

Отчёт по курсу  
«Верификация ПО»

Выполнил:

студент группы 22608 О. Б. Петрина

Преподаватель:

к.ф-м.н., доцент К. А. Кулаков

Петрозаводск

2014

# СОДЕРЖАНИЕ

<u>1</u>	<u>Объект исследования.....</u>	<u>3</u>
<u>2</u>	<u>План тестирования.....</u>	<u>4</u>
<u>2.1</u>	<u>Описание проведения тестирования.....</u>	<u>7</u>
<u>2.1.1</u>	<u>Блочное тестирование.....</u>	<u>7</u>
<u>2.1.2</u>	<u>Интеграционное тестирование.....</u>	<u>8</u>
<u>2.1.3</u>	<u>Аттестационное тестирование.....</u>	<u>9</u>
<u>2.2</u>	<u>Описание тестируемых процедур.....</u>	<u>10</u>
<u>2.3</u>	<u>Критерии тестирования.....</u>	<u>12</u>
<u>3</u>	<u>Тесты.....</u>	<u>13</u>
<u>3.1</u>	<u>Блочные тесты.....</u>	<u>13</u>
<u>3.2</u>	<u>Интеграционные тесты.....</u>	<u>19</u>
<u>Покрытие исполняемого кода тестами.....</u>		<u>21</u>
<u>3.3</u>	<u>Аттестационные тесты.....</u>	<u>21</u>
<u>4</u>	<u>Примеры тестов.....</u>	<u>24</u>
<u>5</u>	<u>Результаты тестирования.....</u>	<u>25</u>

## **1 Объект исследования**

"1С:Деньги 8" – программа для управления личными и семейными финансами.

Программа позволяет вести учет любых операций с деньгами, включая доходы и расходы, перемещения денег, обмен валют, расчеты по долгам и любые другие действия с наличными, безналичными, электронными деньгами, долгами и имуществом.

Для тестирования выбрана демонстрационная база 1С: Деньги 8, редакция 2.0 (2.0.15.9) (<http://v8.1c.ru/money>). Тестирование проводилось на платформе 1С:Предприятие 8.3 (8.3.4.465).

В качестве объекта тестирования выбран документ УниверсальнаяОперация, позволяющий провести основные денежные операции.

Каждый документ содержит форму и модуль. Модуль документа обращается к общим модулям для выполнения типовых операций. Для заполнения документов информация берется из справочников (перечислений, констант), а также вводится пользователем вручную. После проведения документа могут быть добавлены записи в регистры.

Для получения данных при составлении отчета производится запрос к справочникам и нужным регистрам.

Заполнение отчетов не будет тестироваться, поскольку механизм создания отчетов предполагает заполнение графической Схемы компоновки данных, с помощью которой автоматически формируется код и отчет. Инструментов для тестирования Схемы компоновки данных не было найдено.

Для тестирования документа использовался набор инструментов для выполнения тестирования в 1С:Предприятие 8, доступных по адресу <https://github.com/xDrivenDevelopment/xUnitFor1C>

## 2 План тестирования

В общем виде структура система выглядит как показано на рисунке 1. Для тестирования был выбран модуль документа УниверсальнаяОперация. Документ УниверсальнаяОперация позволяет провести основные денежные операции: доход, расход и перемещения денег между кошельками.

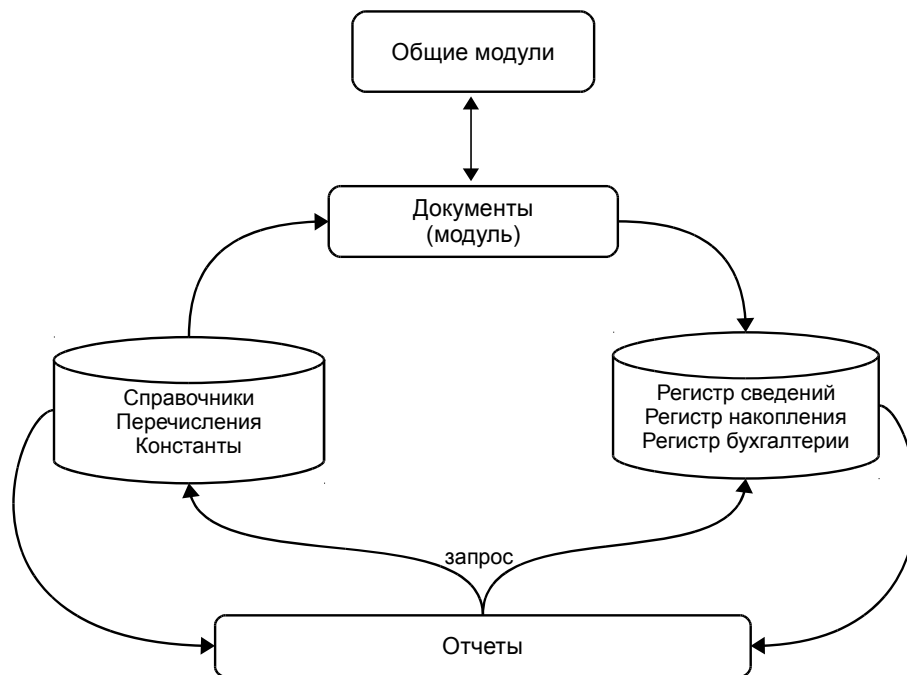


Рисунок 1. Высокоуровневая архитектура

Порядок заполнения документа:

1) Укажите дату операции. Рядом с датой отображается ссылка на шаблон операции, если он использовался при ее создании.

2) Если есть доходы, перечислите их на закладке Доходы. Обратите внимание, что доходы могут быть записаны не только в кошельки, но и в другие разделы учета. Например, можно учесть получение имущества в подарок, выполнение работ в долг, или списание части своей задолженности.

3) Если требуется детализировать учет доходов по дополнительным разрезам, настройте аналитику с помощью команды Аналитика на закладке Доходы.

4) Сделанные расходы можно перечислить на закладке Расходы. Расходовать можно не только деньги из кошельков, но и стоимость имущества, или остатки долгов. Например, можно учесть безвозмездную передачу имущества, начисление арендной платы или долга за коммунальные услуги, начисление комиссий и штрафов по банковским кредитам и т.п.

б) Для учета перемещений заполните строки на закладке Перемещения: укажите откуда, куда и какую сумму перемещаете. В перемещениях могут использоваться не только кошельки, но и другие разделы учета. Например, можно учесть перенос остатка с одного долга на другой, погашение долга имуществом и т.д.

Операцию можно учесть, чтобы она изменила остатки по соответствующим разделам учета, или сохранить без отражения в учете, как черновик. Кроме того, операцию можно пометить на удаление, чтобы она не только не отражалась в учете, но и была безвозвратно удалена при обслуживании информационной базы. Изменить учет операции и пометку на удаление можно с помощью соответствующих команд.

Модуль документа **Универсальная Операция** содержит следующие процедуры:

✗ `РасходыПриИзменении()` не тестируется, так как является обработчиком события формы. Не имеет входных параметров. Вызывает из тела функции, которые будут участвовать в тестировании.

✗ `ДоходыПриИзменении()` не тестируется, так как является обработчиком события формы. Не имеет входных параметров. Вызывает из тела функции, которые будут участвовать в тестировании.

✗ `ПеремещенияПриИзменении()` не тестируется, так как является обработчиком события формы. Не имеет входных параметров. Вызывает из тела функции, которые будут участвовать в тестировании.

✓ `ДобавитьИтогВТаблицуОстатков(ТаблицаПоВалютам, ИмяКолонкиИтого)` тестируется блочными тестами.

✓ Сформировать Движения Документа (Ссылка На Документ, Движения Документа, Дополнительные Свойства) тестируется блочными тестами. Связь с общим модулем **Разделы Учета** путем вызова функции **Добавить Проводку В Движения** тестируется интеграционными тестами.

✓ Записать Объекты Операции (Ссылка На Документ, Таблицы Документа, Дополнительные Свойства) тестируется блочными тестами.

✓ Пересчитать Сумму Строки В Валюту Учета (Строка Итого, Курсы Валют, Валюта Учета) тестируется блочными тестами.

На рисунке 2 показано взаимодействие модуля документа **Универсальная Операция** с другими объектами системы и модулями. Зелеными стрелками показано, что будет тестироваться блочными тестами, голубой – интеграционными.

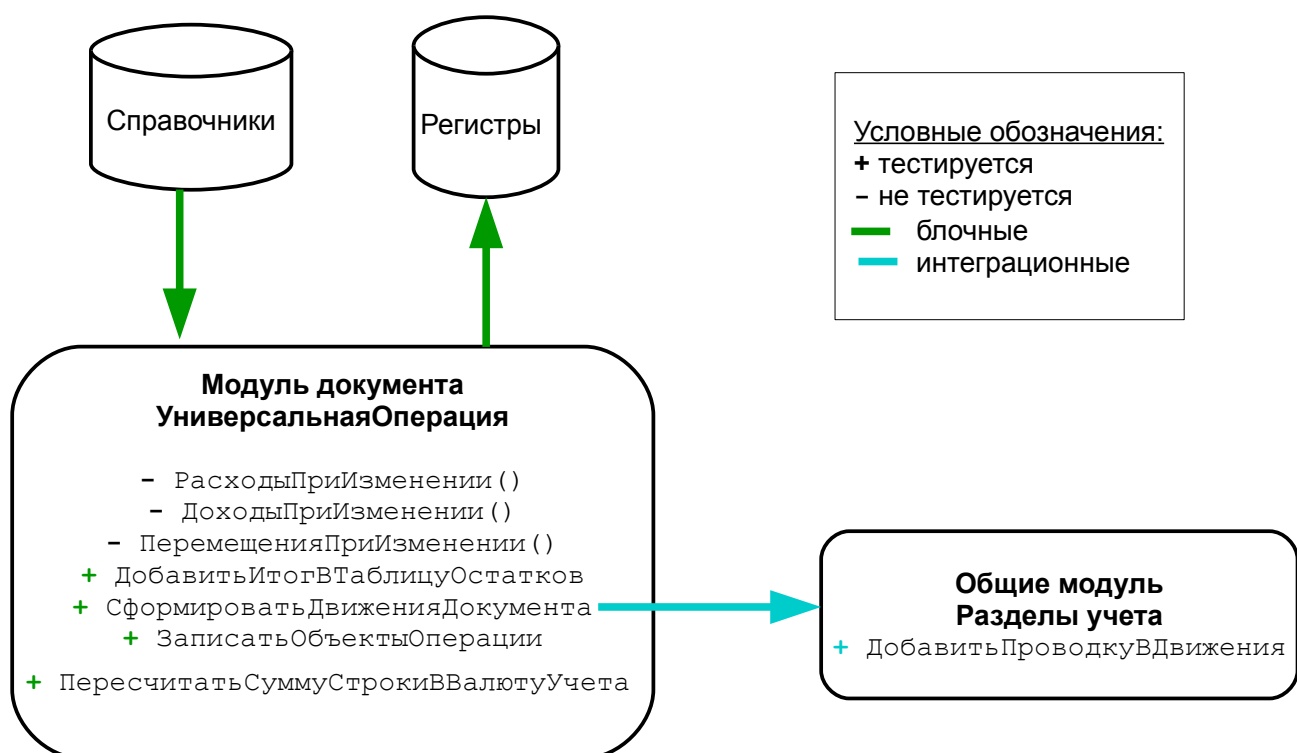


Рисунок 2. Взаимодействие между компонентами системы

## 2.1 Описание проведения тестирования

### 2.1.1 Блочное тестирование

Для реализации блочного тестирования используются встроенные механизмы платформы 1С 8.3 для автоматизированного тестирования — процесс, представляющий собой имитацию интерактивных действий пользователя и проверку результатов этих действий. Процесс автоматизированного тестирования заключается во взаимодействии двух специальным образом запущенных клиентских приложений: **менеджера тестирования** и **клиента тестирования**:

- На менеджере тестирования исполняется алгоритм теста, внешняя обработка *Тест* (которая будет написана мною).
- Клиент тестирования воспроизводит интерактивные действия пользователя. Клиентом будет являться демонстрационная база 1С: Деньги 8, релиз 2.0.15.9.

Результат выполнения автоматизированного теста будет контролироваться визуально путем сравнения полученных результатов с эталонными значениями.

Для написания тестов будет использована возможность автоматической записи интерактивных действий пользователя в XML файл. Запись действий пользователя происходит в XML тест, на основании которого можно создать Тест. Для преобразования XML в программный код используется готовая специальная обработка *UILogToScript.epf*.

В режиме конфигуратора запускается скрипт *xddTestRunner.js*, который позволяет запускать *Тест*. Результат выполнения теста отслеживается также в режиме конфигуратора.

Обработка *UILogToScript.epf* и скрипт *xddTestRunner.js* можно скачать здесь <https://github.com/xDrivenDevelopment/xUnitFor1C>.

Блочными тестами будут тестироваться следующие процедуры в модуле документа УниверсальнаяОперация:

- ✓ ДобавитьИтогВТаблицуОстатков (ТаблицаПоВалютам, ИмяКолонкиИтого)
- ✓ СформироватьДвиженияДокумента (СсылкаНаДокумент, ДвиженияДокумента, ДополнительныеСвойства)
- ✓ ЗаписатьОбъектыОперации (СсылкаНаДокумент, ТаблицыДокумента, ДополнительныеСвойства)
- ✓ ПересчитатьСуммуСтрокиВВалютуУчета (СтрокаИтого, КурсыВалют, ВалютаУчета)

### 2.1.2 Интеграционное тестирование

Восходящее интеграционное тестирование проводилось с целью проверки взаимодействия модулей документа УниверсальнаяОперация и общего модуля РазделыУчета.

Для реализации интеграционных тестов также была написана специальная обработка *Тест*. Макет данных для использования в интеграционных тестах генерируется с помощью использования другой готовой обработки *xddDataFixtureGen.epf*.

Для выполнения тестов не требуется запускать программу в режиме отладки. В режиме конфигуратора запускается скрипт *xddTestRunner.js*, позволяющий выполнять тесты прямо из конфигуратора.

Результат выполнения теста будет контролироваться автоматически путем сравнения полученных результатов с эталонными значениями. Для автоматической проверки используется обработка *Тест\_ПроверитьОтчетНаСоответствиеЭталону.epf*.

Интеграционными тестами будут тестироваться модуль документа УниверсальнаяОперация и общий модуль РазделыУчета. Конкретно, из процедуры СформироватьДвиженияДокумента модуля УниверсальнаяОперация вызывается процедура ДобавитьПроводкуВДвижения из модуля РазделыУчета.

Обработки *UILogToScript.epf* и *ПроверитьОтчетНаСоответствиеЭталону.epf*, а также скрипт



[https://github.com/xDrivenDevelopment/xUnitForIC.](https://github.com/xDrivenDevelopment/xUnitForIC)

### 2.1.3 Аттестационное тестирование

Для проведения аттестационного тестирования необходимо изучить документ УниверсальнаяОперация, определить последовательность заполнения и проведения документа.

Аттестационное тестирование выполняется в режиме отладки программы и анализируется визуально пользователем-экспертом путем сравнения с ожидаемым результатом.

Основные действия с документом УниверсальнаяОперация, которые будут проверены аттестационным тестированием:

- **Учесть и закрыть** сохранит операцию в информационной базе, учтет ее и закроет форму.

- **Учесть** так же сохранит и учтет операцию, но не будет закрывать ее форму.

- **Записать** сохранит сделанные изменения без закрытия формы.

- **Пометить на удаление / снять пометку** установит или снимет пометку на удаление. При этом все изменения сохранятся автоматически, а учтенная операция станет неучтенной.

- **Скопировать** откроет новую операцию - полную копию текущей операции.

- **Отменить учет** сделает учтенную операцию неучтенной.

- **Записи в учете** откроет форму, в которой будут перечислены все доходы, расходы и перемещения, сделанные операцией.

- **Закреть** закроет форму операции. Если есть несохраненные изменения, программа предложит сохранить их.

## 2.2 Описание тестируемых процедур

### 1) Документ УниверсальнаяОперация

ДобавитьИтогВТаблицуОстатков (ТаблицаПоВалютам, ИмяКолонкиИтого)  
после проведения документа, обновляет суммы расходов и доходов основной таблицы операций

#### Параметры:

- ТаблицаПоВалютам – содержит пару [валюта, сумма]
- ИмяКолонкиИтого – может иметь значение: Доход, Расход или Перемещение

### 2) Документ УниверсальнаяОперация

СформироватьДвиженияДокумента (СсылкаНаДокумент, ДвиженияДокумента, ДополнительныеСвойства) формирует и записывает движения в регистры учета

#### Параметры:

- СсылкаНаДокумент – тип ДокументСсылка.УниверсальнаяОперация, регистратор движений
- ДвиженияДокумента – КоллекцияДвижений или иная коллекция, наборы записей регистров учета
- ДополнительныеСвойства – тип структура, содержит дополнительные свойства документа

### 3) Документ УниверсальнаяОперация

ЗаписатьОбъектыОперации (СсылкаНаДокумент, ТаблицыДокумента, ДополнительныеСвойства) записывает в регистр сведений информацию об использованных в документе объектах учета.

#### Параметры:

- СсылкаНаДокумент – тип ДокументСсылка.УниверсальнаяОперация, регистратор движений
- ДополнительныеСвойства – тип структура, содержит дополнительные свойства документа

- ТаблицыДокумента – тип структура, набор таблиц с информацией о документе

#### 4) Документ УниверсальнаяОперация

ПересчитатьСуммуСтрокиВВалютуУчета (СтрокаИтого, КурсыВалют, ВалютаУчета) пересчитывает итоговую сумму строки в predetermined пользователем валюту учета.

##### Параметры:

- СтрокаИтого – содержит пару [валюта, сумма]
- КурсыВалют – справочник, таблица в формате [валюта, коэффициент]
- ВалютаУчета – тип СправочникСсылка.Валюты, наименование валюты учета

#### 5) Общий модуль РазделыУчета

ДобавитьПроводкуВДвижения (ЖурналОпераций, Период, Регистратор, СчетДт, СубконтоДт1, СубконтоДт2, СуммаВВалютеДт, ВалютаДт, СчетКт, СубконтоКт1, СубконтоКт2, СуммаВВалютеКт, ВалютаКт, Содержание = "", ДопАналитика = Неопределено)

Для строки документа делает проводку в регистре.

##### Параметры:

- Журнал операций – ДвиженияДокумента.ЖурналОпераций, predetermined операции для данного документа
- Период – дата и время текущего документа, в формате ДД.ММ.ГГГГ 00:00:00
- Регистратор – ссылка на текущий документ (ДокументСсылка.УниверсальнаяОперация)
- СчетДт – ПланСчетов.РазделыУчета
- СубконтоДт1 – СправочникСсылка.КошелькиИСчета
- СубконтоДт2 – СправочникСсылка.ФинансовыеЦели
- СуммаВВалютеДт – число, сумма в валюте
- ВалютаКт – СправочникСсылка.Валюты
- Содержание – строка, комментарий из строки документа

## 2.3 Критерии тестирования

- **Критерий прохождения тестов (применим для всех видов тестов)**

Любой тест считается успешно пройденным, если ожидаемый и фактический результаты совпадают. Если тест завершается неудачей, то перед принятием решения целесообразно проверить правильность самого теста.

Если тест завершился неудачей и тест реализован правильно, то производится заключение о найденной ошибке.

- **Критерий приостановления тестирования (применим для всех видов тестов)**

Тестирование должно быть приостановлено, если количество не пройденных тестов превысит 10% от общего количества.

- **Критерий возобновления работы (применим для всех видов тестов)**

Необходимо заново начать тестирование при получении уведомления, что найденные при тестировании ошибки исправлены.

## 3 Тесты

### 3.1 Блочные тесты

Процедура ДобавитьИтогВТаблицуОстатков (ТаблицаПоВалютам, ИмяКолонкиИтого)

После проведения документа, обновляет суммы расходов и доходов в основной таблице операций.

*Критерии корректных входных данных:*

ТаблицаПоВалютам содержит пару [СправочникСсылка.Валюты, число]  
ИмяКолонкиИтого может иметь 3 значения: Доход, Расход, Перемещение.

#### ***Тест 1***

Тип теста: простой,

Описание: проверяет обновление суммы после проведения документа,

Входные данные: ТаблицаПоВалютам = [руб., 2000],

ИмяКолонкиИтого = Доход,

Ожидаемый результат: обновит итоговую сумму в колонке Доход на 2000 в рублях.

#### ***Тест 2***

Тип теста: общий,

Описание: проверяет, каким образом обновиться сумма, если будет введено отрицательное значение,

Входные данные: ТаблицаПоВалютам = [СправочникСсылка.Валюты, -10],

Ожидаемый результат: обновит с учетом вычитания.

#### ***Тест 3***

Тип теста: негативный,

Описание: проверяет, каким образом отреагирует система, если будет вместо числового значения суммы будет введено символьное,

Входные данные: ТаблицаПоВалютам = сумма символьное значение,

Ожидаемый результат: Сообщение об ошибке.

#### **Тест 4**

Тип теста: специальный,

Описание: проверяет, каким образом отреагирует система, если вместо предопределенного имени (доход, расход или перемещение), будет введено непредопределенное.

Входные данные: ИмяКолонкиИтого = несуществующее или «»,

Ожидаемый результат: завершение без ошибки, сумма не обновляется.

Процедура СформироватьДвиженияДокумента (СсылкаНаДокумент,  
ДвиженияДокумента, ДополнительныеСвойства)

Формирует и записывает движения в регистры учета

*Критерии корректных входных данных:*

- СсылкаНаДокумент – тип ДокументСсылка.УниверсальнаяОперация,
- ДвиженияДокумента – тип КоллекцияДвижений
- ДополнительныеСвойства – тип Структура

#### **Тест 5**

Тип теста: простой,

Описание: проверяет формирование и запись движения в регистры учета.

Входные данные: СсылкаНаДокумент = Универсальная операция от 18.01.2009 (12:00), ДвиженияДокумента – КоллекцияДвижений, ДополнительныеСвойства – Структура.

Ожидаемый результат: Документ помечается проведенным, движения формируются.

#### **Тест 6**

Тип теста: негативный,

Описание: проверяет работу функции, если передать в качестве параметра ссылку на несуществующий документ.

Входные данные: СсылкаНаДокумент = несуществующая ссылка,

Ожидаемый результат: Документ остается непроведенным, движения не формируются.

### ***Тест 7***

Тип теста: общий,

Описание: проверяет работу функции, если передать в качестве параметра ссылку на другой документ.

Входные данные: СсылкаНаДокумент = ссылка на другой документ,

Ожидаемый результат: Формирование движений по другому документу.

### ***Тест 8***

Тип теста: негативный,

Описание: проверяет проведение документа, когда вместо ссылки ничего не передается.

Входные данные: СсылкаНаДокумент = «»,

Ожидаемый результат: Документ не проводится, движения не формируются.

### ***Тест 9***

Тип теста: общий,

Описание: проверяет правильность выполнения проведения, когда нет движений документа.

Входные данные: ДвиженияДокумента = «»,

Ожидаемый результат: Документ отмечается проведенным, движения не формируются.

### ***Тест 10***

Тип теста: общий,

Описание: проверяет правильность выполнения проведения, когда движения не соответствуют документу.

Входные данные: ДвиженияДокумента = несоответствующая документу КоллекцияДвижений,

Ожидаемый результат: Документ отмечается проведенным, движения формируются неправильно (в соответствии с указанной коллекцией).

### ***Тест 11***

Тип теста: негативный,

Описание: проверяет проведение документа, если переданные в процедуру свойства не соответствуют документу.

Входные данные: ДополнительныеСвойства = «» или не соответствуют документу,

Ожидаемый результат: Сообщение об ошибке.

Процедура ЗаписатьОбъектыОперации (СсылкаНаДокумент,  
ТаблицыДокумента, ДополнительныеСвойства)

Записывает в регистр сведений информацию об использованных в документе объектах учета.

*Критерии корректных входных данных:*

- СсылкаНаДокумент – тип ДокументСсылка.УниверсальнаяОперация
- ТаблицыДокумента – тип структура (ТаблицыЗначений: РасходыКоличество, РасходыНаЦель, СводТаблиц)
- ДополнительныеСвойства – тип структура

### ***Тест 12***

Тип теста: общий,

Описание: проверяет запись в регистр сведений информации об использованных объектах учета,



Входные данные: СсылкаНаДокумент = Универсальная операция от 18.01.2009 (12:00), ТаблицыДокумента – соответствующая структура структура, ДополнительныеСвойства – структура,

Ожидаемый результат: в регистре сведений сделаны соответствующие записи о об использованных в документе объектах учета.

### ***Тест 13***

Тип теста: негативный,

Описание: проверяет реакцию системы при передаче несуществующей ссылки на документ.

Входные данные: СсылкаНаДокумент = несуществующая ссылка,

Ожидаемый результат: Сообщение об ошибке.

### ***Тест 14***

Тип теста: общий,

Описание: проверяет заполнение регистра, если передать в процедуру ссылку на другой документ.

Входные данные: СсылкаНаДокумент = ссылка на другой документ,

Ожидаемый результат: Запись в регистре сведений об использованных в другом документе объектах учета.

### ***Тест 15***

Тип теста: негативный,

Описание: проверяет заполнение регистра, если не передать пустую ссылку на документ.

Входные данные: СсылкаНаДокумент = «»,

Ожидаемый результат: Записи в регистре не формируются.

### ***Тест 16***

Тип теста: общий,

Описание: проверка выполнения процедуры при отсутствии или неверных дополнительных свойств документа,

Входные данные: ДополнительныеСвойства = «» или не соответствуют документу,

Ожидаемый результат: Записи в регистре формируются.

### ***Тест 17***

Тип теста: общий,

Описание: проверка выполнения процедуры при отсутствии или неверных таблицах документа,

Входные данные: ТаблицыДокумента = «» или не соответствуют документу,

Ожидаемый результат: Записи в регистре формируются.

Процедура ПересчитатьСуммуСтрокиВВалютуУчета (СтрокаИтого,  
КурсыВалют, ВалютаУчета)

Пересчитывает итоговую сумму строки в predetermined пользователем валюту учета

*Критерии корректных входных данных:*

- СтрокаИтого – [СправочникСсылка.Валюты, сумма (число)]
- КурсыВалют – СправочникСсылка.КурсыВалют, данные в формате [СправочникСсылка.Валюты, коэффициент (число)]
- ВалютаУчета – СправочникСсылка.Валюты

### ***Тест 18***

Тип теста: специальный,

Описание: проверяет реакцию системы при передаче в параметр курсов валют несуществующие данные.

Входные данные: КурсыВалют = несуществующая таблица или «»,

Ожидаемый результат: Возврат 0.

### **Тест 19**

Тип теста: специальный,

Описание: проверяет реакцию системы при передачи в параметр валюты учета несуществующих данные.

Входные данные: ВалютаУчета = «» или не существующее в справочнике,

Ожидаемый результат: Возврат 0.

### **Тест 20**

Тип теста: простой,

Описание: проверяет правильность выполнения процедуры при передачи в параметры корректные данные.

Входные данные: СтрокаИтого = [«руб.», 1000], КурсВалют = Перечисления.КурсыВалют,

Ожидаемый результат: 1000.

## **3.2 Интеграционные тесты**

Процедура СформироватьДвиженияДокумента → Общий модуль  
РазделыУчета.ДобавитьПроводкуВДвижения (ЖурналОпераций, Период,  
Регистратор, СчетДт, СубконтоДт1, СубконтоДт2, СуммаВВалютеДт,  
ВалютаДт, СчетКт, СубконтоКт1, СубконтоКт2, СуммаВВалютеКт,  
ВалютаКт, Содержание = "", ДопАналитика = Неопределено)

Для строки документа делает проводку в регистре.

*Критерии корректных входных данных:*

- Журнал операций – тип РегистрБухгалтерииНаборЗаписей.ЖурналОпераций,
- Период – тип Дата
- Регистратор – тип ДокументСсылка.УниверсальнаяОперация
- СчетДт – тип ПланСчетов.РазделыУчета
- СубконтоДт1 – тип СправочникСсылка.КошелькиИСчета
- СубконтоДт2 – тип СправочникСсылка.ФинансовыеЦели

- СуммаВВалютеДт – тип число
- ВалютаКт – тип СправочникСсылка.Валюты
- Содержание – тип строка

### ***Тест 21***

Тип теста: общий,

Описание: проверяет создание проводки в регистре для строки документа.

Входные данные: Журнал операций –

РегистрБухгалтерииНаборЗаписей.ЖурналОпераций, Период – 18.01.2015 09:00:00, Регистратор – Универсальная операция от 18.01.2009 (12:00), СчетДт – Основные суммы долгов, СубконтоДт1 – Карта сбербанк, СубконтоДт2 – Накопления без конкретной цели, СуммаВВалютеДт – 3500, ВалютаКт – руб., Содержание – «покупка платья»,

Ожидаемый результат: Проводка формируется.

### ***Тест 22***

Тип теста: негативный,

Описание: проверяет реакцию на некорректные данные.

Входные данные: Период = 30.02.2014 09:00:00,

Ожидаемый результат: Проводка не формируется, сообщение об ошибке.

### ***Тест 23***

Тип теста: негативный,

Описание: проверяет правильность формирования записи в регистре, если в качестве параметра передан другой регистратор.

Входные данные: Регистратор = ссылка на другой документ,

Ожидаемый результат: Проводка не формируется.

### ***Тест 24***

Тип теста: специальный,

Описание: проверяет формирование проводки, если в документе отрицательная сумма.

Входные данные: СуммаВВалютеДт – отрицательное число,

Ожидаемый результат: Формируется проводка, отрицательное значение выделяется красным.

#### **Покрытие исполняемого кода тестами**

- a. Количество строк кода тестируемых модулей: 987;
- b. Количество строк кода, покрытых тестами 710;
- c.  $\text{Покрытие} = (710 / 987) * 100\% = 70,9 \%$

### **3.3 Аттестационные тесты**

#### ***Тест 25***

Тип теста: приемочный,

Описание: проверяет открытие нового документа УниверсальнаяОперация,

Входные данные: открыть документ,

Ожидаемый результат: открыт новый пустой документ УниверсальнаяОперация.

#### ***Тест 26***

Тип теста: приемочный,

Описание: проверка формирования документа УниверсальнаяОперация и функции «учесть».

Входные данные: создание нового документа Универсальная операция, заполнение его произвольными данными, нажимаем кнопку «учесть».

Ожидаемый результат: сохранит операцию в информационной базе, сформирует проводки, форма останется открытой.

#### ***Тест 27***

Тип теста: приемочный,

Описание: проверка формирования документа и функции «учесть и закрыть»,

Входные данные: создание нового документа УниверсальнаяОперация, заполнение его произвольными данными, нажимаем кнопку «учесть и закрыть»,

Ожидаемый результат: сохранит операцию в информационной базе, сформирует проводки и закроет форму.

### ***Тест 28***

Тип теста: приемочный,

Описание: проверка формирования документа и функции «записать»

Входные данные: создание нового документа УниверсальнаяОперация, заполнение его произвольными данными, нажимаем кнопку «записать»,

Ожидаемый результат: сохранит сделанные изменения без закрытия формы.

### ***Тест 29***

Тип теста: приемочный,

Описание: проверка функции «пометить на удаление»

Входные данные: выделить нужный документ в списке операций, нажать кнопку «пометить на удаление»,

Ожидаемый результат: пометка на удаленаие напротив документа, все проводки по документу отменены.

### ***Тест 30***

Тип теста: приемочный,

Описание: проверка функции «снять пометку на удаление»

Входные данные: выделить нужный документ в списке операций, нажать кнопку «снять пометку на удаление»,

Ожидаемый результат: документ не помечен, движения не сформированы.

### ***Тест 31***

Тип теста: приемочный,

Описание: проверка функции «скопировать»

Входные данные: выделить нужный документ в списке операций, нажать кнопку «скопировать»,

Ожидаемый результат: открывает новый документ, полную копию скопированной операции.

### ***Тест 32***

Тип теста: приемочный,

Описание: проверка функции «отменить учет»

Входные данные: выделить нужный документ в списке операций, нажать кнопку «отменить учет»,

Ожидаемый результат: удалит все проводки по документу.

### ***Тест 33***

Тип теста: приемочный,

Описание: проверка функции «записи в учете»

Входные данные: выделить нужный документ в списке операций, нажать кнопку «записи в учете»,

Ожидаемый результат: откроет форму, где отображаются все проводки документа.

### ***Тест 34***

Тип теста: приемочный,

Описание: проверка функции «заккрыть»

Входные данные: открыть документ, нажать кнопку «заккрыть»,

Ожидаемый результат: закроет форму документа. Если есть несохраненные изменения, программа предложит сохранить их.

## 4 Примеры тестов

Пример *Теста 1* для процедуры ДобавитьИтогВТаблицуОстатков.

```
Процедура Тест1ДобавитьИтогВТаблицуОстатков ()
//Параметр ТаблицаПоВалютам = СправочникСсылка.Валюты
//Параметр ИмяКолонкиИтого = «Доход»
//Значение переменной Валюта = «руб.»
    Для Каждого СтрокаВалюты Из СправочникСсылка.Валюты Цикл
        СтрокаИтого = ТаблицаИтогов.НайтиСтроки(Новый
Структура ("Валюта", СтрокаВалюты.Валюта));
        Если СтрокаИтого.Количество() = 0 Тогда
            СтрокаИтого = ТаблицаИтогов.Добавить();
            СтрокаИтого.Валюта = СтрокаВалюты.Валюта;
        Иначе
            СтрокаИтого = СтрокаИтого[0];
        КонецЕсли;
        СтрокаИтого[Доход] = СтрокаИтого[Доход] +
СтрокаВалюты.Сумма;
    КонецЦикла;
КонецПроцедуры
```



## 5 Результаты тестирования

*Блочные тесты:*

№ теста	Модуль	Тестируемая процедура	Кол-во тестов	Кол-во ошибок	Дата проведения
1	Универсальная Операция	ДобавитьИтогВТаблицуОстатков	3	0	21.01.14
2	Универсальная Операция	ДобавитьИтогВТаблицуОстатков	3	0	18.01.14
3	Универсальная Операция	ДобавитьИтогВТаблицуОстатков	3	0	18.01.14
4	Универсальная Операция	ДобавитьИтогВТаблицуОстатков	3	0	18.01.14
5	Универсальная Операция	СформироватьДвиженияДокумента	3	0	21.01.14
6	Универсальная Операция	СформироватьДвиженияДокумента	3	0	18.01.14
7	Универсальная Операция	СформироватьДвиженияДокумента	3	0	18.01.14
8	Универсальная Операция	СформироватьДвиженияДокумента	3	0	18.01.14
9	Универсальная Операция	СформироватьДвиженияДокумента	3	0	18.01.14
10	Универсальная Операция	СформироватьДвиженияДокумента	3	0	18.01.14
11	Универсальная Операция	СформироватьДвиженияДокумента	3	3 (см. Найденные ошибки п.1)	18.01.14
12	Универсальная Операция	ЗаписатьОбъектыОперации	3	0	21.01.14
13	Универсальная Операция	ЗаписатьОбъектыОперации	3	0	18.01.14
14	Универсальная Операция	ЗаписатьОбъектыОперации	3	0	18.01.14
15	Универсальная Операция	ЗаписатьОбъектыОперации	3	0	18.01.14
16	Универсальная Операция	ЗаписатьОбъектыОперации	3	0	18.01.14
17	Универсальная Операция	ЗаписатьОбъектыОперации	3	0	18.01.14
18	Универсальная Операция	ПересчитатьСуммуСтрокиВВалютуУчета	3	0	18.01.14
19	Универсальная Операция	ПересчитатьСуммуСтрокиВВалютуУчета	3	0	18.01.14
20	Универсальная Операция	ПересчитатьСуммуСтрокиВВалютуУчета	3	0	18.01.14

### *Интеграционные тесты:*

№ теста	Модуль	Тестируемая процедура	Кол-во тестов	Кол-во ошибок	Дата проведения
21	Разделы Учета	Добавить Проводку В Движения	3	0	21.01.14
22	Разделы Учета	Добавить Проводку В Движения	3	0	18.01.14
23	Разделы Учета	Добавить Проводку В Движения	3	3 (см. Найденные ошибки п.2)	18.01.14
24	Разделы Учета	Добавить Проводку В Движения	3	0	18.01.14

### *Аттестационные тесты:*

№ теста	Результат	Дата проведения
25	пройден	10.01.14
26	пройден	10.01.14
27	пройден	10.01.14
28	пройден	10.01.14
29	пройден	21.01.14
30	пройден	21.01.14
31	пройден	21.01.14
32	пройден	21.01.14
33	пройден	21.01.14
34	пройден	21.01.14

### *Найденные ошибки:*

1. Тест проверяет проведение документа, если переданные в процедуру свойства не соответствуют документу. В результате проведения теста документ отмечался проведенным и движения документа формировались корректно. Можно сделать вывод, что свойства документа не играют роли для этой процедуры.

2. Тест проверяет правильность формирования записи в регистре. Если в качестве параметра передан другой регистратор, то проводка не должна формироваться. В результате проведения теста сформировалась проводка, но при этом в качестве регистратора указан документ, что является некорректным с точки зрения работы программы.