях.

2. Расчет трудоемкости разработки ПС

Определяем трудоемкость разработки и сроки разработки макета программного средства. Ниже представлен отчет по расчету трудоемкости.

РАСЧЕТ ТРУДОЕМКОСТИ С ИСПОЛЬЗОВАНИЕМ ПО АРМ

Расчет трудоемкости разработки ПС (Каталог)

Параметры расчета

Параметр	Значение
Имя расчета	Программное средство мониторинга финансовых операций, выполняемых через банкомат (макет)
Тип расчета	Расчет трудоемкости разработки ПС
Версия модели	2.5.1.0
Фонд времени разработчика [дней/мес.]	21
Тип ограничения	По продолжительности этапа [мес.]
Тип структуры программного средства	ПС с набором библиотек (=1.00)
Технологии взаимодействия модулей	Оригинальный протокол взаимодействия (sockets) (=0.80)

Коэффициенты уровня расчета

Коэффициент	Уровень значения
Степень новизны ПС	Принципиально новое ПС, не имеющее доступных аналогов на прежнем типе ТС/ОС (=1.06)
Требования к надежности	Сбои ПС приводят к потерям рабочей информации (=1.06)
Требования к производительности	Требования к производительности ПС не установлены (однако производительность ПС должна обеспечивать приемлемое время отклика при работе пользователя в интерактивном режиме) (=1.00)
Информативность документации	Не учтены некоторые потребности жизненного цикла (=0.94)
Опыт разработки ПС подобного типа	Более 5 разработанных ПС (компонентов АС) (=0.81)
Влияние сроков работ на трудоемкость	Равно или более 100 (=1.00)

Коэффициенты уровня подпроцесса

Подпроцесс	Коэффициент	Уровень значения
Анализ требований	Уровень квалификации аналитиков	Высокий (=0.87)
Анализ требований	Опыт работы аналитиков в данной предметной области	5 разработанных ПС (компонентов АС) (=0.87)
Проектирование	Уровень квалификации проектировщиков	Высокий (=0.92)

Проектирование	Опыт работы проектировщиков с используемыми средствами проектирования	5 разработанных ПС (компонентов АС) (=0.92)
Проектирование	Используемые средства проектирования	Средства проектирования БД (BPWIN, ERWIN/ERX, Designer 2000, DataDirect Explorer, Oracle Designer, Silverrun RBM и BPM, Together J) (=1.03)
Программирование	Уровень квалификации программистов	Высокий (=0.95)
Программирование	Используемая среда разработки	Интегрированные среды разработки (Visual Studio) (=1.00)
Тестирование	Уровень квалификации персонала, осуществляющего тестирование	Высокий (=0.89)
Тестирование	Используемые средства тестирования	Автоматизированные средства тестирования использовались (=0.76)
Тестирование	Размер БД	D/P (=0.95)
Ввод в действие ПС	Уровень квалификации персонала, осуществляющего ввод в действие ПС	Высокий (=0.94)

Дерево элементов

№	Элемент	Описание	Vi (на условном языке)	Кср.разр.	ri	ki [0;1]	Ксложн.	Копыт
	АРМ Оператора	Компонент предназначен для работы оператора с программным средством						
5.2.2	+Реализация стандартного графического пользовательского интерфейса (однооконное приложение)	Реализует интерфейс оператора ПС, обеспечивающий работу оператора с программным средством.	300 из [300-2000]	C++ (=1.00)	0.90	1	Высокий (=1.10)	Высокий (=0.91)
2.1.15	+Ввод прочих данных	Добавление информации о клиенте Банка в БД программного средства.	2500 из [1000-4000]	C++ (=1.00)	0.85	1	Высокий (=1.10)	Высокий (=0.91)
1.5.12	+Поиск, представление и вывод информации	Получение и удаление информации о клиенте Банка в базе данных, выполнение запросов к базе данных для получения состояния счета, снятия наличных средств со счета, перевода денежных средств клиента со счета в Банке при оплате услуг или перевода денежных сред	3500 из [1000-6000]	C++ (=1.00)	1.00	1	Высокий (=1.10)	Высокий (=0.91)
	АРМ Администратора	Компонент предназначен для работы администратора с программным средством						
№	Элемент	Описание	Vi (на условном языке)	Кср.разр.	ri	ki [0;1]	Ксложн.	Копыт
5.2.2	+Реализация стандартного графического пользовательского интерфейса (однооконное приложение)	Реализует интерфейс администратора ПС, обеспечивающий работу администратора с программным средством.	300 из [300-2000]	C++ (=1.00)	0.90	1	Высокий (=1.10)	Высокий (=0.91)
2.1.15	+Ввод прочих данных	Добавление информации о банкомате, об операторе сотовой связи, выполнение запросов	2500 из [1000-4000]	C++ (=1.00)	0.85	1	Высокий (=1.10)	Высокий (=0.91)

		к базе данных для ввода информации о пополнении денежных средств банкоматов.						
	База Данных	Компонент предназначен для хранения информации о выданных кредитных картах, операций по снятию наличных, оплаты услуг сотовой связи.						
1.5.1	+Формирование физической структуры аналитических БД (на одну БД)	Обеспечивает формирование структуры базы данных (для обеспечения ее нормального функционирования с разрабатываемым программным средством).	4500 из [1000-8000]	C++ (=1.00)	1.00	1	Высокий (=1.10)	Высокий (=0.91)
1.1.2	+Обеспечение безопасности передачи сообщений и обмена данными (шифрование)	Обеспечивает безопасность передачи информации из базы данных (согласно требованиям технического задания на разработку ПС).	350 из [200-1500]	C++ (=1.00)	1.00	1	Высокий (=1.10)	Высокий (=0.91)
	Отчет	Компонент предназначен для составления отчета о выполняемых операциях (снятие наличных со счета, оплата услуг сотовой связи, получение информации о состоянии счета).						
4.6.1	+Формирование отчетов (на один отчет)*	Формирование отчета о выполненных операциях (снятие наличных со счета, оплата услуг сотовой связи, получение информации о состоянии счета).	2050 из [100-4000]	C++ (=1.00)	1.00	1	Высокий (=1.10)	Высокий (=0.91)
	Анализ	Компонент предназначен для прогнозирования сроков пополнения и количества необходимых наличных денег для банкоматов.						
2.1.5	+Расчет алгебраических выражений	Проведение анализа о выполненных операциях (снятие наличных со счета, оплата услуг сотовой связи, получение информации о состоянии счета) для прогнозирования сроков пополнения и количества необходимых наличных денег для банкоматов.	550 из [100-1000]	C++ (=1.00)	1.00	1	Высокий (=1.10)	Высокий (=0.91)

Расчет трудоемкости подпроцессов

Подпроцесс	Трудоемкость [чел.дн.]	Кол-во разработчиков [чел]	Продолжительность [мес]/[дн]
Анализ требований	3.10	0.07	2.00/42.00
Проектирование	42.79	0.68	3.00/63.00
Программирование	306.94	3.65	4.00/84.00
Тестирование	18.40	0.44	2.00/42.00
Ввод в действие ПС	3.84	0.18	1.00/21.00
Всего	375.06		12.00/252.00
Базовая трудоемкость	408.98		

ПРИЛОЖЕНИЕ 4**ПРИМЕР РАСЧЕТА ТРУДОЕМКОСТИ РАЗРАБОТКИ ПС С ИСПОЛЬЗОВАНИЕМ
КАТАЛОГА ФУНКЦИЙ****1. Функциональный состав ПС**

Определяем функциональный состав программного средства исходя из технического задания на разработку ПС:

1) Компонент «АРМ оператора»:

- добавление, получение и удаление информации о клиенте Банка в базе данных;
- выполнение запросов к базе данных для получения состояния счета, снятия наличных средств со счета, перевода денежных средств клиента со счета в Банке при оплате услуг или перевода денежных средств на счет клиента Банка;

2) Компонент «АРМ администратора»:

- добавление, получение и удаление информации о банкомате;
- добавление, получение и удаление информации об операторе сотовой связи;
- выполнение запросов к базе данных для ввода информации о пополнении денежных средств банкоматов.

3) Компонент «База данных»

- формирование структуры базы данных;
- обеспечение безопасности передачи данных при работе с БД.

4) Компонент «Анализ»:

- проведение анализа о выполненных операциях для прогнозирования необходимого количества наличных денег для банкоматов и сроков их пополнения;
- формирование отчета по анализу.

3) Компонент «Отчет»:

- формирование отчета о выполненных операциях.

2. Расчет трудоемкости разработки ПС

Определяем трудоемкость разработки и сроки разработки программного средства. Ниже представлен отчет по расчету трудоемкости.

РАСЧЕТ ТРУДОЕМКОСТИ С ИСПОЛЬЗОВАНИЕМ ПО АРМ

Расчет трудоемкости разработки ПС (Каталог)

Параметры расчета

Параметр	Значение
Имя расчета	Программное средство мониторинга финансовых операций, выполняемых через банкомат
Тип расчета	Расчет трудоемкости разработки ПС (Каталог)
Версия модели	2.5.1.0
Фонд времени разработчика [дней/мес.]	21
Тип ограничения	По продолжительности этапа [мес.]
Тип структуры программного средства	ПС с набором библиотек (=1.00)
Технологии взаимодействия модулей	Оригинальный протокол взаимодействия (sockets) (=0.80)

Коэффициенты уровня расчета

Коэффициент	Уровень значения
Степень новизны ПС	Принципиально новое ПС, не имеющее доступных аналогов на прежнем типе ТС/ОС (=1.06)
Требования к надежности	Сбои ПС приводят к потерям рабочей информации (=1.06)
Требования к производительности	Повышенные требования к производительности (=1.12)
Информативность документации	Повышенный объем для жизненного цикла данного ПС (=1.07)
Опыт разработки ПС подобного типа	Более 5 разработанных ПС (компонентов АС) (=0.81)
Влияние сроков работ на трудоемкость	Равно или более 100 (=1.00)

Коэффициенты уровня подпроцесса

Подпроцесс	Коэффициент	Уровень значения
Анализ требований	Уровень квалификации аналитиков	Высокий (=0.87)
Анализ требований	Опыт работы аналитиков в данной предметной области	5 разработанных ПС (компонентов АС) (=0.87)
Проектирование	Уровень квалификации проектировщиков	Высокий (=0.92)
Проектирование	Опыт работы проектировщиков с используемыми средствами	5 разработанных ПС (компонентов АС) (=0.92)

	проектирования	
Проектирование	Используемые средства проектирования	Средства проектирования БД (BPWIN, ERWIN/ERX, Designer 2000, DataDirect Explorer, Oracle Designer, Silverrun RBM и BPM, Together J) (=1.03)
Программирование	Уровень квалификации программистов	Высокий (=0.95)
Программирование	Используемая среда разработки	Интегрированные среды разработки (Visual Studio) (=1.00)
Тестирование	Уровень квалификации персонала, осуществляющего тестирование	Высокий (=0.89)
Тестирование	Используемые средства тестирования	Автоматизированные средства тестирования использовались (=0.76)
Тестирование	Размер БД	$10 \leq D/P < 100$ (=1.00)
Ввод в действие ПС	Уровень квалификации персонала, осуществляющего ввод в действие ПС	Высокий (=0.94)

Дерево элементов

№	Элемент	Описание	Vi (на условном языке)	Кср.разр.	gi	ki [0;1]	Ксложн.	Копыт
	АРМ Оператора	Компонент предназначен для работы оператора с программным средством						
2.2.5	+Корректировка балансов**	Реализует интерфейс оператора ПС, обеспечивающий работу оператора с программным средством.	500 из [200-500]	C++ (=1.00)	1	0.90	Высокий (=1.10)	Высокий (=0.91)
2.1.15	+Ввод прочих данных	Добавление информации о клиенте Банка в БД программного средства.	3000 из [1000-4000]	C++ (=1.00)	1	0.85	Высокий (=1.10)	Высокий (=0.91)
1.5.12	+Поиск, представление и вывод информации	Получение и удаление информации о клиенте Банка в базе данных, выполнение запросов к базе данных для получения состояния счета, снятия наличных средств со счета, перевода денежных средств клиента со счета в Банке при оплате услуг или перевода денежных сред	3500 из [1000-6000]	C++ (=1.00)	1	0.90	Высокий (=1.10)	Высокий (=0.91)
	АРМ Администратора	Компонент предназначен для работы администратора с программным средством						
2.2.5	+Корректировка балансов**	Реализует интерфейс администратора ПС, обеспечивающий работу администратора с программным средством.	500 из [200-500]	C++ (=1.00)	1	0.90	Высокий (=1.10)	Высокий (=0.91)
2.1.15	+Ввод прочих данных	Добавление информации о банкомате, об операторе сотовой связи, выполнение запросов к базе данных для ввода информации о пополнении денежных средств банкоматов.	3000 из [1000-4000]	C++ (=1.00)	1	0.85	Высокий (=1.10)	Высокий (=0.91)
1.5.12	+Поиск, представление и вывод информации	Получение и удаление информации о банкомате и операторе сотовой связи.	3500 из [1000-6000]	C++ (=1.00)	1	0.90	Высокий (=1.10)	Высокий (=0.91)
	База Данных	Компонент предназначен для хранения информации о выданных кредитных картах, операций по снятию наличных, оплаты услуг сотовой связи.						

1.5.1	+Формирование физической структуры аналитических БД (на одну БД)	Обеспечивает формирование структуры базы данных (для обеспечения ее нормального функционирования с разрабатываемым программным средством).	4500 из [1000-8000]	C++ (=1.00)	1	1.00	Высокий (=1.10)	Высокий (=0.91)
2.1.1	+Ввод данных первичных документов в интерактивном режиме	Обеспечивает безопасность передачи информации из базы данных (согласно требованиям технического задания на разработку ПС).	2000 из [2000-9000]	C++ (=1.00))	1	1.00	Высокий (=1.10)	Высокий (=0.91)
	Отчет	Компонент предназначен для составления отчета о выполняемых операциях (снятие наличных со счета, оплата услуг сотовой связи, получение информации о состоянии счета).						
4.6.1	+Формирование отчетов (на один отчет)*	Формирование отчета о выполненных операциях (снятие наличных со счета, оплата услуг сотовой связи, получение информации о состоянии счета).	2050 из [100-4000]	C++ (=1.00)	1	1.00	Высокий (=1.10)	Высокий (=0.91)
	Анализ	Компонент предназначен для прогнозирования сроков пополнения и количества необходимых наличных денег для банкоматов.						
5.1.2	+Настройка аналитических таблиц	Проведение анализа о выполненных операциях (снятие наличных со счета, оплата услуг сотовой связи, получение информации о состоянии счета) для прогнозирования сроков пополнения и количества необходимых наличных денег для банкоматов.	1000 из [1000-2000]	C++ (=1.00)	1	1.00	Высокий (=1.10)	Высокий (=0.91)
4.6.1	+Формирование отчетов (на один отчет)*	Формирование отчета по анализу выполненных операций (снятие наличных со счета, оплата услуг сотовой связи, получение информации о состоянии счета) через банкоматы Банка.	2050 из [100-4000]	C++ (=1.00)	1	1.00	Высокий (=1.10)	Высокий (=0.91)
	Общий объем ПС:		22010					

Расчет трудоемкости подпроцессов

Подпроцесс	Трудоемкость [чел.дн.]	Кол-во разработчиков [чел]	Продолжительность [мес]/[дн]
Анализ требований	4.89	0.12	2.00/42.00
Проектирование	67.55	1.07	3.00/63.00
Программирование	484.60	5.77	4.00/84.00
Тестирование	30.57	0.73	2.00/42.00
Ввод в действие ПС	6.07	0.29	1.00/21.00
Всего	593.69		12.00/252.00
Базовая трудоемкость	645.71		

ПРИЛОЖЕНИЕ 5**ПРИМЕР РАСЧЕТА ТРУДОЕМКОСТИ РАЗРАБОТКИ ПС С ИСПОЛЬЗОВАНИЕМ
МЕТОДА ФУНКЦИОНАЛЬНОЙ БАЛЬНОЙ ОЦЕНКИ****1. Функциональный состав ПС**

Программное средство мониторинга финансовых операций, выполняемых через банкомат, решает задачи:

- формирования и ведения базы данных о финансовых операциях, выполняемых клиентами Банка через банкоматы и оператором ПС (запросы о состоянии счета, снятие наличных и оплата услуг оператора сотовой связи);
- учета кредитных карт, выданных клиентам Банка;
- составления отчета о выполненных операциях;
- анализа текущего состояния банкоматов Банка с целью определения сроков обслуживания банкоматов (пополнение наличных денежных средств банкоматов) и количестве необходимых для банкомата наличных денежных средств.

К основным функциям рассматриваемого программного средства относятся следующие:

- добавление информации оператором ПС о клиенте Банка в базу данных при получении клиентом кредитной карты;
- получение/удаление оператором ПС информации о клиенте Банка из базы данных;
- внесение оператором ПС информации о зачислении денежных средств на счет клиента;
- добавление администратором ПС информации о новом банкомате;
- получение/удаление администратором ПС информации о банкомате;
- добавление администратором ПС информации об операторе сотовой связи;
- получение/удаление администратором ПС информации об операторе сотовой связи;
- выполнении администратором ПС операции ввода информации о пополнении денежных средств банкоматов;
- выполнении оператором ПС или клиентом Банка операции получения состояния счета клиента Банка;
- выполнении оператором ПС или клиентом Банка операции снятия наличных средств со счета клиента Банка;

- выполнение оператором ПС или клиентом Банка операции перевода денежных средств клиента со счета в Банке при оплате услуг;
- формирование оператором ПС отчета о выполненных операциях;
- получение администратором ПС анализа о выполненных операциях для прогнозирования необходимого количества наличных денег для банкоматов и сроков их пополнения.

2. Описание базы данных

База данных представляет собой реляционную модель с простыми (атомарными) типами данных, приведенную к 2 нормальной форме (Рисунок 2.1 Структура базы данных). БД состоит из 8 таблиц:

- 1) Клиент – содержит информацию о клиенте банка (фамилия, имя и отчество, адрес, паспортные данные) и договоре на обслуживание кредитной карты, заключенном с ним (номер и дата заключения договора, номер выданной клиенту Банка карты).
- 2) Счет – содержит информацию о состоянии счета клиента банка (сумму денежных средств на карте);
- 3) Банкомат – содержит информацию о банкоматах Банка (номер банкомата, его серийный номер и место расположения), посредством использования которых клиент Банка способен осуществить операции получения баланса персонального счета, снятия наличных со счета и оплаты услуг оператора сотовой связи.
- 4) ДенежныйОстаток – содержит информацию о времени и сумме пополнения денежных средств банкоматов Банка;
- 5) ОператорСотовойСвязи – содержит информацию об операторах сотовой связи (номер счета и название оператора, а также его юридический адрес), услуги которых клиент может оплатить с использованием банкоматов Банка.
- 6) Операция – содержит информацию об операциях (номер операции и банкомата, код типа операции, номер карты, использованной для совершения операции, дату выполнения и результат операции, а также сумму, на которую была совершена операция), выполненных клиентами с использованием банкоматов Банка.
- 7) ОперацияОплатыУслуг – содержит дополнительную информацию (счет и номер абонента) об операциях оплаты услуг оператора сотовой связи.
- 8) Тип операции – содержит информацию о типах операций, которые могут совершаться через банкоматы Банка.

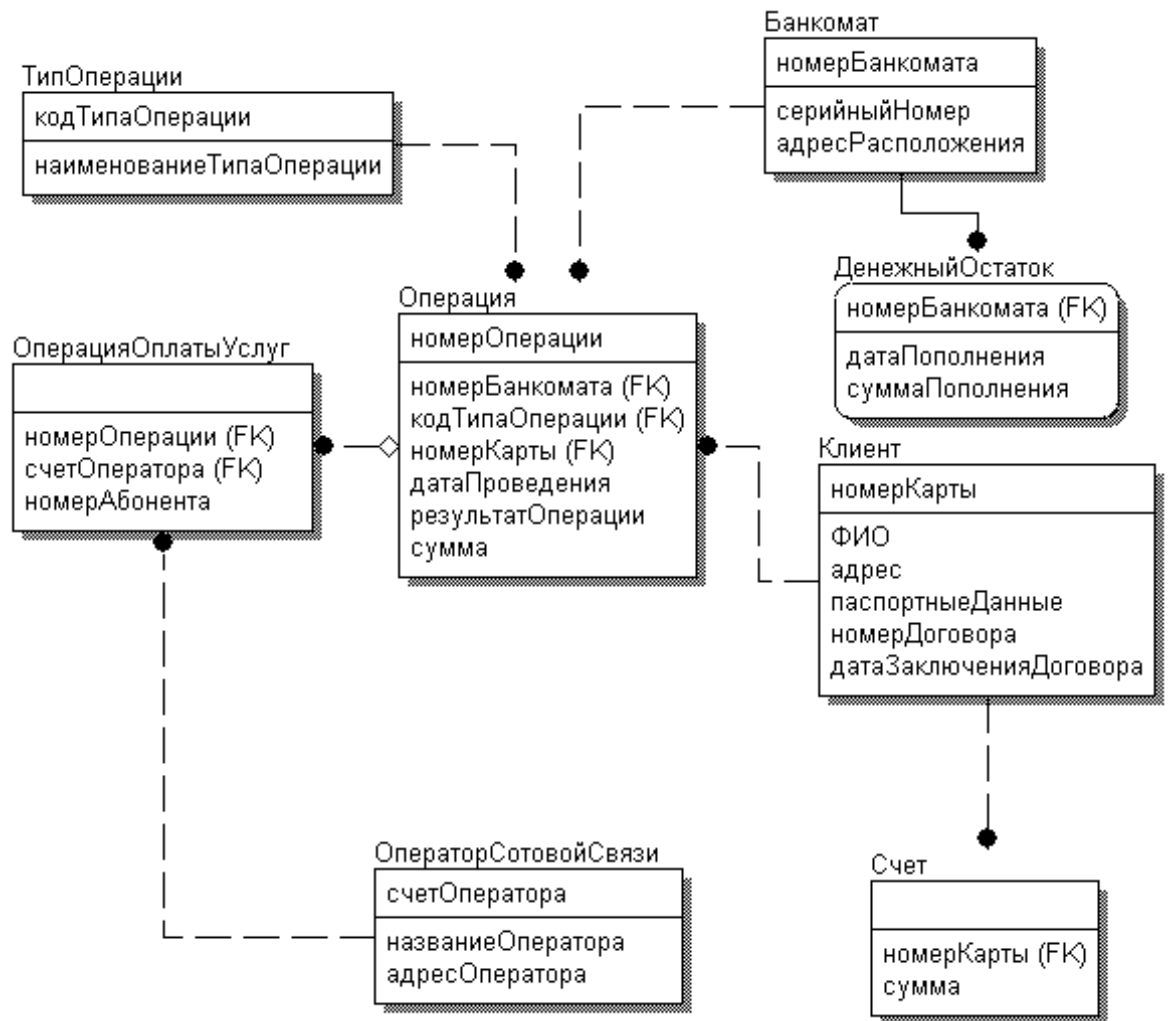


Рисунок 2.1 Структура базы данных

3. Описание архитектуры ПС.

Программное средство мониторинга финансовых операций, выполняемых через банкомат, реализуется в виде набора модулей:

- исполняемый модуль «АРМ Оператора»;
- исполняемый модуль «АРМ Администратора»;
- база данных;
- модуль формирования отчета;
- модуль анализа.

Компонентная структура ПС представлена на рисунке 2.2 в виде UML-диаграммы компонентов.

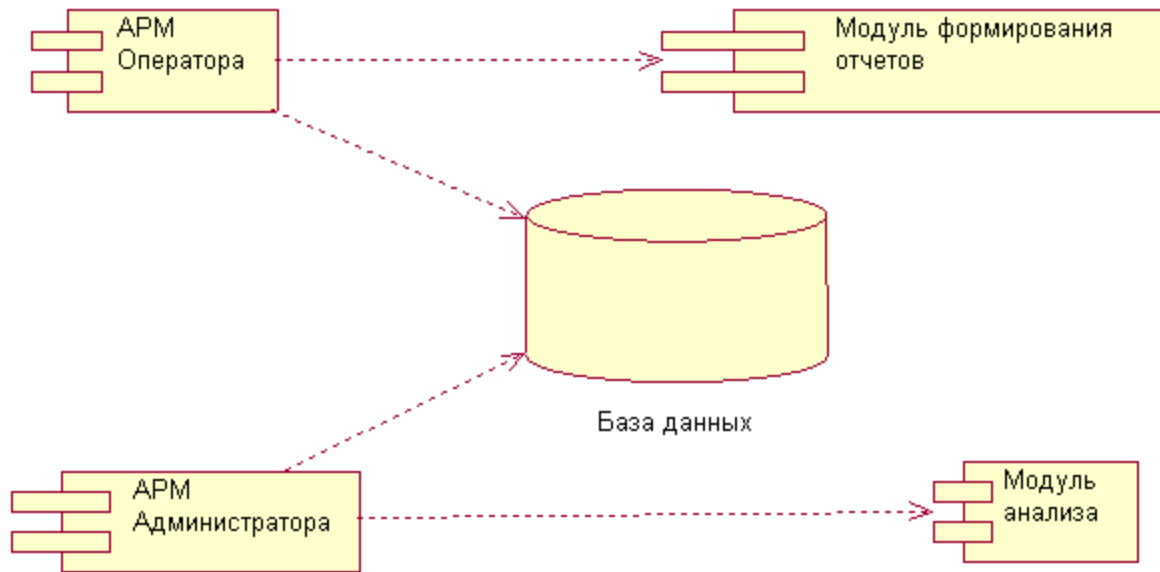


Рисунок 2.2 UML-диаграмма компонентов

Зависимость компонентов в соответствии с требованиями нотации UML показана на диаграмме в виде пунктирных стрелок.

Для описания функциональных возможностей реализуемого ПС составим UML-диаграмму вариантов использования (рис. 2.3). Варианты использования должны соответствовать требуемым функциям программного средства.

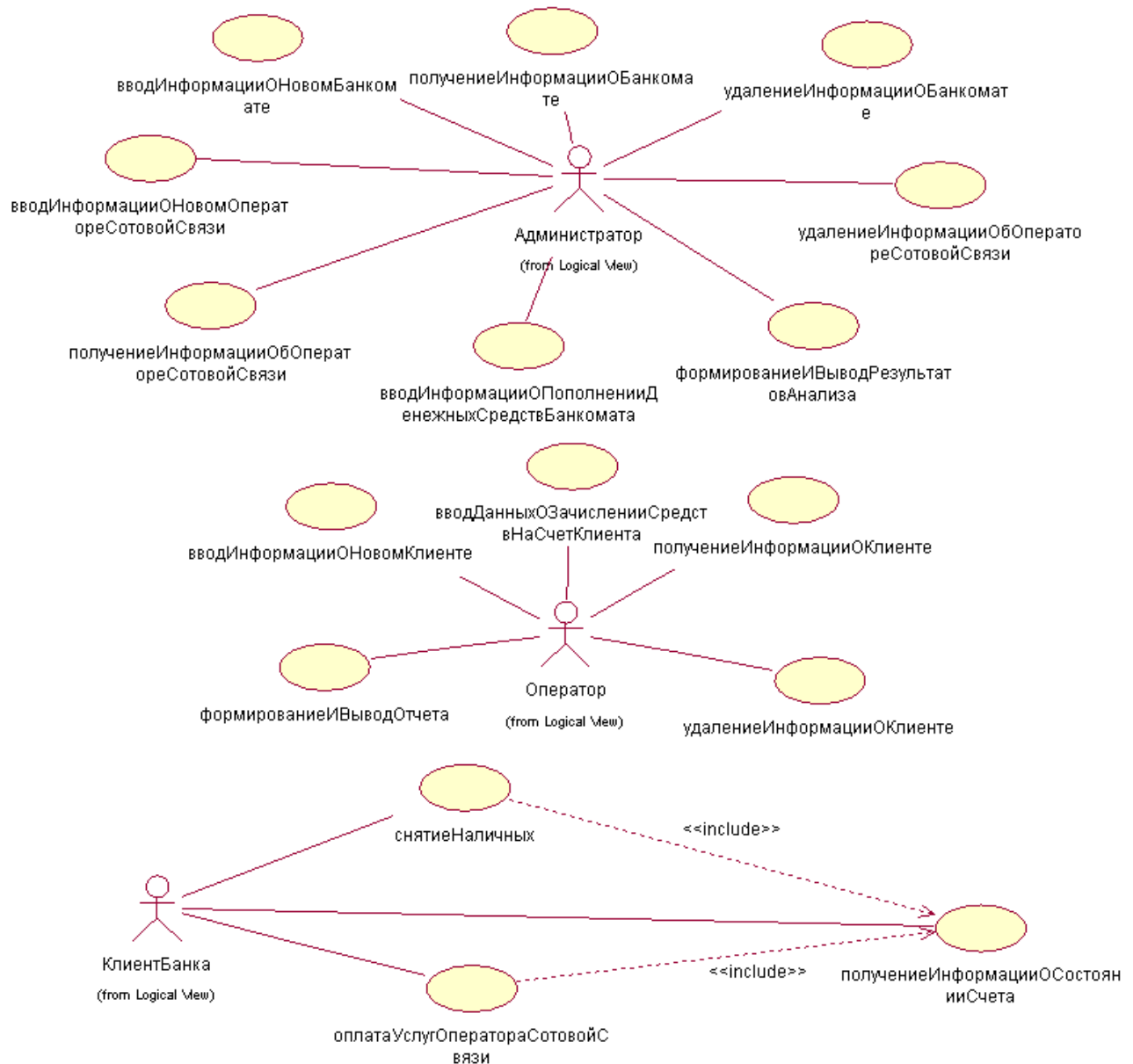


Рисунок 2.3 Диаграмма вариантов использования

Функции, выполнение которых инициируется администратором, оператором ПС или клиентом Банка, отмечены на диаграмме классов (рис. 2.3) в виде ассоциативной связи (сплошная прямая линия) между соответствующими сущностями и вариантами использования.

Для описания внутренней структуры реализуемого ПС построим диаграмму классов, построенную в соответствии с требованиями нотации UML (рис. 2.4).

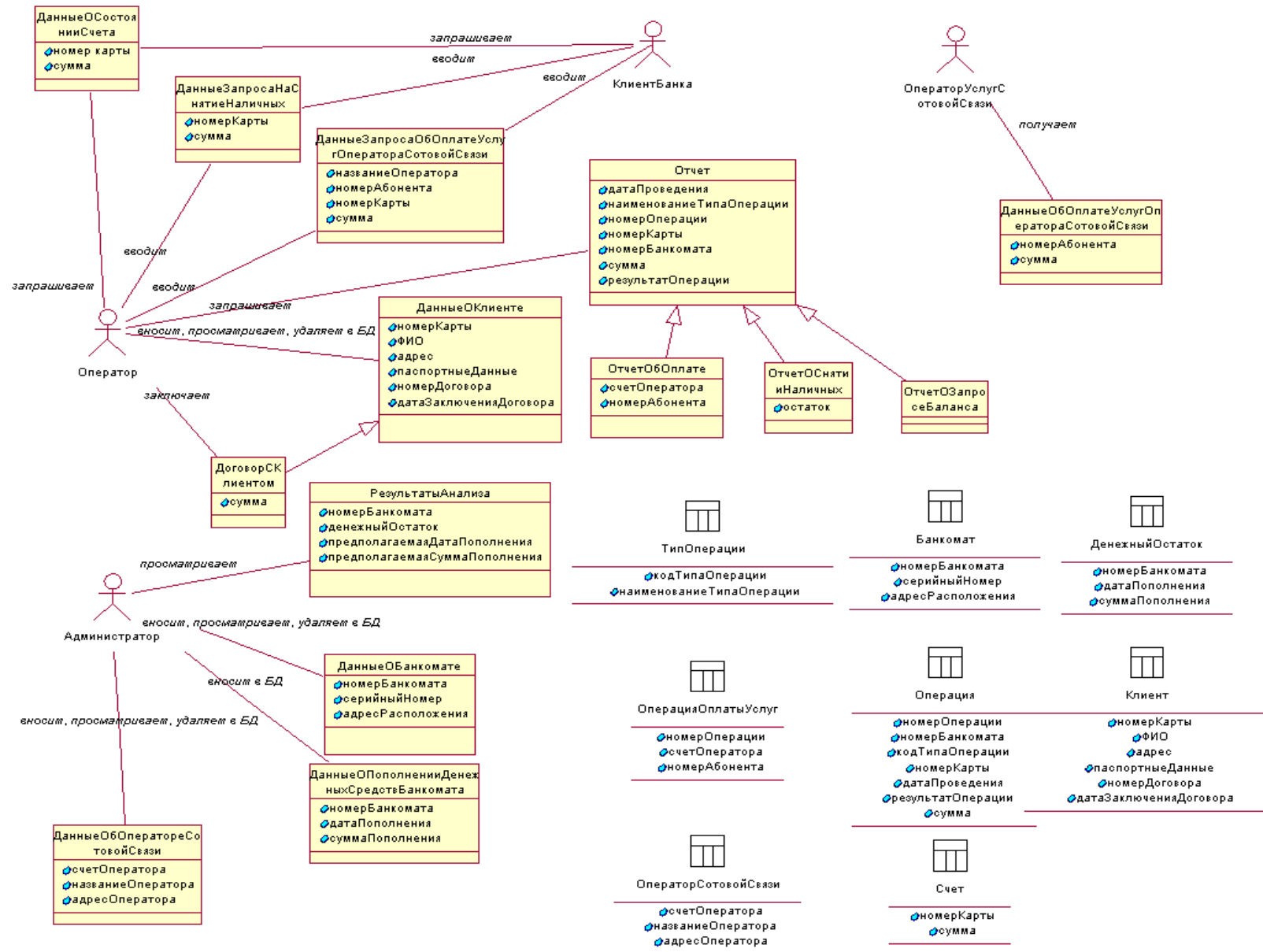


Рисунок 2.4 Диаграмма классов (внутренняя структура ПС)

Классы «ТипОперации», «Банкомат», «ДенежныйОстаток», «ОперацияОплатыУслуг», «Операция», «Клиент», «ОператорСотовойСвязи», «Счет» являются таблицами базы данных, с которыми осуществляется работа программного средства. Для того, чтобы не загромождать диаграмму, в рассматриваемом примере не показаны ассоциативные связи между таблицами (рассмотрены в пункте 2 «Описание базы данных»), а также их связь с остальными классами (учтены в таблице 2.1).

Варианты использования (рис. 2.3) как элементарные функции программного средства предполагают чтение и запись атрибутов определенных классов (рис. 2.4). В таблице 2.1 показаны классы и атрибуты классов для каждого из вариантов использования.

Атрибуты, используемые в рамках процессов, определяемых вариантами использования

Таблица 2.1

Вариант использования	Класс	Атрибут
вводИнформацииОНовомКлиенте	«ДанныеОКлиенте»	номерКарты, ФИО, адрес, паспортныеДанные, номерДоговора, датаЗаключенияДоговора
	«Клиент»	номерКарты, ФИО, адрес, паспортныеДанные, номерДоговора, датаЗаклученияДоговора
вводДанныхОЗачисленииСредствНаСчетКлиента	«ДанныеОКлиенте», «ДоговорСКлиентом»	номерКарты, сумма
	«Счет»	номерКарты, сумма
получениеИнформацииОКлиенте	«Клиент»	номерКарты, ФИО, адрес, паспортныеДанные, номерДоговора, датаЗаклученияДоговора
	«ДанныеОКлиенте»	номерКарты, ФИО, адрес, паспортныеДанные, номерДоговора, датаЗаклученияДоговора
формированиеИВыводОтчета	«Отчет», «ОтчетОбОплате», «ОтчетОСнятииНаличных», «ОтчетОЗапросеБаланса»	датаПроведения, наименованиеТипаОперации, номерОперации, номерКарты, номерБанкомата, сумма, результатОперации, счетОператора, номерАбонента, остаток

	«Операция»	номерОперации, номерБанкомата, кодТипаОперации, номерКарты, датаПроведения, результатОперации, сумма
	«ОперацияОп латыУслуг»	номерОперации, счетОператора, номерАбонента
	«ТипОперации »	кодТипаОперации, наименованиеТипаОперац ии
	«Счет»	номерКарты, сумма
удалениеИнформацииОКлиенте	«ДанныеОКли енте»	номерКарты
вводИнформацииОНовомБанкомате	«Банкомат»	номерБанкомата, серийныйНомер, адресРасположения
	«ДанныеОБанк омате»	номерБанкомата, серийныйНомер, адресРасположения
получениеИнформацииОБанкомате	«Банкомат»	номерБанкомата, серийныйНомер, адресРасположения
	«ДанныеОБанк омате»	номерБанкомата, серийныйНомер, адресРасположения
удалениеИнформацииОБанкомате	«ДанныеОБанк омате»	номерБанкомата
вводИнформацииОНовомОператореСотово йСвязи	«ОператорСот овойСвязи»	счетОператора, названиеОператора, адресОператора
	«ДанныеОбОп ератореСотово йСвязи»	счетОператора, названиеОператора, адресОператора
получениеИнформацииОбОператореСотово йСвязи	«ОператорСот овойСвязи»	счетОператора, названиеОператора, адресОператора
	«ДанныеОбОп ератореСотово йСвязи»	счетОператора, названиеОператора, адресОператора
удалениеИнформацииОбОператореСотовой Связи	«ДанныеОбОп ератореСотово йСвязи»	счетОператора
вводИнформацииОПополненииДенежныхС редствБанкомата	«ДенежныйОс таток»	номерБанкомата, датаПополнения, суммаПополнения

	«Данные О Пополнении Денежных Средств Банкомата»	номер Банкомата, дата Пополнения, сумма Пополнения
формирование И Вывод Результатов Анализа	«Результаты Анализа»	номер Банкомата, денежный Остаток, предполагаемая Дата Пополнения Денежных Средств, предполагаемая Сумма Пополнения Денежных Средств
	«Денежный Остаток»	номер Банкомата, дата Пополнения, сумма Пополнения
	«Операция»	номер Банкомата, код Типа Операции, дата Проведения, результат Операции, сумма
снятие Наличных	«Банкомат»	номер Банкомата, серийный Номер
	«Операция»	номер Операции, номер Банкомата, код Типа Операции, номер Карты, дата Проведения, результат Операции, сумма
	«Клиент»	дата Заключения Договора, номер Карты
	«Счет»	номер Карты, сумма
	«Данные Запроса На Снятие Наличных»	номер Карты, сумма
оплата Услуг Оператора Сотовой Связи	«Банкомат»	номер Банкомата, серийный Номер
	«Операция»	номер Операции, номер Банкомата, код Типа Операции, номер Карты, дата Проведения, результат Операции, сумма
	«Клиент»	дата Заключения Договора, номер Карты
	«Счет»	номер Карты, сумма
	«Оператор Сотовой Связи»	счет Оператора, название Оператора
	«Операция Оплаты Услуг»	номер Операции, счет Оператора, номер Абонента

	«ДанныеЗапросаОбОплатеУслугОператораСотовойСвязи»	названиеОператора, номерАбонента, номерКарты, сумма
	«ДанныеОбОплатеУслугОператораСотовойСвязи»	номерАбонента, сумма
получениеИнформацииОСостоянииСчета	«Банкомат»	номерБанкомата, серийныйНомер
	«Операция»	номерОперации, номерБанкомата, кодТипаОперации, номерКарты, датаПроведения, результатОперации, сумма
	«Клиент»	датаЗаключенияДоговора, номерКарты
	«Счет»	номерКарты, сумма
	«ДанныеОСостоянииСчета»	номерКарты, сумма

4. Расчет трудоемкости разработки ПС

Шаг 1. Определяем границу программного средства. В границу ПС входят:

- класс «Клиент»;
- класс «Банкомат»;
- класс «Операция»;
- класс «ТипОперации»;
- класс «Счет»;
- класс «ДенежныйОстаток»;
- класс «ОперацияОплатыУслуг»;
- класс «ОператорСотовойСвязи»;
- вариант использования «вводИнформацииОНовомБанкомате»;
- вариант использования «получениеИнформацииОБанкомате»;
- вариант использования «удалениеИнформацииОБанкомате»;
- вариант использования «вводИнформацииОНовомОператореСотовойСвязи»;
- вариант использования «получениеИнформацииОбОператореСотовойСвязи»;
- вариант использования «удалениеИнформацииОбОператореСотовойСвязи»;
- вариант использования
«вводИнформацииОПополнениеДенежныхСредствБанкомата»;
- вариант использования «формированиеИВыводРезультатовАнализа»;
- вариант использования «вводИнформацииОНовомКлиенте»;
- вариант использования «получениеИнформацииОКлиенте»;
- вариант использования «удалениеИнформацииОКлиенте»;
- вариант использования «формированиеИВыводОтчета»;
- вариант использования «снятиеНаличных»;
- вариант использования «оплатаУслугОператораСотовойСвязи»;
- вариант использования «получениеИнформацииОСостоянииСчета»;
- вариант использования «вводДанныхОЗачисленииСредствНаСчетКлиента».

В границу ПС не входят:

- класс «ДанныеОбОператореСотовойСвязи»;
- класс «ДанныеОПополненииДенежныхСредствБанкомата»;
- класс «ДанныеОБанкомате»;

- класс «РезультатыАнализа»;
- класс «ДоговорСКлиентом»;
- класс «ДанныеОКлиенте»;
- класс «ДанныеЗапросаОбОплатеУслугОператораСотовойСвязи»;
- класс «ДанныеЗапросаНаСнятиеНаличных»;
- класс «ДанныеОСостоянииСчета»;
- класс «Отчет»;
- класс «ОтчетОбОплате»;
- класс «ОтчетОСнятииНаличных»;
- класс «ОтчетОЗапросеБаланса»;
- класс «ДанныеОбОплатеУслугОператораСотовойСвязи»;

Перечисленные классы с точки зрения логической организации ПС представляют собой формы ввода (все кроме «Отчет» и классов, связанных с ним отношением обобщения, класса «РезультатыАнализа», класса «ДанныеОбОплатеУслугОператораСотовойСвязи») либо представления информации (класс «Отчет» и классы, связанные с ним отношением обобщения, класс «РезультатыАнализа», а также класс «ДанныеОбОплатеУслугОператораСотовойСвязи», который является интерфейсом для выхода посредством сетевого интерфейса данных об оплате услуг оператору сотовой связи).

Шаг 2. Определяем границу расчета. В границу расчета входят:

- класс «Клиент»;
- класс «Банкомат»;
- класс «Операция»;
- класс «ТипОперации»;
- класс «Счет»;
- класс «ДенежныйОстаток»;
- класс «ОперацияОплатыУслуг»;
- класс «ОператорСотовойСвязи»;
- класс «ДанныеОбОператореСотовойСвязи»;
- класс «ДанныеОПополненииДенежныхСредствБанкомата»;
- класс «ДанныеОБанкомате»;
- класс «РезультатыАнализа»;
- класс «ДоговорСКлиентом»;

- класс «ДанныеОКлиенте»;
- класс «ДанныеЗапросаОбОплатеУслугОператораСотовойСвязи»;
- класс «ДанныеЗапросаНаСнятиеНаличных»;
- класс «ДанныеОСостоянииСчета»;
- класс «Отчет»;
- класс «ОтчетОбОплате»;
- класс «ОтчетОСнятииНаличных»;
- класс «ОтчетОЗапросеБаланса»;
- класс «ДанныеОбОплатеУслугОператораСотовойСвязи»;
- вариант использования «вводИнформацииОНовомБанкомате»;
- вариант использования «получениеИнформацииОБанкомате»;
- вариант использования «удалениеИнформацииОБанкомате»;
- вариант использования «вводИнформацииОНовомОператореСотовойСвязи»;
- вариант использования «получениеИнформацииОбОператореСотовойСвязи»;
- вариант использования «удалениеИнформацииОбОператореСотовойСвязи»;
- вариант использования
«вводИнформацииОПополнениеДенежныхСредствБанкомата»;
- вариант использования «формированиеИВыводРезультатовАнализа»;
- вариант использования «вводИнформацииОНовомКлиенте»;
- вариант использования «получениеИнформацииОКлиенте»;
- вариант использования «удалениеИнформацииОКлиенте»;
- вариант использования «формированиеИВыводОтчета»;
- вариант использования «снятиеНаличных»;
- вариант использования «оплатаУслугОператораСотовойСвязи»;
- вариант использования «получениеИнформацииОСостоянииСчета»;
- вариант использования «вводДанныхОЗачисленииСредствНаСчетКлиента».

Шаг 3. Определяем логические файлы ПС и их сложность. Выделяем на диаграмме классов те, группы классов, которые связаны отношением обобщения (наследования) и композиции или агрегации.

Класс «ДанныеОКлиенте» является обобщением для класса «ДоговорСКлиентом», поэтому вместе они образуют один логический файл.

Класс «Отчет» является обобщением для классов «ОтчетОбОплате», «ОтчетОСнятииНаличных» и «ОтчетОЗапросеБаланса», поэтому вместе они образуют один логический файл.

Классов, имеющих отношение композиции или агрегации на диаграмме нет. Каждый из оставшихся классов является логическим файлом.

Аналогичным образом, необходимо выделить варианты использования, связанные отношением обобщения. В нашем случае таких элементов на диаграмме вариантов использования нет.

В границу ПС согласно шагу 1 входят классы:

- класс «Клиент»;
- класс «Банкомат»;
- класс «Операция»;
- класс «ТипОперации»;
- класс «Счет»;
- класс «ДенежныйОстаток»;
- класс «ОперацияОплатыУслуг»;
- класс «ОператорСотовойСвязи»;

Соответствующие данным классам логические файлы являются внутренними логическими файлами. Все остальные классы, не входящие в границу ПС, но входящие в границу расчета, являются внешними логическими файлами.

Сложность логических файлов зависит от количества их элементов данных и числа записей. Данные значения определяются в соответствии с Методикой. Результаты приведены в таблице 3.1.

Количество элементов данных и записей логических файлов

Таблица 3.1

Логический файл	Количество элементов данных	Количество записей
Внутренний логический файл, состоящий из класса «Клиент»	7 (все атрибуты класса + первичный ключ «номерКарты»)	1
Внутренний логический файл, состоящий из класса «Банкомат»	4 (все атрибуты класса + первичный ключ «номерБанкомата»)	1
Внутренний логический файл, состоящий из класса «ОператорСотовойСвязи»	4 (все атрибуты класса + первичный ключ «счетОператора»)	1
Внутренний логический файл, состоящий из класса «Операция»	8 (все атрибуты класса + первичный ключ «номерОперации»)	1
Внутренний логический файл, состоящий из класса «ОперацияОплатыУслуг»	3 (все атрибуты класса)	1
Внутренний логический файл, состоящий из класса «Счет»	2 (все атрибуты класса)	1
Внутренний логический файл, состоящий из класса «ТипОперации»	3 (все атрибуты класса + первичный ключ «кодТипаОперации»)	1
Внутренний логический файл, состоящий из класса «ДенежныйОстаток»	3 (все атрибуты класса)	1
Внешний логический файл, состоящий из класса «ДанныеОбОператореСотовойСвязи»	3 (все атрибуты класса)	1
Внешний логический файл, состоящий из класса «ДанныеОПополнении ДенежныхСредствБанкомата»	3 (все атрибуты класса)	1
Внешний логический файл, состоящий из класса «ДанныеОБанкомате»	3 (все атрибуты класса)	1
Внешний логический файл, состоящий из класса «РезультатыАнализа»	4 (все атрибуты класса)	1

Внешний логический файл, состоящий из класса «ДанныеОСостоянииСчета»	2 (все атрибуты класса)	1
Внешний логический файл, состоящий из класса «ДанныеЗапросаНаСнятиеНаличных»	2 (все атрибуты класса)	1
Внешний логический файл, состоящий из класса «ДанныеЗапросаОбОплате УслугОператораСотовойСвязи»	4 (все атрибуты класса)	1
Внешний логический файл, состоящий из класса «ДанныеОбОплатеУслуг ОператораСотовойСвязи»	2 (все атрибуты класса)	1
Внутренний логический файл, состоящий из группы классов «Отчет», «ОтчетОбОплате», «ОтчетОСнятииНаличных», «ОтчетОЗапросеБаланса»	10 (все атрибуты классов)	3 (обобщаемые классы – «ОтчетОбОплате», «ОтчетОСнятииНаличных», «ОтчетОЗапросеБаланса»)
Внутренний логический файл, состоящий из группы классов «ДоговорСКлиентом», «ДанныеОКлиенте»	7 (все атрибуты классов)	1 (обобщаемые классы – «ДоговорСКлиентом»)

На основе количества элементов данных и количества записей, входящих в состав каждого логического файла, по таблице 4.1 Методики определяем уровень функциональной сложности для каждого логического файла:

Уровни сложности логических файлов

Таблица 3.2

Логический файл	Уровень сложности
Внутренний логический файл, состоящий из класса «Клиент»	низкий
Внутренний логический файл, состоящий из класса «Банкомат»	низкий
Внутренний логический файл, состоящий из класса «ОператорСотовойСвязи»	низкий
Внутренний логический файл, состоящий из класса «Операция»	низкий
Внутренний логический файл, состоящий из класса «ОперацияОплатыУслуг»	низкий
Внутренний логический файл, состоящий из класса «Счет»	низкий
Внутренний логический файл, состоящий из класса «ТипОперации»	низкий

Внутренний логический файл, состоящий из класса «ДенежныйОстаток»	низкий
Внешний логический файл, состоящий из класса «ДанныеОбОператореСотовойСвязи»	низкий
Внешний логический файл, состоящий из класса «ДанныеОПополнении ДенежныхСредствБанкомата»	низкий
Внешний логический файл, состоящий из класса «ДанныеОБанкомате»	низкий
Внешний логический файл, состоящий из класса «РезультатыАнализа»	низкий
Внешний логический файл, состоящий из класса «ДанныеОСостоянииСчета»	низкий
Внешний логический файл, состоящий из класса «ДанныеЗапросаНаСнятиеНаличных»	низкий
Внешний логический файл, состоящий из класса «ДанныеЗапросаОбОплате УслугОператораСотовойСвязи»	низкий
Внешний логический файл, состоящий из класса «ДанныеОбОплатеУслуг ОператораСотовойСвязи»	низкий
Внутренний логический файл, состоящий из группы классов «Отчет», «ОтчетОбОплате», «ОтчетОСнятииНаличных», «ОтчетОЗапросеБаланса»	низкий
Внутренний логический файл, состоящий из группы классов «ДоговорСКлиентом», «ДанныеОКлиенте»	низкий

Шаг 4. Выделяем транзакционные функции ПС и определяем их сложность.

Согласно Методике, разделяем транзакционные функции на три группы: единицы входной информации, единицы выходной информации и внешние запросы.

1 группа. Единицы входной информации:

- вариант использования «вводИнформацииОНовомКлиенте»;
- вариант использования «вводДанныхОЗачисленииСредствНаСчетКлиента»;
- вариант использования «удалениеИнформацииОКлиенте»;
- вариант использования «вводИнформацииОНовомБанкомате»;
- вариант использования «удалениеИнформацииОБанкомате»;
- вариант использования «вводИнформацииОНовомОператореСотовойСвязи»;
- вариант использования «удалениеИнформацииОбОператореСотовойСвязи»;
- вариант использования «вводИнформацииОПополненииДенежных СредствБанкомата»;
- вариант использования «снятиеНаличных»;

- вариант использования «оплатаУслугОператораСотовойСвязи».

2 группа. Единицы выходной информации:

- вариант использования «формированиеИВыводОтчета»;
- вариант использования «формированиеИВыводРезультатовАнализа»;
- вариант использования «получениеИнформацииОСостоянииСчета».

3 группа. Внешние запросы:

- вариант использования «получениеИнформацииОКлиенте»;
- вариант использования «получениеИнформацииОБанкомате»;
- вариант использования «получениеИнформацииОбОператореСотовойСвязи».

Определяем характеристики единиц входной информации:

Единицы входной информации

Таблица 3.3

Единица входной информации	Количество элементов данных, пересекающих границу ПС	Количество используемых логических файлов
вводИнформацииОНовомКлиенте	7 (все атрибуты логического файла + элемент на возможность вывода транзакционной функцией сообщения об ошибке)	2
вводДанныхОЗачисленииСредствНаСчетКлиента	3 (все атрибуты логического файла + элемент на возможность вывода транзакционной функцией сообщения об ошибке)	2
удалениеИнформацииОКлиенте	2 (все атрибуты логического файла + элемент на возможность вывода транзакционной функцией сообщения об ошибке)	1

вводИнформацииОНовомБанкомате	4 (все атрибуты логического файла + элемент на возможность вывода транзакционной функцией сообщения об ошибке)	2
удалениеИнформацииОБанкомате	2 (все атрибуты логического файла + элемент на возможность вывода транзакционной функцией сообщения об ошибке)	1
вводИнформацииОНовомОператореСотовойСвязи	4 (все атрибуты логического файла + элемент на возможность вывода транзакционной функцией сообщения об ошибке)	2
удалениеИнформацииОбОператореСотовойСвязи	2 (все атрибуты логического файла + элемент на возможность вывода транзакционной функцией сообщения об ошибке)	1
вводИнформацииОПополненииДенежных СредствБанкомата	4 (все атрибуты логического файла + элемент на возможность вывода транзакционной функцией сообщения об ошибке)	2

снятиеНаличных	6 (номерБанкомата, кодТипаОперации, номерКарты, сумма + элемент на возможность вывода транзакционной функцией сообщения об ошибке + элемент на возможность различного способа выполнения элементарного процесса – вызывается клиентом банка через интерфейс банкомата либо оператором ПС)	5
оплатаУслугОператораСотовойСвязи	7 (номерБанкомата, кодТипаОперации, номерКарты, сумма, номерАбонента + элемент на возможность вывода транзакционной функцией сообщения об ошибке + элемент на возможность различного способа выполнения элементарного процесса – вызывается клиентом банка через интерфейс банкомата либо оператором ПС)	8

Определяем уровень сложности единиц входной информации:

Сложность единиц входной информации

Таблица 3.4

Единица входной информации	Уровень сложности
вводИнформацииОНовомКлиенте	средний
вводДанныхОЗачисленииСредствНаСчетКлиента	низкий
удалениеИнформацииОКлиенте	низкий
вводИнформацииОНовомБанкомате	низкий
удалениеИнформацииОБанкомате	низкий
вводИнформацииОНовомОператореСотовойСвязи	низкий
удалениеИнформацииОбОператореСотовойСвязи	низкий
вводИнформацииОПополненииДенежных СредствБанкомата	низкий
снятиеНаличных	высокий
оплатаУслугОператораСотовойСвязи	высокий

Определяем характеристики единиц выходной информации:

Единицы выходной информации

Таблица 3.5

Единица выходной информации	Количество элементов данных, пересекающих границу ПС	Количество используемых логических файлов
формированиеИВыводОтчета	11 (датаПроведения, наименованиеТипаОперации, номерОперации, номерКарты, номерБанкомата, сумма, результатОперации, счетОператора, номерАбонента, остаток+ элемент на возможность вывода транзакционной функцией сообщения об ошибке)	5
формированиеИВывод РезультатовАнализа	5 (номерБанкомата, денежныйОстаток, предполагаемаяДатаПополнения ДенежныхСредств, предполагаемаяСуммаПополнения ДенежныхСредств+ элемент на возможность вывода транзакционной функцией сообщения об ошибке)	3

получениеИнформацииО СостоянииСчета	4 (номерКарты, сумма+ элемент на возможность вывода транзакционной функцией сообщения об ошибке+ элемент на возможность различного способа выполнения элементарного процесса – вызывается клиентом банка через интерфейс банкомата либо оператором ПС)	5
--	--	---

Определяем уровень сложности единиц выходной информации:

Сложность единиц выходной информации

Таблица 3.6

Единица выходной информации	Уровень сложности
формированиеИВыводОтчета	высокий
формированиеИВыводРезультатовАнализа	низкий
получениеИнформацииОСостоянииСчета	средний

Определяем характеристики внешних запросов:

Внешние запросы

Таблица 3.7

Внешние запросы	Количество элементов данных, пересекающих границу ПС	Количество используемых логических файлов
получениеИнформацииОКлиенте	7 (все атрибуты логического файла + элемент на возможность вывода транзакционной функцией сообщения об ошибке)	2
получениеИнформацииОБанкомате	4 (все атрибуты логического файла + элемент на возможность вывода транзакционной функцией сообщения об ошибке)	2

получениеИнформацииОбОператореСотовойСвязи	4 (все атрибуты логического файла + элемент на возможность вывода транзакционной функцией сообщения об ошибке)	2
--	--	---

Определяем уровень сложности внешних запросов:

Уровень сложности внешних запросов

Таблица 3.8

Внешний запрос	Уровень сложности
получениеИнформацииОКлиенте	средний
получениеИнформацииОБанкомате	низкий
получениеИнформацииОбОператореСотовойСвязи	низкий

Шаг 5. Определяем нескорректированный объем ПС в функциональных точках:

Количество функциональных точек, соответствующих элементу данных или транзакционной функции

Таблица 3.9

Элемент данных или транзакционная функция	Количество функциональных точек
Внутренний логический файл, состоящий из класса «Клиент»	7
Внутренний логический файл, состоящий из класса «Банкомат»	7
Внутренний логический файл, состоящий из класса «ОператорСотовойСвязи»	7
Внутренний логический файл, состоящий из класса «Операция»	7
Внутренний логический файл, состоящий из класса «ОперацияОплатыУслуг»	7
Внутренний логический файл, состоящий из класса «Счет»	7
Внутренний логический файл, состоящий из класса «ТипОперации»	7
Внутренний логический файл, состоящий из класса «ДенежныйОстаток»	7
Внешний логический файл, состоящий из класса «ДанныеОбОператореСотовойСвязи»	5
Внешний логический файл, состоящий из класса «ДанныеОПополнении ДенежныхСредствБанкомата»	5
Внешний логический файл, состоящий из класса «ДанныеОБанкомате»	5

Внешний логический файл, состоящий из класса «РезультатыАнализа»	5
Внешний логический файл, состоящий из класса «ДанныеОСостоянииСчета»	5
Внешний логический файл, состоящий из класса «ДанныеЗапросаНаСнятиеНаличных»	5
Внешний логический файл, состоящий из класса «ДанныеЗапросаОбОплате УслугОператораСотовойСвязи»	5
Внешний логический файл, состоящий из класса «ДанныеОбОплатеУслуг ОператораСотовойСвязи»	5
Внутренний логический файл, состоящий из группы классов «Отчет», «ОтчетОбОплате», «ОтчетОСнятииНаличных», «ОтчетОЗапросеБаланса»	5
Внутренний логический файл, состоящий из группы классов «ДоговорСКлиентом», «ДанныеОКлиенте»	5
Единица входной информации «вводИнформацииОНовомКлиенте»	4
Единица входной информации «вводДанныхОЗачисленииСредствНаСчетКлиента»	3
Единица входной информации «удалениеИнформацииОКлиенте»	3
Единица входной информации «вводИнформацииОНовомБанкомате»	3
Единица входной информации «удалениеИнформацииОБанкомате»	3
Единица входной информации «вводИнформацииОНовомОператореСотовойСвязи»	3
Единица входной информации «удалениеИнформацииОбОператореСотовойСвязи»	3
Единица входной информации «вводИнформацииО ПополненииДенежныхСредствБанкомата»	3
Единица входной информации «снятиеНаличных»	6
Единица входной информации «оплатаУслугОператораСотовойСвязи»	6
Единица выходной информации «формированиеИВыводОтчета»	7
Единица выходной информации «формированиеИВыводРезультатовАнализа»	4
Единица выходной информации «получениеИнформацииОСостоянииСчета»	5
Внешний запрос «получениеИнформацииОКлиенте»	4
Внешний запрос «получениеИнформацииОБанкомате»	3
Внешний запрос «получениеИнформацииОбОператореСотовойСвязи»	3

Нескорректированный размер ПС ($V_{\phi m}$) в функциональных точках определяется как сумма количества функциональных точек, соответствующих элементам данных и транзакционным функциям. $V_{\phi m} = 169$.

Шаг 6. Определяем размер ПС в строках исходного кода:

$$V = V_{\phi m} * 125 = 21125$$

Ниже представлен отчет по расчету трудоемкости.

РАСЧЕТ ТРУДОЕМКОСТИ С ИСПОЛЬЗОВАНИЕМ ПО АРМ

Расчет трудоемкости разработки ПС (метод ФБО)

Параметры расчета

Параметр	Значение
Имя расчета	Программное средство мониторинга финансовых операций, выполняемых через банкомат
Тип расчета	Расчет трудоемкости разработки ПС (метод ФБО)
Версия модели	2.5.1.0
Фонд времени разработчика [дней/мес.]	21
Тип ограничения	По продолжительности этапа [мес.]
Тип структуры программного средства	ПС с набором библиотек (=1.00)
Технологии взаимодействия модулей	Оригинальный протокол взаимодействия (sockets) (=0.80)

Коэффициенты уровня расчета

Коэффициент	Уровень значения
Степень новизны ПС	Принципиально новое ПС, не имеющее доступных аналогов на прежнем типе ТС/ОС (=1.06)
Требования к надежности	Сбои ПС приводят к потерям рабочей информации (=1.06)
Требования к производительности	Повышенные требования к производительности (=1.12)
Информативность документации	Повышенный объем для жизненного цикла данного ПС (=1.07)
Опыт разработки ПС подобного типа	Более 5 разработанных ПС (компонентов АС) (=0.81)
Влияние сроков работ на трудоемкость	Равно или более 100 (=1.00)

Коэффициенты уровня подпроцесса

Подпроцесс	Коэффициент	Уровень значения
Анализ требований	Уровень квалификации аналитиков	Высокий (=0.87)
Анализ требований	Опыт работы аналитиков в данной предметной области	5 разработанных ПС (компонентов АС) (=0.87)
Проектирование	Уровень квалификации проектировщиков	Высокий (=0.92)
Проектирование	Опыт работы проектировщиков с используемыми средствами	5 разработанных ПС (компонентов АС) (=0.92)

	проектирования	
Проектирование	Используемые средства проектирования	Средства проектирования БД (BPWIN, ERWIN/ERX, Designer 2000, DataDirect Explorer, Oracle Designer, Silverrun RBM и BPM, Together J) (=1.03)
Программирование	Уровень квалификации программистов	Высокий (=0.95)
Программирование	Используемая среда разработки	Интегрированные среды разработки (Visual Studio) (=1.00)
Тестирование	Уровень квалификации персонала, осуществляющего тестирование	Высокий (=0.89)
Тестирование	Используемые средства тестирования	Автоматизированные средства тестирования использовались (=0.76)
Тестирование	Размер БД	$10 \leq D/P < 100$ (=1.00)
Ввод в действие ПС	Уровень квалификации персонала, осуществляющего ввод в действие ПС	Высокий (=0.94)

Дерево элементов

Элемент	Описание	Тип	Количество элементов данных (DET)	Количество записей (RET)	Объем кода на условном языке [строк]	Объем [ФТ]
Клиент		Внутренний логический файл	7	1	875	7
Банкомат		Внутренний логический файл	4	1	875	7
ОператорСотовойСвязи		Внутренний логический файл	4	1	875	7
Операция		Внутренний логический файл	8	1	875	7
ОперацияОплатыУслуг		Внутренний логический файл	3	1	875	7
Счет		Внутренний логический файл	2	1	875	7
ТипОперации		Внутренний логический файл	3	1	875	7
ДенежныйОстаток		Внутренний логический файл	3	1	875	7
ДанныеОбОператореСотовойСвязи		Внешний логический файл	3	1	625	5
ДанныеОПополненииДенежныхСредствБанкомата		Внешний логический файл	3	1	625	5
ДанныеОБанкомате		Внешний логический файл	3	1	625	5
РезультатыАнализа		Внешний логический файл	4	1	625	5

ДанныеОСостоянииСчета		Внешний логический файл	2	1	625	5
ДанныеЗапросаНаСнятиеНаличных		Внешний логический файл	2	1	625	5
ДанныеЗапросаОбОплатеУслугОператораСотовойСвязи		Внешний логический файл	4	1	625	5
ДанныеОбОплатеУслугОператораСотовойСвязи		Внешний логический файл	2	1	625	5
Отчет, ОтчетОбОплате, ОтчетОСнятииНаличных, ОтчетОЗапросеБаланса		Внешний логический файл	10	3	625	5
ДоговорСКлиентом, ДанныеОКлиенте		Внешний логический файл	7	1	625	5
вводДанныхОЗачисленииСредствНаСчетКлиента		Единица входной информации	3	2	375	3
вводИнформацииОНовомКлиенте		Единица входной информации	7	2	500	4
удалениеИнформацииОКлиенте		Единица входной информации	2	1	375	3
вводИнформацииОНовомБанкомате		Единица входной информации	4	2	375	3
удалениеИнформацииОБанкомате		Единица входной информации	2	1	375	3
вводИнформацииОНовомОператореСотовойСвязи		Единица входной информации	4	2	375	3
удалениеИнформацииОбОператореСотовойСвязи		Единица входной информации	2	1	375	3
вводИнформацииОПополненииДенежныхСредств		Единица входной информации	4	2	375	3
снятиеНаличных		Единица входной информации	6	5	750	6

оплатаУслугОператораСотовойСвязи		Единица входной информации	7	8	750	6
формированиеИВыводОтчета		Единица выходной информации	11	5	875	7
формированиеИВыводРезультатов Анализа		Единица выходной информации	5	3	500	4
получениеИнформацииОСостоянииСчета		Единица выходной информации	4	5	625	5
получениеИнформацииОКлиенте		Внешний запрос	7	2	500	4
получениеИнформацииОБанкомате		Внешний запрос	4	2	375	3
получениеИнформацииОбОператореСотовойСвязи		Внешний запрос	4	2	375	3
Общий объем ПС:					21125	169

Расчет трудоемкости подпроцессов

Подпроцесс	Трудоемкость [чел.дн.]	Кол-во разработчиков [чел]	Продолжительность [мес]/[дн]
Анализ требований	4.48	0.11	2.00/42.00
Проектирование	61.90	0.98	3.00/63.00
Программирование	444.04	5.29	4.00/84.00
Тестирование	28.01	0.67	2.00/42.00
Ввод в действие ПС	5.56	0.26	1.00/21.00
Всего	543.99		12.00/252.00
Базовая трудоемкость	591.66		

ПРИЛОЖЕНИЕ 6

ПРИМЕР РАСЧЕТА ТРУДОЕМКОСТИ РАЗРАБОТКИ ПС С ИСПОЛЬЗОВАНИЕМ МЕТОДА РАСЧЕТА РАЗМЕРА ПС ПО ЭЛЕМЕНТАМ ГРАФИЧЕСКОГО ИНТЕРФЕЙСА ПОЛЬЗОВАТЕЛЯ

1. Функциональный состав ПС

Программное средство предназначено для оперативной подготовки платежных поручений, мемориальных ордеров и инкассовых документов.

Основные функции рассматриваемого программного средства:

- добавление информации оператором ПС о новом платежном документе в базу данных;
- редактирование реквизитов уже введенных в базу данных платежных документов;
- удаление из базы данных ошибочно введенных документов;
- ведение справочника БИК для корректного автозаполнения в документах реквизитов банков отправителя и получателя;
- ведение справочника операторов для идентификации автора сформированных документов;
- печать подготовленных платежных документов.

2. Выделение экземпляров программных объектов, которые будут разработаны в создаваемом ПС

2.1. Описание интерфейса пользователя

Интерфейс разрабатываемого ПС содержит следующий набор клиентских и диалоговых окон:

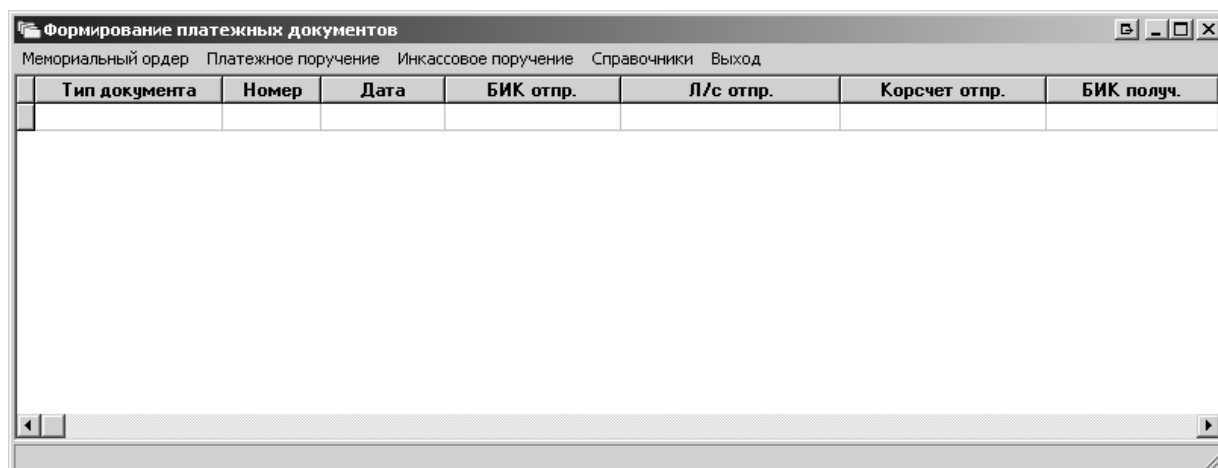


Рис. 2.1 Основное окно приложения

Таблица основного окна приложения на рис 2.1 содержит часть колонок, не отображаемых на экране по умолчанию. Для доступа к ним необходимо «пролистать» таблицу с помощью полосы горизонтальной прокрутки. Для определения функциональной сложности окна необходимо учесть все колонки таблицы:

- тип документа;
- номер;
- дата;
- БИК отправителя;
- л/с отправителя;
- корсчет отправителя;
- БИК получателя;
- л/с получателя;
- корсчет получателя;
- сумма;
- оператор.

Формирование

МЕМОРИАЛЬНЫЙ ОРДЕР № 1

Поступ. в банк плат. 16.11.2009

Банк плательщика БИК 1111 Коррсчет

Платательщик Дебет сч.№ ИНН

Наименование КПП

Банк получателя БИК 1111 Коррсчет

Получатель Кредит сч.№ ИНН

Наименование КПП

Сумма 0.00 Очередность 6

Назначение платежа

Ok Отмена

Рисунок 2.2 Окно для работы с документом «Мемориальный ордер»

Формирование

ПЛАТЕЖНОЕ ПОРУЧЕНИЕ № 1

Поступ. в банк плат. 16.11.2009 Вид плат. <не заполнено>

Банк плательщика БИК 1111 Коррсчет

Платательщик Дебет сч.№ ИНН

Наименование КПП

Банк получателя БИК 1111 Коррсчет

Получатель Кредит сч.№ ИНН

Наименование КПП

Сумма 0.00 Очередность 6

Назначение платежа

Ok Отмена

Рисунок 2.3 Окно для работы с документом «Платежное поручение»

Формирование

ИНКАССОВОЕ ПОРУЧЕНИЕ № 1 Срок для акцепта 16.11.2009

Поступ. в банк плат. 16.11.2009 Вид плат. <не заполнено>

Банк плательщика БИК 1111 Коррсчет

Плательщик Дебет сч.№ ИНН

Наименование КПП

Банк получателя БИК 1111 Коррсчет

Получатель Кредит сч.№ ИНН

Наименование КПП

Сумма 0.00 Очередность 6

Назначение платежа

Ok Отмена

Рисунок 2.4 Окно для работы с документом «Инкассовое поручение»

Справочник БИК

БИК	Тип КО	Наименование	н/п	Населенный пункт

Добавить
Удалить
Поиск
OK
Отмена

Рисунок 2.5 Окно справочника БИК

Справочник операторов

№	ФИО

Добавить
Удалить
OK
Отмена

Рисунок 2.6 Окно справочника операторов

№	Вид платежа

Добавить

Удалить

OK

Отмена

Рисунок 2.7 Окно справочника видов платежей

Начало работы с приложением

Оператор:

Текущая дата:

OK Отмена

Рисунок 2.8 Диалог для задания режима работы программы

2.2. Описание выходных документов

Разрабатываемое ПС должно формировать следующие (рис. 2.9 – 2.11) выходные документы с возможностью их вывода на печать.

МЕМОРИАЛЬНЫЙ ОРДЕР № Дата Форма № 0401108

Сумма прописью рублей копеек

ИНН

Платеж:

Банк платежа:

Банк получателя:

ИНН

Получатель:

Назначение платежа:

Подпись:

Сумма	<input type="text"/>
Сч. №	<input type="text"/>
БИК	<input type="text"/>
Сч. №	<input type="text"/>
БИК	<input type="text"/>
Сч. №	<input type="text"/>
Сч. №	<input type="text"/>
Шифр док.	<input type="text"/>
Наим. плат.	<input type="text"/>
Код	<input type="text"/>
Срок плат.	<input type="text"/>
Очер. плат.	<input type="text"/>
Рез. поле	<input type="text"/>

Рисунок 2.9 Документ «Мемориальный ордер»

Поступ. в банк плат.		Списание со сч. плат.		0401060	
ПЛАТЕЖНОЕ ПОРУЧЕНИЕ №				Дата	
Сумма прописью				Вид платежа	
Сумма прописью		рублей		копеек	
ИНН		КПП		Сумма	
Платательщик				Сч. №	
Банк плательщика				БИК	
Банк получателя				Сч. №	
ИНН				Сч. №	
Получатель				Вид оп.	
				Срок плат.	
				Очер. плат.	
				Рез. поле	
Назначение платежа					
Подписи					
Отметки банка					
М.П.					

Рисунок 2.10 Документ «Платежное поручение»

<div style="display: flex; justify-content: space-between;"> Поступ. в банк плат. Списано со сч. плат. </div>		0401071				
ИНКАССОВОЕ ПОРУЧЕНИЕ № _____						
		Дата _____	Вид платежа _____			
Сумма прописью	_____ рубль _____ копеек					
ИНН _____	КПП _____	Сумма _____				
Платательщик _____		Сч. № _____				
		БИК _____				
Банк плательщика _____		Сч. № _____				
		БИК _____				
Банк получателя _____		Сч. № _____				
		Сч. № _____				
ИНН _____		КПП _____	Сч. № _____			
Получатель _____		Вид оп. _____	Очер. плат. _____			
		Наз. гп. _____	Рез. поле _____			
		Код _____				
Назначение платежа _____						
Подписи		Отметки банка получателя				
М.П. _____						

№ ч. плат.	№ плат. ордера	Дата плат. ордера	Сумма частного платежа	Сумма остатка платежа	Подпись	Дата помещения в картотеку
						Отметки банка плательщика

Рисунок 2.11 Документ «Инкассовое поручение»

2.3. Описание базы данных

База данных построена на основе реляционной модели. Структура базы данных включает следующие таблицы:

- Справочник «БИК» (BIK) содержит информацию о банках отправителя и получателя платежных документов.
- Мемориальный ордер (Memorial_order) содержит информацию о реквизитах документа «Мемориальный ордер».

- Платежное поручение (Platnoe_poruchenie) содержит информацию о реквизитах документа «Платежное поручение».
- Инкассовое поручение (Inkassovoe_poruchenie) содержит информацию о реквизитах документа «Инкассовое поручение».
- Справочник «Тип платежа» (Type_platega) содержит информацию о типах платежей (почтовый, электронный, телеграф).
- Справочник «Операторы» (Operators) содержит список операторов ПС.

Состав полей таблиц базы данных приведен в таблицах 2.1 – 2.6.

Справочник «БИК» (BIK)

Таблица 2.1

Обозначение поля	Наименование
Id_bik (PK)	Идентификатор справочника БИК
BIK	БИК
Type_KO	Тип КО
Name_bik	Наименование
N/P	н/п
Region	Населенный пункт

Документ «Мемориальный ордер» (Memorial_order)

Таблица 2.2

Обозначение поля	Наименование
Id_order (PK)	Идентификатор документа «Мемориальный ордер»
Number_order	Номер документа
Operator (FK)	Оператор
BIK (FK)	БИК банка
Date_to_bank	Дата поступления в банк платежа
Debet_schet	Дебет счета №
Kredit_schet	Кредит счета №
Name_platelchik	Наименование плательщика
Name_poluchatel	Наименование получателя
INN_platelchik	ИНН плательщика
INN_poluchatel	ИНН получателя
KPP_platelchik	КПП плательщика
KPP_poluchatel	КПП получателя
Summa	Сумма
Ocherednost	Очередность
Naznachenie_platega	Назначение платежа

Документ «Платежное поручение» (Platezhnoe_poruchenie)

Таблица 2.3

Обозначение поля	Наименование
Id_plat_poruch (PK)	Идентификатор документа «Платежное поручение»
Number_plat_poruch	Номер документа
Operator (FK)	Оператор
BIK (FK)	БИК
Type (FK)	Вид платежа
Date_to_bank	Дата поступления в банк платежа
Debet_schet	Дебет счета №
Kredit_schet	Кредит счета №
Name_platelchik	Наименование плательщика
Name_poluchatel	Наименование получателя
INN_platelchik	ИНН плательщика
INN_poluchatel	ИНН получателя
KPP_platelchik	КПП плательщика
KPP_poluchatel	КПП получателя
Summa	Сумма
Ocherednost	Очередность
Naznachenie_platega	Назначение платежа

Документ «Инкассовое поручение» (Inkassovoe_poruchenie)

Таблица 2.4

Обозначение поля	Наименование
Id_ink_poruch (PK)	Идентификатор документа «Инкассовое поручение»
Number_ink_poruch	Номер документа
Operator (FK)	Оператор
Akzept	Срок для акцепта
Type (FK)	Вид платежа
BIK (FK)	БИК
Date_to_bank	Дата поступления в банк платежа
Debet_schet	Дебет счета №
Kredit_schet	Кредит счета №
Name_platelchik	Наименование плательщика
Name_poluchatel	Наименование получателя
INN_platelchik	ИНН плательщика
INN_poluchatel	ИНН получателя
KPP_platelchik	КПП плательщика
KPP_poluchatel	КПП получателя
Summa	Сумма
Ocherednost	Очередность
Naznachenie_platega	Назначение платежа

Справочник «Тип платежа» (Type_platega)

Таблица 2.5

Обозначение поля	Наименование
Id_type (PK)	Идентификатор типа платежа
Name_plat	Наименование типа платежа

Справочник «Операторы» (Operators)

Таблица 2.6

Обозначение поля	Наименование
Id_operator (PK)	Идентификатор оператора
Name_operator	Наименование оператора
Prava	Права оператора

Структура базы данных представлена на рисунке 2.12.

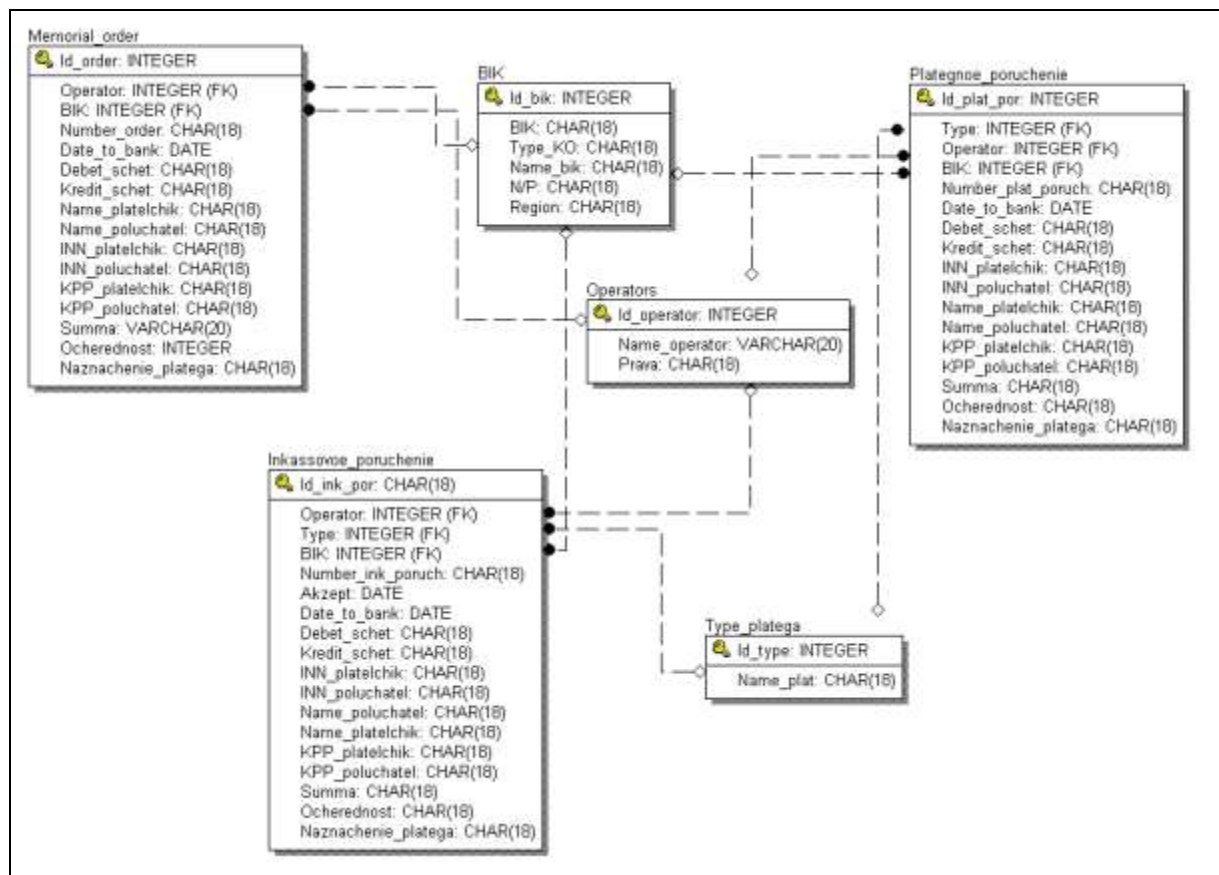


Рисунок 2.12 Логическая схема БД «Формирование платежных документов»

3. Определение уровня сложности выделенных программных объектов

3.1. Определение уровня сложности окон интерфейса пользователя

Для определения уровня функциональной сложности программного объекта «клиентское окно» и «диалоговое окно» необходимо подсчитать:

- 1) количество элементов интерфейса для ввода/редактирования/просмотра данных, расположенных в окне;
- 2) количество файлов данных (таблиц РСУБД) связанных с окном.

В разрабатываемом приложении будут созданы следующие окна:

- Основное клиентское окно приложения.
- Клиентское окно для работы с документом «Мемориальный ордер».
- Клиентское окно для работы с документом «Платежное поручение».
- Клиентское окно для работы с документом «Инкассовое поручение».
- Клиентское окно справочника БИК.
- Клиентское окно справочника операторов.
- Клиентское окно справочника видов платежей.
- Диалоговое окно для задания режима работы программы.

Основное клиентское окно (рис. 2.1) содержит только таблицу из 11 колонок. Соответственно, согласно Методике, данное окно содержит 11 элементов интерфейса для ввода/редактирования/просмотра данных. В основном окне выводятся данные из всех таблиц базы данных, кроме таблицы «Тип платежа» (Type_platega). Таким образом, с окном связано 5 таблиц базы данных. Аналогично анализируя все экранные формы приложения в соответствии с Методикой, получим таблицу:

Подсчет количественных характеристик окон приложения

Таблица 3.1

№	Тип окна	Название окна	Количество элементов интерфейса для ввода/редактирования/просмотра данных	Количество файлов данных (таблиц РСУБД) связанных с окном
1	Клиентское окно	Основное окно приложения	11	5
2	Клиентское окно	Окно для работы с документом «Мемориальный ордер»	17	2
3	Клиентское окно	Окно для работы с документом «Платежное поручение»	18	3

№	Тип окна	Название окна	Количество элементов интерфейса для ввода/редактирования/просмотра данных	Количество файлов данных (таблиц РСУБД) связанных с окном
4	Клиентское окно	Окно для работы с документом «Инкассовое поручение»	19	3
5	Клиентское окно	Окно справочника БИК.	5	1
6	Клиентское окно	Окно справочника операторов	2	1
7	Клиентское окно	Окно справочника видов платежей	2	1
8	Диалоговое окно	Диалог для задания режима работы программы	2	0

На основе количества элементов интерфейса для ввода/редактирования/просмотра данных и количества файлов данных (таблиц РСУБД) связанных с окном, по таблице 5.2 (для клиентских окон) и 5.3 (для диалоговых окон) Методики определяем уровень функциональной сложности для каждого окна.

Уровни сложности окон

Таблица 3.2

№	Тип окна	Название окна	Уровень сложности
1	Клиентское окно	Основное окно приложения	высокий
2	Клиентское окно	Окно для работы с документом «Мемориальный ордер»	высокий
3	Клиентское окно	Окно для работы с документом «Платежное поручение»	высокий
4	Клиентское окно	Окно для работы с документом «Инкассовое поручение»	высокий
5	Клиентское окно	Окно справочника БИК.	низкий
6	Клиентское окно	Окно справочника операторов	низкий
7	Клиентское окно	Окно справочника видов платежей	низкий
8	Диалоговое окно	Диалог для задания режима работы программы	низкий

3.2. Определение уровня сложности выходных документов

Для определения уровня функциональной сложности программного объекта «отчет» (выходной документ) необходимо подсчитать:

- 1) количество реквизитов документа;
- 2) количество файлов данных (таблиц РСУБД) связанных с выходным документом.

В процессе работы с ПС могут быть сформированы следующие документы:

- Мемориальный ордер.
- Платежное поручение.
- Инкассовое поручение.

Все выходные документы разрабатываемого ПС относятся к типу «Бланк», т.е. не содержат повторяющихся групп. Все подсчитываемые реквизиты документов на рис. 2.9 – 2.11 выделены цветом.

Для формирования реквизитов документов используются следующие таблицы БД:

Список таблиц БД, используемых при формировании выходных документов

Таблица 3.3

Документ	Используемые таблицы БД
Мемориальный ордер	«Мемориальный ордер» (Memorial_order)
	«БИК» (BIK)
Платежное поручение	«Платежное поручение» (Plategnoe_poruchenie)
	«БИК» (BIK)
	«Тип платежа» (Type_platega)
Инкассовое поручение	«Инкассовое поручение» (Inkassovoe_poruchenie)
	«БИК» (BIK)
	«Тип платежа» (Type_platega)

Проведя анализ всех выходных документов приложения в соответствии с Методикой, получим следующую таблицу распределения количественных свойств:

Подсчет количественных характеристик выходных документов

Таблица 3.4

№	Название документа	Количество реквизитов документа	Количество файлов данных (таблиц РСУБД) связанных с документом
1	Мемориальный ордер	22	2
2	Платежное поручение	34	3
3	Инкассовое поручение	35	3

На основе количества реквизитов и количества таблиц РСУБД, по таблице 5.4 Методики определяем уровень функциональной сложности для каждого выходного документа.

Уровни сложности выходных документов

Таблица 3.5

№	Название документа	Уровень сложности
1	Мемориальный ордер	высокий
2	Платежное поручение	высокий
3	Инкассовое поручение	высокий

3.3. Определение уровня сложности таблиц БД

Для определения уровня сложности программного объекта «таблица БД» необходимо подсчитать количество элементов данных, которые он содержит. Каждой таблице БД соответствует столько элементов данных, сколько полей она содержит. При этом также учитываются все ключевые поля.

Все таблицы БД используются только разрабатываемым приложением и поэтому являются внутренними. Проведя анализ структуры БД приложения в соответствии с Методикой, получим следующую таблицу распределения количественных свойств:

Подсчет количественных характеристик таблиц БД

Таблица 3.6

№	Тип	Наименование таблицы	количество элементов данных
1	внутренний	«Мемориальный ордер» (Memorial_order)	16
2	внутренний	«Платежное поручение» (Plategnoe_poruchenie)	17
3	внутренний	«Инкассовое поручение» (Inkassovoe_poruchenie)	18
4	внутренний	«БИК» (BIK)	6
5	внутренний	«Тип платежа» (Type_platega)	2
6	внутренний	«Операторы» (Operators)	3

На основе количества элементов данных и типа, по таблице 5.5 Методики определяем уровень функциональной сложности для каждой таблицы БД.

Уровни сложности таблиц БД

Таблица 3.7

№	Наименование таблицы	Уровень сложности
1	«Мемориальный ордер» (Memorial_order)	средний
2	«Платежное поручение» (Plategnoe_poruchenie)	средний
3	«Инкассовое поручение» (Inkassovoe_poruchenie)	средний
4	«БИК» (BIK)	средний
5	«Тип платежа» (Type_platega)	средний
6	«Операторы» (Operators)	средний

4. Получение частных оценок размеров программных объектов разрабатываемого ПС

Для всех выделенных программных объектов на основе их уровня сложности по таблице 5.7 Методики подсчитывается оценка их размера в функциональных точках (таблица 4.1).

Количество функциональных точек, соответствующих выделенным объектам ПС

Таблица 4.1

№	Тип объекта	Наименование	Количество функциональных баллов
1	Клиентское окно	Основное окно приложения	13
2	Клиентское окно	Окно для работы с документом «Мемориальный ордер»	13
3	Клиентское окно	Окно для работы с документом «Платежное поручение»	13
4	Клиентское окно	Окно для работы с документом «Инкассовое поручение»	13
5	Клиентское окно	Окно справочника БИК.	4
6	Клиентское окно	Окно справочника операторов	4
7	Клиентское окно	Окно справочника видов платежей	4
8	Диалоговое окно	Диалог для задания режима работы программы	3
9	Вых. документ	Мемориальный ордер	7
10	Вых. документ	Платежное поручение	7
11	Вых. документ	Инкассовое поручение	7
12	Внутренняя таблица БД	«Мемориальный ордер» (Memorial_order)	7
13	Внутренняя таблица БД	«Платежное поручение» (Plategnoe_poruchenie)	7
14	Внутренняя таблица БД	«Инкассовое поручение» (Inkassovoe_poruchenie)	7
15	Внутренняя таблица БД	«БИК» (BIK)	7
16	Внутренняя таблица БД	«Тип платежа» (Type_platega)	7
17	Внутренняя таблица БД	«Операторы» (Operators)	7

5. Получение итоговой оценки размера ПС

Нескорректированный размер ПС ($V_{\phi m}$) в функциональных точках определяется как сумма количества функциональных точек, соответствующих всем программным объектам.
 $V_{\phi m} = 130$.

Определяем размер ПС в строках исходного кода (V):

$$V = V_{\phi m} * 125 = 130 * 125 = 16250.$$

Ниже представлен отчет по расчету трудоемкости.

РАСЧЕТ ТРУДОЕМКОСТИ С ИСПОЛЬЗОВАНИЕМ ПО АРМ

Расчет трудоемкости разработки ПС (метод подсчета функциональных баллов по элементам графического интерфейса пользователя)

Параметры расчета

Параметр	Значение
Имя расчета	ПС для оперативной подготовки платежных поручений, мемориальных ордеров и инкассовых документов
Тип расчета	Расчет трудоемкости разработки ПС (метод подсчета функциональных баллов по элементам графического интерфейса пользователя)
Версия модели	2.5.1.0
Фонд времени разработчика [дней/мес.]	21
Тип ограничения	По количеству персонала
Тип структуры программного средства	ПС с набором библиотек (=1.00)
Технологии взаимодействия модулей	Объектно-ориентированные технологии (COM/DCOM, ODBC, OLEDB, ADO, ActiveX) (=0.35)

Коэффициенты уровня расчета

Коэффициент	Уровень значения
Степень новизны ПС	ПС, являющееся развитием определенного параметрического ряда ПС на прежнем типе ТС и новом типе ОС (=0.94)
Требования к надежности	Средний, возмолнимый ущерб (=1.00)
Требования к производительности	Требования к производительности ПС не установлены (однако производительность ПС должна обеспечивать приемлемое время отклика при работе пользователя в интерактивном режиме) (=1.00)
Информативность документации	Соответствует потребностям жизненного цикла (=1.00)
Опыт разработки ПС подобного типа	Более 5 разработанных ПС (компонентов АС) (=0.81)
Влияние сроков работ на трудоемкость	Равно или более 100 (=1.00)

Коэффициенты уровня подпроцесса

Подпроцесс	Коэффициент	Уровень значения
Анализ требований	Уровень квалификации аналитиков	Высокий (=0.87)
Анализ требований	Опыт работы аналитиков в данной предметной области	5 разработанных ПС (компонентов АС) (=0.87)
Проектирование	Уровень квалификации проектировщиков	Высокий (=0.92)
Проектирование	Опыт работы проектировщиков с используемыми средствами проектирования	5 разработанных ПС (компонентов АС) (=0.92)

Проектирование	Используемые средства проектирования	Средства проектирования БД (BPWIN, ERWIN/ERX, Designer 2000, DataDirect Explorer, Oracle Designer, Silverrun RBM и BPM, Together J) (=1.03)
Программирование	Уровень квалификации программистов	Высокий (=0.95)
Программирование	Используемая среда разработки	Среды разработки на основе СУБД типа Oracle (Developer/FORMS), Informix, MS SQLServer, Paradox, dbase, FoxPro (Clarion) (=0.92)
Тестирование	Уровень квалификации персонала, осуществляющего тестирование	Высокий (=0.89)
Тестирование	Используемые средства тестирования	Автоматизированные средства тестирования использовались (=0.76)
Тестирование	Размер БД	D/P (=0.95)
Ввод в действие ПС	Уровень квалификации персонала, осуществляющего ввод в действие ПС	Высокий (=0.94)

Дерево элементов

Элемент	Описание	Тип	Значение свойства 1	Значение свойства 2	Объем кода на условном языке [строк]	Объем [ФТ]
Основное окно приложения		Клиентское окно	6 - 15 (Средний)	более 3 (Высокий)	1625	13
Окно для работы с документом "Мемориальный ордер"		Клиентское окно	более 15 (Высокий)	2 - 3 (Средний)	1625	13
Окно для работы с документом "Платежное поручение"		Клиентское окно	более 15 (Высокий)	2 - 3 (Средний)	1625	13
Окно для работы с документом "Инкассовое поручение"		Клиентское окно	более 15 (Высокий)	2 - 3 (Средний)	1625	13
Окно справочника БИК		Клиентское окно	не более 5 (Низкий)	не более 1 (Низкий)	500	4
Окно справочника операторов		Клиентское окно	не более 5 (Низкий)	не более 1 (Низкий)	500	4
Окно справочника видов платежей		Клиентское окно	не более 5 (Низкий)	не более 1 (Низкий)	500	4
Диалог для задания режима работы программы		Диалоговое окно	не более 5 (Низкий)	не более 1 (Низкий)	375	3
Мемориальный ордер		Отчет	более 19 (Высокий)	2 - 3 (Средний)	875	7
Платежное поручение		Отчет	более 19 (Высокий)	2 - 3 (Средний)	875	7
Инкассовое поручение		Отчет	более 19 (Высокий)	2 - 3 (Средний)	875	7
"Мемориальный ордер" (Memorial_order)		Внутренний файл данных (таблица РСУБД)	не более 50 (Средний)		875	7
"Платежное поручение" (Plategnoe_poruchenie)		Внутренний файл данных (таблица РСУБД)	не более 50 (Средний)		875	7
"Инкассовое поручение" (Inkassovoe_poruchenie)		Внутренний файл данных (таблица РСУБД)	не более 50 (Средний)		875	7
"БИК" (BIK)		Внутренний файл данных (таблица РСУБД)	не более 50 (Средний)		875	7

"Тип платежа" (Type_platega)		Внутренний файл данных (таблица РСУБД)	не более 50 (Средний)		875	7
"Операторы" (Operators)		Внутренний файл данных (таблица РСУБД)	не более 50 (Средний)		875	7
Общий объем ПС:						

Расчет трудоемкости подпроцессов

Подпроцесс	Трудоемкость [чел.дн.]	Кол-во разработчиков [чел]	Продолжительность [мес]/[дн]
Анализ требований	1.93	1.00	0.09/1.89
Проектирование	26.69	1.00	1.27/26.67
Программирование	176.14	2.00	4.19/87.99
Тестирование	11.48	1.00	0.55/11.55
Ввод в действие ПС	2.40	1.00	0.11/2.31
Всего	218.64		6.21/130.41
Базовая трудоемкость	255.11		

ПРИЛОЖЕНИЕ 7**ПРИМЕР РАСЧЕТА ТРУДОЕМКОСТИ СОПРОВОЖДЕНИЯ ПС**

Выполняется сопровождение программного средства мониторинга финансовых операций, выполняемых через банкомат. В процессе сопровождения требуется доработать форму выдаваемого ПС отчета, а также устранить некоторые ошибки, допущенные при программировании и приводящие к невозможности полноценно использовать некоторые функции программного средства. В соответствии с изменениями в ПС требуется также доработать документацию. При сопровождении осуществляется консультация пользователей ПС.

В процессе сопровождения таким образом требуется выполнение следующих работ:

- подготовка процесса;
- анализ проблем и изменений;
- внесение изменений в программное средство (предполагается изменение 15-ти% программного средства);
- внесение изменений в документацию (предполагается изменение 10-ти% документации сопровождения);
- проверка и приемка при сопровождении;
- формирование и ведение ФАП;
- поддержка пользователя.

Ниже представлен отчет по расчету трудоемкости. Расчет выполнен с учетом общей продолжительности работ 1 год.

РАСЧЕТ ТРУДОЕМКОСТИ С ИСПОЛЬЗОВАНИЕМ ПО АРМ

Расчет трудоемкости сопровождения ПС

Параметры расчета

Параметр	Значение
Имя расчета	Программное средство мониторинга финансовых операций, выполняемых через банкомат
Тип расчета	Расчет трудоемкости сопровождения ПС
Версия модели	2.4.1.0
Фонд времени разработчика [дней/мес.]	21
Тип ограничения	По продолжительности этапа [мес.]
Объем кода [строк]	21125
Объем документации сопровождения [строк]	5500
Объем доработок кода [строк кода]	3168
Объем доработок документации [строк]	550
Количество установленных экземпляров ПО	10
Число пользователей	20

Коэффициенты уровня работы

Работа	Коэффициент	Уровень значения
Подготовка процесса	Механизм обновления	Автоматическое обновление (=0.81)
Подготовка процесса	Наличие аналогов	Нет (=1.58)
Подготовка процесса	Степень участия службы сопровождения в разработке ПС	ССо разрабатывала ПС или значительную его часть (=0.49)
Подготовка процесса	Количество модификаций ПС за год	от 10 до 50 (=1.01)
Анализ проблем и изменений	Механизм обновления	Автоматическое обновление (=0.76)
Анализ проблем и изменений	Степень участия службы сопровождения в разработке ПС	ССо разрабатывала ПС или значительную его часть (=0.75)
Анализ проблем и изменений	Характер внедрения ПС	Внедрение ПС в составе комплекса взаимосвязанных ПС (=1.68)

Анализ проблем и изменений	Количество модификаций ПС за год	от 10 до 50 (=1.068)
Анализ проблем и изменений	Периодичность использования ПС	Ежедневно (=1.00)
Внесение изменений в программное средство	Степень участия службы сопровождения в разработке ПС	ССо разрабатывала ПС или значительную его часть (=0.49)
Внесение изменений в программное средство	Сложность ПС	Высокий (=1.35)
Внесение изменений в программное средство	Периодичность использования ПС	Ежедневно (=1.00)
Внесение изменений в документацию	Механизм обновления	Автоматическое обновление (=0.49)
Внесение изменений в документацию	Сложность ПС	Высокий (=1.15)
Внесение изменений в документацию	Периодичность использования ПС	Ежедневно (=1.00)
Проверка и приемка при сопровождении	Сложность ПС	Высокий (=1.09)
Проверка и приемка при сопровождении	Характер внедрения ПС	Внедрение ПС в составе комплекса взаимосвязанных ПС (=1.10)
Проверка и приемка при сопровождении	Полнота тестирования ПС	Тестирование всех режимов (=1.23)
Проверка и приемка при сопровождении	Степень участия службы сопровождения в разработке ПС	ССо разрабатывала ПС или значительную его часть (=0.84)
Проверка и приемка при сопровождении	Количество модификаций ПС за год	от 10 до 50 (=1.07)
Формирование и ведение ФАП	Количество модификаций ПС за год	от 10 до 50 (=1.07)
Поддержка пользователя	Количество модификаций ПС за год	от 10 до 50 (=1.05)
Поддержка пользователя	Периодичность использования ПС	Ежедневно (=1.00)

Используемые средства разработки

Средство разработки	Доля использования
C++ (=1.00)	1.00

Расчет трудоемкости работ

Работа	Трудоемкость [чел.дн.]	Кол-во разработчиков [чел]	Продолжительность [мес]/[дн]
Подготовка процесса	9.76	0.46	1.00/21.00
Анализ проблем и изменений	10.49	0.25	2.00/42.00
Внесение изменений в программное средство	77.69	0.92	4.00/84.00
Внесение изменений в документацию	6.53	0.31	1.00/21.00
Проверка и приемка при сопровождении	22.71	1.08	1.00/21.00
Формирование и ведение ФАП	12.17	0.58	1.00/21.00
Поддержка пользователя	13.02	0.31	2.00/42.00
Всего	152.37		12.00/252.00

ПРИЛОЖЕНИЕ 8**ПРИМЕР РАСЧЕТА ТРУДОЕМКОСТИ ЭКСПЛУАТАЦИИ ПС**

В процессе эксплуатации программного средства мониторинга финансовых операций проводятся следующие работы:

- подготовка процесса;
- эксплуатационные испытания;
- эксплуатация ПС (поддержка функционирования ПС в эксплуатационной среде);
- формирование и ведение ФАП;
- поддержка пользователя;
- снятие с эксплуатации.

Ниже представлен отчет по расчету трудоемкости. Расчет выполнен с учетом общей продолжительности работ 1 год.

РАСЧЕТ ТРУДОЕМКОСТИ С ИСПОЛЬЗОВАНИЕМ ПО АРМ

Расчет трудоемкости эксплуатации ПС

Параметры расчета

Параметр	Значение
Имя расчета	Программное средство мониторинга финансовых операций, выполняемых через банкомат
Тип расчета	Расчет трудоемкости эксплуатации ПС
Версия модели	2.5.1.0
Фонд времени разработчика [дней/мес.]	21
Тип ограничения	По продолжительности этапа [мес.]
Объем документации эксплуатации [строк]	3600
Количество установленных экземпляров ПО	10
Число пользователей	20

Коэффициенты уровня работы

Работа	Коэффициент	Уровень значения
Подготовка процесса	Механизм обновления	Автоматическое обновление (=0.67)
Подготовка процесса	Наличие аналогов	Нет (=1.20)
Подготовка процесса	Сложность ПС	Высокий (=1.07)
Подготовка процесса	Количество модификаций ПС за год	от 10 до 50 (=1.18)
Подготовка процесса	Тип используемой БД	MS SQL Server, Informix, Oracle (=1.07)
Эксплуатационные испытания	Механизм обновления	Автоматическое обновление (=0.82)
Эксплуатационные испытания	Полнота тестирования ПС	Тестирование всех режимов (=1.17)
Эксплуатационные испытания	Сложность ПС	Высокий (=1.03)
Эксплуатационные испытания	Количество модификаций ПС за год	от 10 до 50 (=1.19)
Эксплуатационные испытания	Тип используемой БД	MS SQL Server, Informix, Oracle (=1.05)
Эксплуатация ПС	Количество модификаций ПС за год	от 10 до 50 (=1.264)

Эксплуатация ПС	Периодичность использования ПС	Ежедневно (=1.00)
Формирование и ведение ФАП	Количество модификаций ПС за год	от 10 до 50 (=1.07)
Поддержка пользователя	Количество модификаций ПС за год	от 10 до 50 (=1.05)
Поддержка пользователя	Периодичность использования ПС	Ежедневно (=1.00)
Снятие с эксплуатации	Характер поставки	Поставка стандартного комплекта ПС в составе комплекса взаимосвязанных ПС (=1.80)

Расчет трудоемкости работ

Работа	Трудоемкость [чел.дн.]	Кол-во разработчиков [чел]	Продолжительность [мес]/[дн]
Подготовка процесса	9.16	0.44	1.00/21.00
Эксплуатационные испытания	17.29	0.82	1.00/21.00
Эксплуатация ПС	19.71	0.16	6.00/126.00
Формирование и ведение ФАП	11.32	0.54	1.00/21.00
Поддержка пользователя	11.62	0.28	2.00/42.00
Снятие с эксплуатации	10.63	0.51	1.00/21.00
Всего	79.72		12.00/252.00

Форма отчета расчета по оценке суммарной трудоемкости (с использованием расчетов по разработке, сопровождению и эксплуатации «Программное средство мониторинга финансовых операций, выполняемых через банкомат»), формируемая ПО АРМ представлена ниже:

Оценка суммарной трудоемкости

Параметры расчета

Параметр	Значение
Имя расчета	Программное средство мониторинга финансовых операций, выполняемых через банкомат
Тип расчета	Оценка суммарной трудоемкости
Общие трудозатраты [чел.дней]	825.78

Тип расчетов: Расчет трудоемкости разработки ПС (Каталог)

	Версия модели	Анализ требований	Проектирование	Программирование	Тестирование	Ввод в действие ПС	Всего
Программное средство мониторинга финансовых операций, выполняемых через банкомат	2.5.1.0	4.89	67.55	484.60	30.57	6.07	593.69
Всего		4.89	67.55	484.60	30.57	6.07	593.69

Тип расчетов: Расчет трудоемкости сопровождения ПС

	Версия модели	Подготовка процесса	Анализ проблем и изменений	Внесение изменений в программное средство	Внесение изменений в документацию	Проверка и приемка при сопровождении	Формирование и ведение ФАП	Поддержка пользователя	Всего
Программное средство мониторинга финансовых операций, выполняемых через банкомат	2.5.1.0	9.76	10.49	77.69	6.53	22.71	12.17	13.02	152.37
Всего		9.76	10.49	77.69	6.53	22.71	12.17	13.02	152.37

Тип расчетов: Расчет трудоемкости эксплуатации ПС

	Версия модели	Подготовка процесса	Эксплуатационные испытания	Эксплуатация ПС	Формирование и ведение ФАП	Поддержка пользователя	Снятие с эксплуатации	Всего
Программное средство мониторинга финансовых операций, выполняемых через банкомат	2.4.1.0	9.16	17.29	19.71	11.32	11.62	10.63	79.72
Всего		9.16	17.29	19.71	11.32	11.62	10.63	79.72

ПРИЛОЖЕНИЕ 9

ШКАЛЫ ДЛЯ ОЦЕНКИ ТОЧНОСТИ ВЫЧИСЛЕНИЙ

Шкалы для оценки точности расчета трудоемкости процесса разработки ПС

Таблица 7.1

Подпроцесс	Диапазон величины трудоемкости*	Зеленая зона	Желтая зона	Красная зона
Анализ требований к ПС	$T \leq 10$	$\delta \leq 0,2311$	$0,2311 < \delta \leq 0,3821$	$\delta > 0,3821$
Анализ требований к ПС	$T > 10$	$\delta \leq 0,20$	$0,20 < \delta \leq 0,27$	$\delta > 0,27$
Проектирование	$T \leq 10$	$\delta \leq 0,7200$	$0,7200 < \delta \leq 0,7290$	$\delta > 0,7290$
Проектирование	$10 < T \leq 100$	$\delta \leq 0,39$	$0,39 < \delta \leq 0,44$	$\delta > 0,44$
Проектирование	$T > 100$	$\delta \leq 0,40$	$0,40 < \delta \leq 0,42$	$\delta > 0,42$
Программирование	$T \leq 10$	$\delta \leq 0,06$	$0,06 < \delta \leq 0,0610$	$\delta > 0,0610$
Программирование	$10 < T \leq 100$	$\delta \leq 0,09$	$0,09 < \delta \leq 0,17$	$\delta > 0,17$
Программирование	$T > 100$	$\delta \leq 0,24$	$0,24 < \delta \leq 0,2410$	$\delta > 0,2410$
Тестирование ПС	$T \leq 10$	$\delta \leq 0,16$	$0,16 < \delta \leq 0,1610$	$\delta > 0,1610$
Тестирование ПС	$10 < T \leq 100$	$\delta \leq 0,45$	$0,45 < \delta \leq 0,4510$	$\delta > 0,4510$
Тестирование ПС	$T > 100$	$\delta \leq 0,3970$	$0,3970 < \delta \leq 0,3980$	$\delta > 0,3980$
Ввод в действие ПС	$T \leq 10$	$\delta \leq 0,48$	$0,48 < \delta \leq 0,4836$	$\delta > 0,4836$
Ввод в действие ПС	$T > 10$	$\delta \leq 0,08$	$0,08 < \delta \leq 0,18$	$\delta > 0,18$

*Примечание: в таблицах 7.1–7.3 в данном столбце имеется ввиду планируемая трудоемкость в чел. - днях

Шкалы для оценки точности расчета трудоемкости процесса сопровождения ПС

Таблица 7.2

Работа	Диапазон величины трудоемкости	Зеленая зона	Желтая зона	Красная зона
Подготовка процесса	$T \leq 10$	$\delta \leq 0,20$	$0,20 < \delta \leq 0,2140$	$\delta > 0,2140$
Подготовка процесса	$10 < T \leq 100$	$\delta \leq 0,28$	$0,28 < \delta \leq 0,31$	$\delta > 0,31$
Подготовка процесса	$T > 100$	$\delta \leq 0,52$	$0,52 < \delta \leq 0,63$	$\delta > 0,63$
Анализ проблем и изменений	$T \leq 10$	$\delta \leq 0,35$	$0,35 < \delta \leq 0,38$	$\delta > 0,38$
Анализ проблем и изменений	$10 < T \leq 100$	$\delta \leq 0,49$	$0,49 < \delta \leq 0,50$	$\delta > 0,50$
Анализ проблем и изменений	$T > 100$	$\delta \leq 0,73$	$0,73 < \delta \leq 0,7310$	$\delta > 0,7310$
Внесение изменений	$T \leq 10$	$\delta \leq 1,70$	$1,70 < \delta \leq 1,7010$	$\delta > 1,7010$
Внесение изменений	$10 < T \leq 100$	$\delta \leq 3,90$	$3,90 < \delta \leq 5,24$	$\delta > 5,24$
Внесение изменений	$T > 100$	$\delta \leq 0,44$	$0,44 < \delta \leq 0,46$	$\delta > 0,46$
Проверка и приемка при сопровождении	$T \leq 10$	$\delta \leq 0,58$	$0,58 < \delta \leq 0,6270$	$\delta > 0,6270$

Работа	Диапазон величины трудоемкости	Зеленая зона	Желтая зона	Красная зона
Проверка и приемка при сопровождении	$10 < T \leq 100$	$\delta \leq 0,39$	$0,39 < \delta \leq 0,40$	$\delta > 0,40$
Проверка и приемка при сопровождении	$T > 100$	$\delta \leq 0,19$	$0,19 < \delta \leq 0,20$	$\delta > 0,20$
Перенос	$T \leq 10$	$\delta \leq 0,36$	$0,36 < \delta \leq 0,70$	$\delta > 0,70$
Перенос	$10 < T \leq 100$	$\delta \leq 0,16$	$0,16 < \delta \leq 0,21$	$\delta > 0,21$
Перенос	$T > 100$	$\delta \leq 0,37$	$0,37 < \delta \leq 0,39$	$\delta > 0,39$
Снятие с эксплуатации	$T \leq 10$	$\delta \leq 0,54$	$0,54 < \delta \leq 0,55$	$\delta > 0,55$
Снятие с эксплуатации	$T > 10$	$\delta \leq 0,41$	$0,41 < \delta \leq 0,52$	$\delta > 0,52$
Формирование и ведение фонда алгоритмов и программ (ФАП)	$T \leq 10$	$\delta \leq 0,1874$	$0,1874 < \delta \leq 0,2024$	$\delta > 0,2024$
Формирование и ведение фонда алгоритмов и программ (ФАП)	$T > 10$	$\delta \leq 0,0686$	$0,0686 < \delta \leq 0,2082$	$\delta > 0,2082$
Поддержка пользователя	$T \leq 10$	$\delta \leq 0,20$	$0,20 < \delta \leq 0,24$	$\delta > 0,24$
Поддержка пользователя	$T > 10$	$\delta \leq 0,5070$	$0,5070 < \delta \leq 0,5090$	$\delta > 0,5090$

ПРИЛОЖЕНИЕ 10**ВЫДЕЛЕНИЕ КОМПОНЕНТОВ АВТОМАТИЗИРОВАННОЙ СИСТЕМЫ**

Допускается расчет трудоемкости для компонентов автоматизированной системы в соответствии с Методикой оценки трудоемкости при условии:

– если компонент АС соответствует определению программного средства по ГОСТ 28806-90 (программное средство – объект, состоящий из программ, процедур, правил, а также, если предусмотрено, сопутствующих им документации и данных, относящихся к функционированию системы обработки информации);

– если компонент АС разрабатывался, тестировался и внедрялся независимо от той автоматизированной системы, для которой он предназначен, то этот компонент допускается считать программным средством и рассчитывать трудоемкость для него в соответствии с Методикой.

Если автоматизированная система разрабатывалась, тестировалась и внедрялась как одно целое, то для оценки трудоемкости ее отдельного компонента первоначально необходимо представить АС в виде отдельных компонентов, т.е. провести декомпозицию. Декомпозицию рекомендуется выполнять в соответствии со следующим алгоритмом:

- 1) Согласно техническому заданию на разработку АС выделяются группы задач, решаемых АС. Каждая группа задач включает в себя отдельные родственные задачи АС, которые можно объединить по одному или нескольким общим признакам.
- 2) В соответствии с техническим заданием на разработку АС из Каталога функций Методики выбираются те функции, которые реализованы для достижения каждой группы задач (п.1) рассматриваемой АС. Причем объем функций выбирается таким образом, чтобы суммарный объем, подсчитанный согласно пунктам 3.2.1 – 3.2.5 Методики, совпал с реальным объемом исходного кода АС.
- 3) Функции АС группируются таким образом, чтобы каждый образуемый компонент соответствовал одной группе задач (п.1) рассматриваемой АС. Допускается, что отдельные функции могут реализовываться в нескольких компонентах системы, что должно быть учтено коэффициентом повторного использования и числом реализации соответствующих функций. Общий объем в строках кода должен остаться неизменным. Следует помнить, что

ограничения на использование отдельных функций из Каталога также остаются.

- 4) Каждый выделенный таким образом компонент условно считается программным средством (объем такого условного ПС приблизительно находится в диапазоне величин объемов ПС, использованных при статистической обработке для построения моделей оценки, поэтому результат с определенной ошибкой можно объяснить моделью расчета, применяемой в Методике). Такая условность имеет ряд важных следствий, которые необходимо иметь ввиду при использовании описываемого подхода:

- величина трудоемкости, полученная в соответствии с Методикой оценки трудоемкости, будет являться примерной;

- величина ошибки оценки трудоемкости будет тем выше, чем больше функциональные возможности оцениваемого компонента используются другими компонентами данной АС, т.е. чем больше данный компонент интегрирован в АС;

- величина ошибки при расчете трудоемкости для нескольких компонентов АС или всей АС путем суммирования значений трудоемкости также будет суммироваться и возрастать пропорционально числу суммируемых компонентов, поэтому не рекомендуется оценивать трудоемкость по нескольким компонентам АС и не допускается оценивать трудоемкость всей АС как сумму трудоемкостей по ее отдельным компонентам.

Рассмотрим следующий пример. Имеется автоматизированная система документооборота (33500 строк кода), которая, согласно ТЗ на разработку, должна обеспечить возможность:

- автоматизированного ввода и обработки информации (проверка достоверности данных, выполнение проводок) по платежным документам, поступающей на вход системы и накапливающейся в базе данных;

- криптографической защиты информации с помощью электронной подписи документов;

- формирования отчетных форм и предоставление информации о выполненных операциях по поступившим платежным документам.

Таким образом, основные группы задач, решаемые системой, включают в себя:

- автоматизированный ввод, обработку и сохранение информации (задачи автоматизированной работы с данными);
- криптозащиту (реализация задач по безопасности обмена данными);
- формирование отчетов (задача предоставления информации оператору АС).

Функции АС имеют средний уровень сложности ($K_{\text{сложн}} = 1,00$), реализуются на языке C++ ($K_{\text{ср.разр}} = 1,00$), опыт работы программистов со средством разработки также средний ($K_{\text{опыт}} = 1,00$). Выбираем из Каталога необходимые функции АС (для примера устанавливается примерный набор функций; для подобной реальной системы набор функций скорее всего будет более полным):

- 1.1.5 «Реализация базовых функций информационной безопасности в соответствии с нормативными документами Кредитных организаций», 8000 строк кода;
- 1.2.2 «Взаимодействие с транспортной (почтовой) системой», 2000 строк кода;
- 1.5.1 «Формирование физической структуры аналитических БД (на одну БД)», 4000 строк кода (одна база данных);
- 1.5.5 «Ведение журнала операций с аналитическими БД», 1000 строк кода;
- 1.5.12 «Поиск, предоставление и вывод информации», 3400 строк кода;
- 2.1.3 «Ввод и первичный контроль документов из файлов», 3000 строк кода;
- 2.1.4 «Логический, синтаксический и номенклатурный контроль данных», 3000 строк кода;
- 3.2.1 «Регистрация входных документов», 1000 строк кода;
- 4.6.1 «Формирование отчетов», 5 отчетов, $k_i = 0.6$, $200 \times 5 \times 0,6 = 600$ строк кода;
- 4.6.2 «Просмотр и редактирование отчетных форм», 5 отчетов, $k_i = 0.7$, $1000 \times 5 \times 0.7 = 3500$ строк кода;
- 5.2.1 «Реализация интерфейса пользователя», 4000 строк кода.

Общий объем, рассчитанный в соответствии с пп. 3.2.1 – 3.2.5 Методики, составляет 33500 строк кода, что соответствует объему АС.

Выделим компоненты согласно группам задач и определим назначение каждой функции компоненту:

- 1) Компонент «Автоматизированный ввод, обработка и сохранение информации», состоящий из функций:
 - 1.2.2 «Взаимодействие с транспортной (почтовой) системой»;

- 1.5.1 «Формирование физической структуры аналитических БД (*на одну БД*)»;
 - 1.5.5 «Ведение журнала операций с аналитическими БД»;
 - 1.5.12 «Поиск, предоставление и вывод информации»;
 - 2.1.3 «Ввод и первичный контроль документов из файлов»;
 - 2.1.4 «Логический, синтаксический и номенклатурный контроль данных»;
 - 3.2.1 «Регистрация входных документов».
- 2) Компонент «Криптозащита», состоящий из функций:
- 1.1.5 «Реализация базовых функций информационной безопасности в соответствии с нормативными документами Кредитных организаций»;
- 3) Компонент «Формирование отчетов», состоящий из функций:
- 4.6.1 «Формирование отчетов»;
 - 4.6.2 «Просмотр и редактирование отчетных форм»;
 - 5.2.1 «Реализация интерфейса пользователя».

Согласно рекомендуемому подходу к декомпозиции автоматизированной системы условно получаем три программных средства, расчет трудоемкости для которых можем выполнять по Методике (с учетом допущений, приведенных на шаге 4 рассматриваемого алгоритма):

- «Автоматизированный ввод, обработка и сохранение информации»;
- «Криптозащита»;
- «Формирование отчетов».

Необходимо учитывать следующее: декомпозиция для одной АС выполняется один раз, и результаты декомпозиции могут использоваться неоднократно, например, если возникает потребность в доработке других компонентов АС. В этом случае не требуется заново разбивать АС на составляющие.