

**Министерство образования и науки Российской Федерации**  
федеральное государственное автономное образовательное учреждение  
высшего образования  
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

---

Институт кибернетики  
Направление подготовки 09.04.01 «Информатика и вычислительная техника»  
Кафедра автоматизации и компьютерных систем

**МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ**

Тема работы
Автоматизация процесса управления тестами объектов системы автоматизированного проектирования бортового программного обеспечения

УДК 004.42:004.896:629.73.05

Студент

Группа	ФИО	Подпись	Дата
8ВМ5Г	Баглаева Елена Алексеевна		

Руководитель

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент кафедры АиКС	Цапко Сергей Геннадьевич	к.т.н.		

**КОНСУЛЬТАНТЫ:**

По разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент кафедры МЕН	Конотопский Владимир Юрьевич	к.э.н.		

По разделу «Социальная ответственность»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент кафедры ЭБЖ	Извеков Владимир Николаевич	к.т.н.		

**ДОПУСТИТЬ К ЗАЩИТЕ:**

И.о. зав. кафедрой	ФИО	Ученая степень, звание	Подпись	Дата
АиКС	Суходоев Михаил Сергеевич.	к.т.н.		

## ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ООП

Код результата	Результат обучения (выпускник должен быть готов)
<b><i>Общепрофессиональные компетенции</i></b>	
P1	Воспринимать и самостоятельно приобретать, развивать и применять математические, естественнонаучные, социально-экономические и профессиональные знания для решения нестандартных задач, в том числе в новой или незнакомой среде и в междисциплинарном контексте.
P2	Владеть и применять методы и средства получения, хранения, переработки и трансляции информации посредством современных компьютерных технологий, в том числе в глобальных компьютерных сетях.
P3	Демонстрировать культуру мышления, способность выстраивать логику рассуждений и высказываний, основанных на интерпретации данных, интегрированных из разных областей науки и техники, выносить суждения на основании неполных данных, анализировать профессиональную информацию, выделять в ней главное, структурировать, оформлять и представлять в виде аналитических обзоров с обоснованными выводами и рекомендациями.
P4	Анализировать и оценивать уровни своих компетенций в сочетании со способностью и готовностью к саморегулированию дальнейшего образования и профессиональной мобильности. Владеть, по крайней мере, одним из иностранных языков на уровне социального и профессионального общения, применять специальную лексику и профессиональную терминологию языка.
<b><i>Профессиональные компетенции</i></b>	
P5	Выполнять инновационные инженерные проекты по разработке аппаратных и программных средств автоматизированных систем различного назначения с использованием современных методов проектирования, систем автоматизированного проектирования, передового опыта разработки конкурентно способных изделий.
P6	Планировать и проводить теоретические и экспериментальные исследования в области проектирования аппаратных и программных средств автоматизированных систем с использованием новейших достижений науки и техники, передового отечественного и зарубежного опыта. Критически оценивать полученные данные и делать выводы.
P7	Осуществлять авторское сопровождение процессов проектирования, внедрения и эксплуатации аппаратных и программных средств автоматизированных систем различного назначения.
<b><i>Общекультурные компетенции</i></b>	
P8	Использовать на практике умения и навыки в организации исследовательских, проектных работ и профессиональной эксплуатации современного оборудования и приборов, в управлении коллективом.
P9	Осуществлять коммуникации в профессиональной среде и в обществе в целом, активно владеть иностранным языком, разрабатывать документацию, презентовать и защищать результаты инновационной инженерной деятельности, в том числе на иностранном языке.
P10	Совершенствовать и развивать свой интеллектуальный и общекультурный уровень. Проявлять инициативу, в том числе в ситуациях риска, брать на себя всю полноту ответственности.
P11	приобретать в процессе практической деятельности новые знания и умения, в том числе в новых областях знаний, непосредственно не связанных со сферой деятельности, способность к педагогической деятельности.

**Министерство образования и науки Российской Федерации**  
федеральное государственное автономное образовательное учреждение  
высшего образования  
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

---

Институт кибернетики  
Направление подготовки 09.04.01 «Информатика и вычислительная техника»  
Кафедра автоматизации и компьютерных систем

УТВЕРЖДАЮ:  
И.о. зав. кафедрой  
\_\_\_\_\_  
(Подпись)      \_\_\_\_\_ (Дата)      Суходоев М.С.  
(Ф.И.О.)

**ЗАДАНИЕ**  
**на выполнение выпускной квалификационной работы**

В форме:

магистерской диссертации

(бакалаврской работы, дипломного проекта/работы, магистерской диссертации)

Студенту:

Группа	ФИО
8ВМ5Г	Баглаевой Елене Алексеевне

Тема работы:

Автоматизация процесса управления тестами объектов системы автоматизированного проектирования бортового программного обеспечения

Утверждена приказом директора (дата, номер)	№786/с от 09.02.2017 г.
---	-------------------------

Срок сдачи студентом выполненной работы:	30.05.2017 г.
--	---------------

**ТЕХНИЧЕСКОЕ ЗАДАНИЕ:**

<b>Исходные данные к работе</b> <i>(наименование объекта исследования или проектирования; производительность или нагрузка; режим работы (непрерывный, периодический, циклический и т. д.); вид сырья или материал изделия; требования к продукту, изделию или процессу; особые требования к особенностям функционирования (эксплуатации) объекта или изделия в плане безопасности эксплуатации, влияния на окружающую среду, энергозатратам; экономический анализ и т. д.).</i>	<b>Объект исследования:</b> автоматизация процесса управления жизненным циклом бортового программного обеспечения космических аппаратов. <b>Требования:</b> реализация механизмов тестов в рамках доработки системы автоматизированного проектирования бортового программного обеспечения космических аппаратов.
--	---

<p><b>Перечень подлежащих исследованию, проектированию и разработке вопросов</b></p> <p><i>(аналитический обзор по литературным источникам с целью выяснения достижений мировой науки техники в рассматриваемой области; постановка задачи исследования, проектирования, конструирования; содержание процедуры исследования, проектирования, конструирования; обсуждение результатов выполненной работы; наименование дополнительных разделов, подлежащих разработке; заключение по работе).</i></p>	<p>1 Провести анализ предметной области – процесса проектирования бортового программного обеспечения космических аппаратов.</p> <p>2 Описать архитектуру САПР БПО.</p> <p>3 Реализовать механизм тестов и описать реализованные модификации.</p> <p>4 Составить программу и методику испытаний.</p>
<p><b>Перечень графического материала</b></p> <p><i>(с точным указанием обязательных чертежей)</i></p>	<p>Презентация в формате *.pptx</p>
<p><b>Консультанты по разделам выпускной квалификационной работы</b></p> <p><i>(с указанием разделов)</i></p>	
<p><b>Раздел</b></p>	<p><b>Консультант</b></p>
<p>Финансовый менеджмент, ресурсоэффективность и ресурсосбережение</p>	<p>Конотопский Владимир Юрьевич</p>
<p>Социальная ответственность</p>	<p>Извеков Владимир Николаевич</p>
<p>The testing programme and procedure</p>	<p>Денико Роман Викторович</p>
<p><b>Названия разделов, которые должны быть написаны на русском и иностранном языках:</b></p>	
<p>Программа и методика испытаний</p>	

<p><b>Дата выдачи задания на выполнение выпускной квалификационной работы по линейному графику</b></p>	<p>7.02.2017</p>
--	------------------

**Задание выдал руководитель:**

Должность	ФИО	Ученая степень, звание	Подпись	Дата
<p>Доцент каф. АиКС</p>	<p>Цапко С.Г.</p>	<p>к.т.н.</p>		

**Задание принял к исполнению студент:**

Группа	ФИО	Подпись	Дата
<p>8ВМ5Г</p>	<p>Баглаева Елена Алексеевна</p>		

**ЗАДАНИЕ ДЛЯ РАЗДЕЛА  
«ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСОЭФФЕКТИВНОСТЬ И  
РЕСУРСОСБЕРЕЖЕНИЕ»**

Студенту:

<b>Группа</b>	<b>ФИО</b>
8ВМ5Г	Баглаевой Елене Алексеевне

<b>Институт</b>	<b>ИК</b>	<b>Кафедра</b>	<b>АиКС</b>
Уровень образования	магистратура	Направление/специальность	09.04.01 Информатика и вычислительная техника

**Исходные данные к разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»:**

<i>1. Стоимость ресурсов научного исследования (НИ): материально-технических, энергетических, финансовых, информационных и человеческих</i>	1. Зарботная плата доцента – 23264,86Р. 2. Зарботная плата инженера - 7 864,11Р. 3. Стоимость 1 квт/ч для ТПУ – 5,257Р. 4. Стоимость оборудования - 61000Р.
<i>2. Нормы и нормативы расходования ресурсов</i>	1. 8-часовой рабочий день. 2. 5- и 6-дневная рабочая неделя. 3. Годовая норма амортизации – 33%.
<i>3. Используемая система налогообложения, ставки налогов, отчислений, дисконтирования и кредитования</i>	1. Единый социальный налог (ЕСН) – 30%; 2. Налог на добавленную стоимость (НДС) – 18%.

**Перечень вопросов, подлежащих исследованию, проектированию и разработке:**

<i>1. Организация и планирование работ</i>	1. Продолжительность этапов работ; 2. Расчет накопления готовности проекта.
<i>2. Расчет сметы затрат на выполнение проекта</i>	1. Расчеты затрат на заработную плату. 2. Расчеты амортизации и прочих расходов. 3. Расчет себестоимости и цены проекта, а также предполагаемой прибыли от реализации проекта.
<i>3. Оценка экономической эффективности проекта</i>	1. Определение типов эффекта; 2. Определение форм, которые может принимать экономический эффект.
<i>4. Оценка научно-технического уровня НИР</i>	Экспертная оценка параметров научно-технического эффекта разработки и расчет интегрального показателя научно-технического уровня разработки

**Перечень графического материала (с точным указанием обязательных чертежей):**

<i>1. Линейный график работ (диаграмма Ганта)</i>
---

**Дата выдачи задания для раздела по линейному графику**

**Задание выдал консультант:**

<b>Должность</b>	<b>ФИО</b>	<b>Ученая степень, звание</b>	<b>Подпись</b>	<b>Дата</b>
Доцент каф. МЕН	Конотопский Владимир Юрьевич	К.Э.Н.		

**Задание принял к исполнению студент:**

<b>Группа</b>	<b>ФИО</b>	<b>Подпись</b>	<b>Дата</b>
8ВМ5Г	Баглаева Елена Алексеевна		

## ЗАДАНИЕ ДЛЯ РАЗДЕЛА «СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ»

Студенту:

Группа	ФИО
8ВМ5Г	Баглаевой Елене Алексеевне

Институт	ИК	Кафедра	АиКС
Уровень образования	магистратура	Направление/специальность	09.04.01 Информатика и вычислительная техника

### Исходные данные к разделу «Социальная ответственность»:

<p>1. <i>Характеристика объекта исследования (вещество, материал, прибор, алгоритм, методика, рабочая зона) и области его применения</i></p>	<p>Информационная система поддержки процесса проектирования бортового программного обеспечения космических аппаратов предполагается для использования в организациях, специализирующихся на производстве программного обеспечения для космических аппаратов.</p>
--	--

### Перечень вопросов, подлежащих исследованию, проектированию и разработке:

<p><b>1. Производственная безопасность</b>  1.1. <i>Анализ вредных и опасных факторов, которые могут возникнуть на рабочем месте при проведении исследований;</i>  1.2. <i>Обоснование мероприятий по защите исследователя от действия опасных и вредных факторов.</i></p>	<p>Опасные и вредные факторы:  – неблагоприятный климат;  – недостаточная освещенность рабочей зоны;  – повышенный уровень шума;  – опасность поражения электрическим током;  – опасность возникновения пожара.  Мероприятия по защите от вредных факторов включают в себя измерение текущих показателей вредных факторов и обеспечение соблюдения нормативных показателей.  Для защиты от опасных факторов необходимо проводить организационные и технические мероприятия по предотвращению возникновения опасных ситуаций.</p>
<p><b>2. Экологическая безопасность:</b>  2.1. <i>Анализ влияния процесса исследования на окружающую среду;</i>  2.2. <i>Обоснование мероприятий по защите окружающей среды.</i></p>	<p>Объекты, несущие угрозу окружающей среде:  – люминесцентные лампы.  Необходимо обеспечить утилизацию объектов в специальных организациях.</p>
<p><b>3. Безопасность в чрезвычайных ситуациях:</b>  3.1. <i>Анализ вероятных ЧС, которые могут возникнуть на рабочем месте при проведении исследований;</i>  3.2. <i>Обоснование мероприятий по предотвращению ЧС и разработка порядка действий в случае возникновения ЧС.</i></p>	<p>ЧС, которые могут возникнуть в процессе разработки и эксплуатации:  – пожар в здании.  Требуется следовать инструкциям, чтобы не допустить возникновения ЧС. Однако, если ЧС произошло, требуется вызвать соответствующие службы для ликвидации последствий ЧС.</p>
<p><b>4. Правовые и организационные вопросы обеспечения безопасности:</b>  4.1. <i>Организационные мероприятия при компоновке рабочей зоны</i></p>	<p>Описание правил компоновки рабочего места с учетом специфики работы исполнителя проекта и пользователя программного продукта.</p>

Дата выдачи задания для раздела по линейному графику

**Задание выдал консультант:**

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент каф. ЭБЖ	Извеков Владимир Николаевич	к.т.н.		

**Задание принял к исполнению студент:**

Группа	ФИО	Подпись	Дата
8ВМ5Г	Баглаева Елена Алексеевна		

## РЕФЕРАТ

Выпускная квалификационная работа 116 страниц, 90 рисунков, 11 таблиц, 27 источников, 1 приложение.

Ключевые слова: разработка ПО, архитектура ПО, шаблон модель-представление-представление-модель, клиент-серверная архитектура, система управления данными об изделии, космический аппарат, бортовое программное обеспечение.

Объект исследования – автоматизация процесса управления жизненным циклом бортового программного обеспечения космических аппаратов.

Целью работы является расширение функционала информационной системы управления данными о бортовом программном обеспечении для хранения информации о тестах.

В процессе исследования проводился анализ требований предметной области.

В результате исследования реализован новый функционал прототипа информационной системы.

Степень внедрения: разработанная система находится на стадии внедрения в процесс проектирования.

Областью применения является АО «ИСС» или аналогичная компания, занимающаяся производством бортового программного обеспечения.

Оценка экономической эффективности возможна по косвенным показателям, связанным с экономией затрат на разработку, тестирование и сопровождение бортового программного обеспечения.

В будущем планируется внедрение информационной системы в процессы тестирования, верификации и сопровождения бортового программного обеспечения.



## **ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ**

КА – космический аппарат.

НКУ – наземный комплекс управления.

БКУ – бортовой комплекс управления.

БСУ – бортовая система управления.

БПО – бортовое программное обеспечение.

БВС – бортовая вычислительная система.

АО «ИСС» – АО «Информационные спутниковые системы» имени академика М. Ф. Решетнёва.

ТКПП БПО – технологический комплекс производства программ бортового программного обеспечения.

КСП М2 – кросс-система программирования Модуля-2.

СП ИСПОЛКОМ – система программирования команд управления.

СП РЕАЛ – система программирования макропрограмм интегрального управления.

НОК БПО – наземный отладочный комплекс БПО.

СИП БПО – система изготовления программ БПО.

АСПИД – автоматизированная система сопровождения программ, изделий и документов БПО.

САПР БПО – система автоматизированного проектирования бортового программного обеспечения космических аппаратов.

ORM – object-relational mapping (объектно-реляционное отображение).

## ОГЛАВЛЕНИЕ

Введение.....	13
1 Анализ предметной области .....	14
1.1 Назначение, состав и принцип функционирования БПО .....	14
1.2 Основные задачи при проектировании БПО.....	19
1.3 Подходы к проектированию БПО .....	20
1.4 Постановка задачи .....	29
2 Реализация механизма тестов .....	32
2.1 Описание архитектуры САПР БПО .....	32
2.1.1 Сервер базы данных.....	34
2.1.2 Сервер приложений .....	34
2.1.3 Клиентское приложение.....	39
2.2 Модификация САПР БПО.....	47
2.2.1 Модификация программных классов предметной области и структуры таблиц БД.....	48
2.2.2 Подготовка к реализации сервиса управления тестами.....	51
2.2.3 Реализация сервиса управления тестами.....	52
2.2.4 Доработка клиентского приложения для работы с тестами.....	53
3 Программа и методика испытаний.....	56
3.1 Перечень проводимых испытаний .....	57
3.1.1 Управление тестами на изделие .....	57
3.1.2 Управление тестами на систему.....	62
3.1.3 Матрица трассируемости требований в тесты.....	67
4 Финансовый менеджмент, ресурсоэффективность и ресурсосбережение...	70

4.1 Организация и планирование работ .....	70
4.1.1 Продолжительность этапов работ .....	71
4.1.2 Расчет накопления готовности проекта .....	76
4.2 Расчет сметы затрат на выполнение проекта .....	77
4.2.1 Расчет заработной платы .....	77
4.2.2 Расчет затрат на социальный налог .....	78
4.2.3 Расчет затрат на электроэнергию .....	78
4.2.4 Расчет амортизационных расходов .....	79
4.2.5 Расчет прочих расходов .....	80
4.2.8 Расчет общей себестоимости разработки .....	80
4.2.9 Расчет прибыли .....	81
4.2.10 Расчет НДС .....	81
4.2.11 Цена разработки НИР .....	81
4.3 Оценка экономической эффективности проекта .....	81
4.3.1 Оценка научно-технического уровня НИР .....	82
5 Социальная ответственность .....	84
Аннотация .....	84
Введение .....	84
5.1. Производственная безопасность .....	85
5.1.1. Микроклимат рабочего помещения .....	86
5.1.2. Производственное освещение .....	87
5.1.3. Производственные шумы .....	89
5.1.4. Электробезопасность .....	90
5.1.5. Пожарная безопасность .....	91
5.2. Экологическая безопасность .....	92

5.3. Безопасность в чрезвычайных ситуациях .....	93
5.4. Правовые и организационные вопросы обеспечения безопасности .....	95
Заключение .....	97
Список публикаций студента.....	98
Список использованных источников .....	99
Приложение А .....	102
3 The testing programme and procedure.....	103
3.1 The list of performed tests .....	104
3.1.1 Satellite test management .....	104
3.1.2 System test management.....	108
3.1.3 The traceability matrix of the requirements and the tests.....	113

## **ВВЕДЕНИЕ**

В процессе управления жизненным циклом космических аппаратов от этапа проектирования до запуска и сопровождения возникают вопросы контроля конфигурации аппаратов, включающие в себя хранение и поиск актуальных и предыдущих сборок бортовых систем, находящихся под управлением бортового программного обеспечения (БПО).

Система автоматизированного проектирования бортового программного обеспечения космических аппаратов (САПР БПО) является информационной системой хранения данных об изделиях (космических аппаратах, спутниках), обеспечивающей возможности управления их конфигурацией и заимствования составных элементов путем копирования.

На этапе проектирования БПО проектант заполняет информацию о составляющих изделия, однако при переходе к этапам тестирования и верификации как отдельных систем, так и изделия в целом, возникают трудности с оперативным сопоставлением и управлением тестами изделий и отдельных систем.

Расширение функционала САПР БПО для управления тестами изделий и систем позволит проектантам интегрировать в систему и структурировать справочники тестов для их использования на этапах тестирования и верификации БПО.

Предлагаемые доработки повысят эффективность контроля процесса тестирования и верификации отдельных программных компонент систем и изделий путем сокращения времени на поиск связанных с БПО тестов.

В данной работе рассматривается добавление в САПР БПО следующих видов тестов:

- тесты на изделие;
- тесты на систему;
- тесты на функциональное требование.

# 1 АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ

## 1.1 Назначение, состав и принцип функционирования БПО

КА состоит из целевой аппаратуры и служебных систем, выполняющих функции обеспечения энергией, регулирования температуры, защиты от радиации, управления движением и ориентации в пространстве, аварийного спасения, а также управления посадкой, отделением от носителя, стыковкой и др. В зависимости от выполняемой космическим аппаратом функции список служебных систем может отличаться.

Управление космическим аппаратом может осуществляться как наземным комплексом управления (НКУ), так и бортовым комплексом управления (БКУ).

Наземный комплекс управления представляет собой систему служб и средств, обладающую системой связи, включающей в себя кабельные, радиорелейные и спутниковые каналы.

Бортовой комплекс управления представляет собой комплекс систем, которые располагаются на борту космического летательного аппарата и осуществляют управление аппаратом. Общее число систем, отвечающих за функционирование космического аппарата и входящих в состав бортового комплекса управления, может быть более 50.

Для обеспечения стабильной работы КА решаются следующие задачи:

- обмен информацией с НКУ, в том числе сбор, хранение, обработка и передача телеметрической информации;
- снабжение КА электроэнергией;
- распределение электропитания между приборами и системами;
- поддержание теплового режима КА;
- управление ориентацией КА в пространстве;
- управление движением КА в пространстве, осуществляемым за счёт перемещения центра масс;

- управление угловым движением КА в пространстве, т.е. вращением КА вокруг центра масс;
- прогнозирование актуального местоположения КА на орбите;
- управление положением солнечных батарей (при их наличии);
- управление работой систем и оборудования КА с учетом программы полета КА и его реальным состоянием [1, 2].

Для автоматических космических аппаратов, имеющих целевую направленность (спутники связи, аппараты зондирования Земли или наблюдения участков звездного пространства) можно выделить общих перечень бортовых систем (Рисунок 1), включающий в себя:

- бортовую вычислительную систему (БВС), представляющую собой совокупность вычислительных средств и адаптеров связи и обеспечивающую информационное взаимодействие с бортовыми абонентами и предоставляющую ресурсы для управления и контроля работы систем КА;
- систему управления движением и навигации (систему ориентации и управления движением), управляющую движением центра масс КА и вращением вокруг центра масс КА;
- систему управления бортовой аппаратурой, осуществляющей коммутацию электропитания, усиления и преобразования электрических сигналов, а также передачи команд управления системам и приборам КА;
- систему бортовых измерений, осуществляющей сбор, обработку и передачу в НКУ телеметрической информации о результатах измерений состояния систем КА и протекающих на КА процессах;
- бортовые приборы служебного канала управления или командной радиолинии, представляющие собой радиотехнический комплекс, осуществляющий обмен служебной информацией между БКУ и НКУ;
- объединенная двигательная установка, которая состоит из комплекса двигателей и осуществляющая перемещение КА как относительно орбиты, так и угловое перемещение КА;

- система обеспечения теплового режима внутри КА;
- система электроснабжения, выполняющая преобразование первичной (солнечной) энергии в электрическую.

Для некоторых классов КА бортовая кабельная сеть рассматривается как отдельная структурная единица [1].

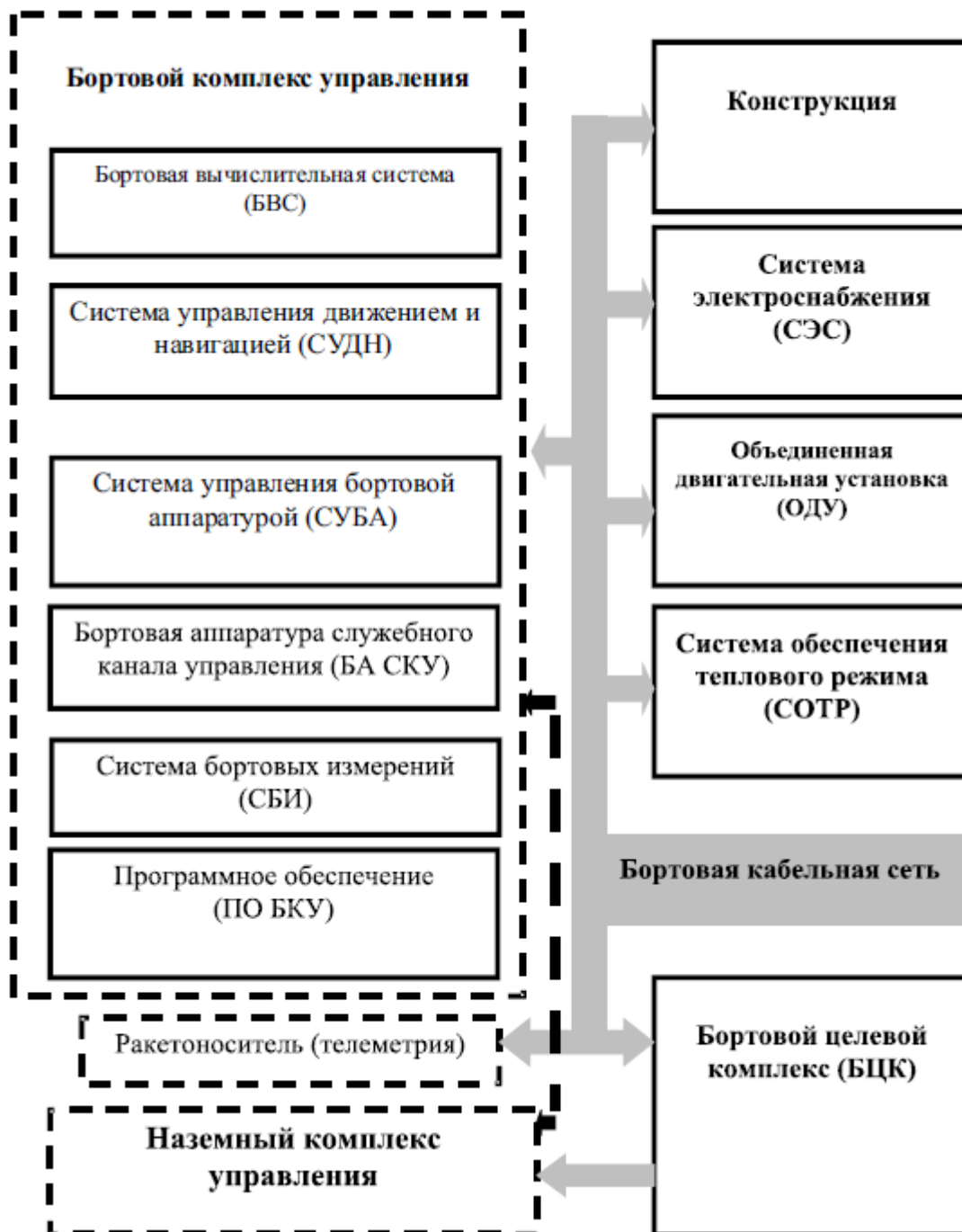


Рисунок 1 – Схема бортовых систем КА.

Иерархический принцип построения программного обеспечения БКУ подразумевает наличие следующих уровней:



- первый (нижний) уровень представлен драйверами обмена с аппаратурой и программами, организующими вычислительный процесс;
- второй уровень содержит программы, обеспечивающие управление и контроль работы бортового оборудования и приборов;
- третий уровень составляют расчетные программы и программы, обеспечивающие полетные режимы бортовых систем;
- четвертый (верхний) уровень включает в себя программы планирования и организации режимов работы всего БКУ, а также контроль состояния систем КА.

Архитектура программного обеспечения БКУ классифицирует программы на служебные и функциональные (Рисунок 2).

Служебные программы решают задачи:

- диспетчера,
- обмена информацией,
- управления конфигурацией БВС,
- таймирования и др.

В список функциональных программ входят:

- программы, управляющие включением и выключением отдельных приборов;
- программы, предназначенные для расчета подготовительной и сопроводительной информации;
- программы, предназначенные для формирования управляющих воздействий на отдельные приборы и др.

Каждый программный модуль имеет свои настроечные параметры и логико-информационные связи с другими программами.

Построение программного обеспечения БКУ требует организованного обмена информацией между программами всех уровней с учетом того, что управляющая информация должна поступать от верхнего уровня программ к

программам нижних уровней (сверху вниз), а контрольно-диагностическая информация – снизу вверх.



Рисунок 2 – Программное обеспечение БКУ.

Для осуществления функционирования программного обеспечения БКУ в реальном времени для каждой программы выделяются последовательность и конкретное время подключения на вычислительном такте бортового компьютера, а также требуемые вычислительные ресурсы.

Программное обеспечение БКУ позволяет выполнять не только функции контроля и управления, но и решать следующие задачи, например:

- решение задач, связанных с повитковым планированием полета КА;
- решения задач по оптимизации расхода бортовых ресурсов;
- обеспечение автономности существования КА;
- оперативное реагирование бортовых систем и оборудования на нештатные ситуации [1].

## **1.2 Основные задачи при проектировании БПО**

В ходе проектирования бортового программного обеспечения (БПО) спутников выделяются следующие крупные задачи:

- определение структуры и требований к БПО;
- разработка архитектурных проектов программного обеспечения систем;
- разработка и автономное тестирование компонент ПО системы;
- получение исполнимого кода БПО и тестирование интеграции БПО;
- системное тестирование программного обеспечения отдельных систем;
- системное тестирование БПО в целом;
- изготовление БПО и разработка эксплуатационной документации.

Кроме этого, при проектировании к космическим аппаратам предъявляются следующие требования:

- обеспечение долговременного функционирования спутника на заявленном уровне качества;
- оперативность создания, восстановления и восполнения функционирования КА.

Выполнение перечисленных требований зависит от качества БПО, технологии его создания и сопровождения, возможностей инструментальных средств разработки и сопровождения [3].

### **1.3 Подходы к проектированию БПО**

На предприятии ОАО «Информационные спутниковые системы» имени академика М.Ф. Решетнёва» (г. Железногорск, Красноярский край) на этапах создания БПО спутников применяется технологический комплекс производства программ БПО (ТКПП БПО), представляющий собой совокупность АРМ пользователей (проектантов, программистов, сборщиков БПО), участвующих в разработке БПО, и серверов средств и архивов ТКПП БПО, объединенных локальной сетью предприятия.

В настоящее время в составе ТКПП БПО используются следующие средства:

- кросс-система программирования Модуля-2 (КСП М2);
- система программирования команд управления (СП ИСПОЛКОМ);
- система программирования макропрограмм интегрального управления (СП РЕАЛ);
- наземный отладочный комплекс БПО (НОК БПО);
- система изготовления программ БПО (СИП БПО);
- автоматизированная система сопровождения программ, изделий и документов БПО (АСПИД).

Далее показано, с помощью каких средств решается весь комплекс задач по разработке и сопровождению БПО спутников (Рисунок 3):

- определение требований к ПО систем и БПО в целом – АСПИД;
- архитектурное проектирование ПО систем и БПО в целом – АСПИД;
- детальное проектирование и программирование программ и данных – КСП М2, СП ИСПОЛКОМ, СП РЕАЛ;

- автономное тестирование программ – КСП М2;
- сборка и изготовление БПО – СИП БПО;
- системное тестирование ПО систем и БПО в целом – НОК БПО;
- сопровождение архивов и баз данных БПО – АСПИД, СИП;
- управление работами, конфигурацией и качеством – АСПИД.



Рисунок 3 – Составляющие технологии разработки БПО.

**КСП М2** позволяет создавать БПО для трех бортовых компьютеров. Она позволяет заимствовать и переносить изменения в ПО систем спутника на спутники с отличающимися бортовыми компьютерами, в том числе, когда спутник уже находится в эксплуатации.

**СП ИСПОЛКОМ** предназначена для создания блоков команд управления спутником.

**СП РЕАЛ** предназначена для создания макропрограмм интегрального управления спутником,

**СИП БПО** осуществляет автоматизацию процесса сборки и изготовления БПО не только для системного тестирования, но и для перепрошивки БПО эксплуатируемых спутников.

**НОК БПО** предназначен для выполнения системного тестирования реального БПО спутника путем создания для каждого спутника собственной программной модели.

**АСПИД** автоматизирует работы по управлению конфигурацией БПО – в том числе процедуры управления работами, объектами и проблемами БПО, а также создание и использование электронного архива сопровождения БПО [4].

Определение объектов разработки и проектирования БПО основывается на архитектурном расслоении БПО (Рисунок 4), что позволяет выделить набор компонентов, слоев и подсистем БПО, привязать их функционально к подсистемам изделия и структурно объединить набором стандартных интерфейсов и слоев.

При проектировании и разработке БПО выделяются следующие конфигурационные единицы:

- компонент ПО подсистемы спутника;
- сборка ПО подсистемы;
- выпуск БПО.

Структурная декомпозиция БПО (Рисунок 4) в процессе его разработки и сопровождения в модели управления объектами и работами (Рисунок 5) выделяет следующие разновидности объектов:

- проект программы БПО подсистемы;
- проект БПО;
- проект ПО подсистемы;
- проект макропрограмм интегрального управления;
- проект блоков команд управления ПО подсистемы;
- сборка ПО подсистемы изделия;

- сборка БПО изделия;
- выпуск БПО изделия.

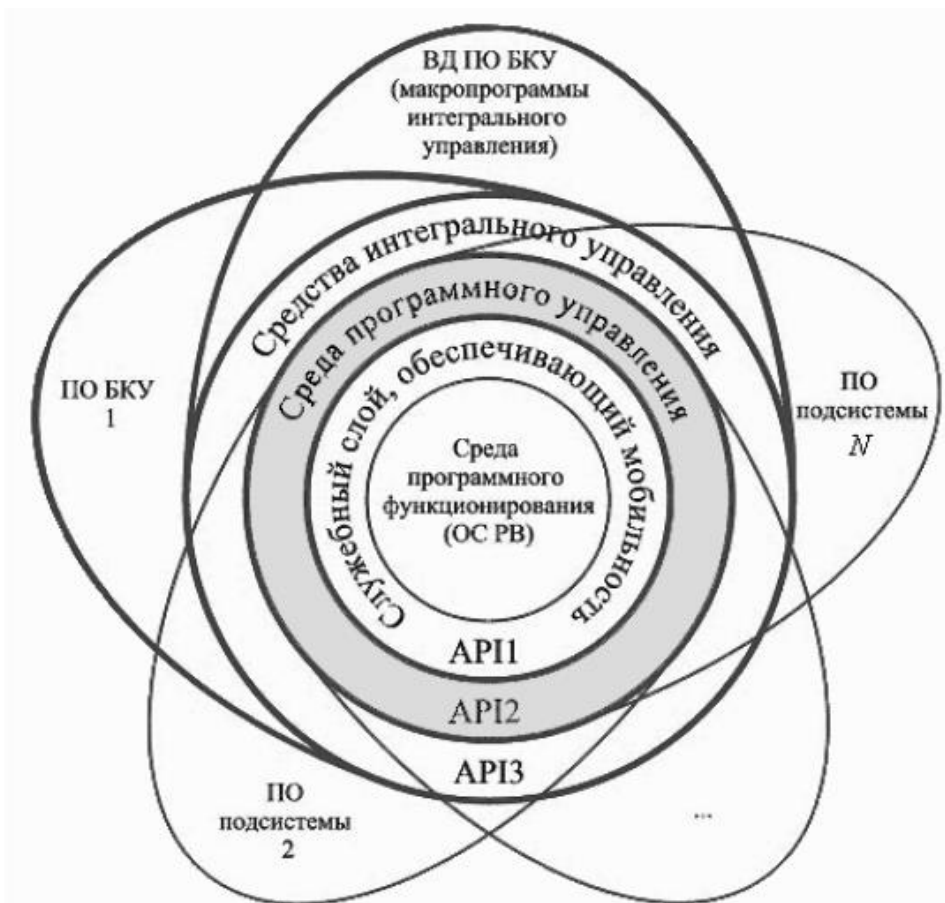


Рисунок 4 – Структурная декомпозиция БПО.

Конфигурационные единицы БПО представлены в виде программного проекта, представляющего собой электронную папку специальной структуры, определяющей версию объекта и хранящей всю необходимую информацию как для разработки, так и для сопровождения объекта.

**Компонент ПО подсистемы** включает в себя:

- документацию;
- исходные модули программы;
- пакеты для проведения автономной отладки программы;
- результаты отладки;
- результаты работы системы измерения;
- другие файлы, необходимые для разработки или доработки

компонента.

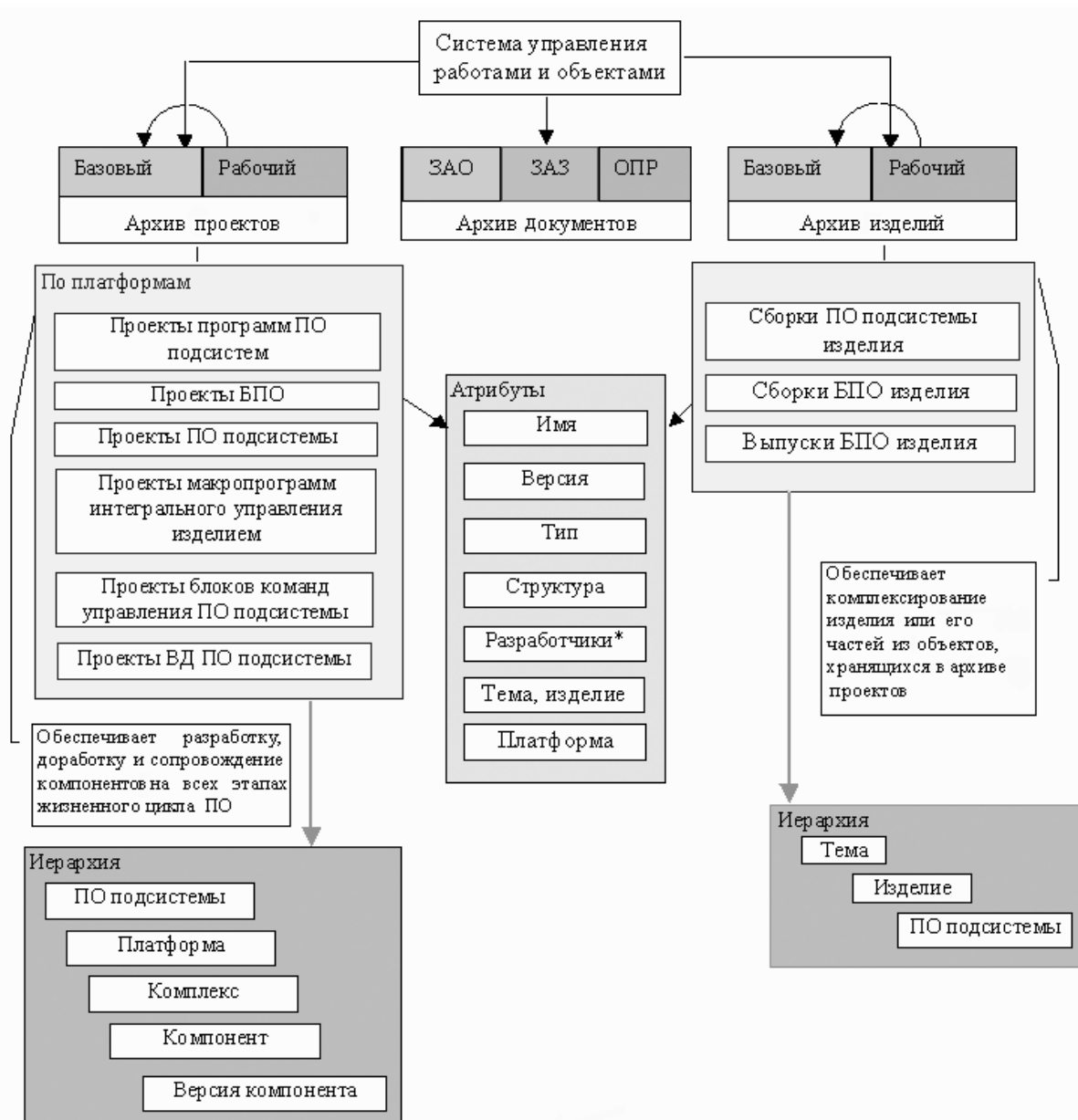


Рисунок 5 – Модель управления объектами и работами.

Компонент разрабатывается в рамках изделия в качестве объекта, унифицированного по соглашениям, интерфейсам, языку программирования и библиотекам типов данных, что позволяет ему быть пригодным для заимствования в другие изделия. Средства разработки компонента ПО системы, т.е. кросс-система программирования на языке Модуля-2, определяет структуру компонента ПО системы.

**Сборка ПО подсистемы** содержит исходные модули программ ПО подсистемы и все необходимые составляющие для проведения сборки.



Название сборки ПО подсистемы включает в себя тему, изделие и номер изменения сборки.

**Выпуск БПО изделия** содержит:

- сборку БПО изделия, которая проводится для отработки и включает в себя сборки ПО подсистем и входные данные ПО БКУ;
- массивы КПИ изделия;
- материалы прошивки изделия;
- спецификацию.

Название выпуска включает тему, изделие и номер изменения сборки [5].

Качество разрабатываемого БПО обеспечивается выполнением следующих принципов:

- качество компонентов БПО;
- качество управления конфигурацией БПО;
- качество верификации и качество подтверждения БПО.

**Качество компонентов** гарантируется внедрением в процесс разработки следующих пунктов:

1. Определение архитектурного разделения и расслоения БПО спутника, включающее в себя выделение типового набора компонентов, слоев и подсистем БПО, сопоставленных с подсистемами изделия и объединенных стандартными интерфейсами и слоями.

2. Составление стандартного сценария детального проектирования и разработки компонента, определяемого в рамках стандартного документа описания программы, происходящее параллельно с проектированием набора тестов и кодированием программы.

3. Программирование с использованием сильно структурированного языка со строгой типизацией данных Модуля-2.

4. Использование методов и средств измерения статистических и динамических характеристик программы, позволяющих в автоматическом

режиме оценивать достигнутые эксплуатационные характеристики программы (например, объем памяти, время выполнения или размер стека), а также выполнять оценку добротности программы (например, оценку уровня сложности, уровня комментируемости или выполнения системных соглашений).

5. Использование методов документирования процессов тестирования и применение соответствующих программных средств, обеспечивающих автоматическое определение и документирование достигнутого уровня критериев тестирования, а также автоматизацию процессов регрессивного тестирования компонентов.

6. Использование проблемно-ориентированных командных языков и систем для программирования команд управления конкретным изделием и логики функционирования изделия в целом.

7. Повторное использование ранее созданного и уже отработанного на должном уровне бортового и инструментального ПО, возможное за счет постоянства пользовательских и программных интерфейсов БПО и инструментальных средств, надежного управления конфигурацией программного обеспечения и архивами сопровождения.

**Качество управления конфигурацией БПО** обеспечивается использованием методов и средств идентификации и управления объектами, работами и проблемами ПО, процедурами и средствами, автоматизирующими процессы управления конфигурацией, а также формы и процедуры электронного документооборота.

Управлению подлежат следующие электронные архивы:

- архив сопровождения работ и проблем;
- архив проектов унифицированных программ;
- архивы БПО изделий.

Каждый объект разработки первоначально описывается как компонент, соотносящийся с архитектурным проектом в рамках конкретного изделия. Информацию о компонентах и график их разработки переносят в архив

сопровождения работ и проблем изделия после утверждения первой редакции архитектурного проекта.

Окончание работ по разработке компонента наступает после передачи всей документации сопровождения, оформленной как проект программы, в архив проектов унифицированных программ. Сам разработанный компонент, настроенный на конкретное применение, передается в архив БПО изделия. Интеграция БПО и его изготовление осуществляются в рамках данного архива.

С момента занесения в архив, последующие выявленные ошибки в компоненте должны исправляться на основании отчетов о проблеме в программном обеспечении. В случае необходимости, изменения в связи с уточнением требований осуществляются по запросам-отчетам, уточняющим требования архитектурного проекта и определяющим полный цикл работ, требующих повторения.

**Качество верификации и подтверждения БПО** обеспечивается использованием следующих методов и средств:

- верификация требований;
- верификация архитектурного проекта;
- верификация средств разработки и тестирования БПО;
- верификация детального проекта;
- верификация кода;
- автономное тестирование;
- тестирование интеграции БПО;
- системное тестирование БПО;
- квалификационное тестирование ПО систем и БПО в целом в составе подсистем на инженерных моделях;
- квалификационное тестирование ПО систем и БПО в целом в составе подсистем на летном изделии.

Главными аспектами в обеспечении верификации и подтверждения ПО подсистем и БПО в целом являются:

- проведение автономного и системного тестирования БПО без использования реальной аппаратуры благодаря использованию в составе средств тестирования верифицированных программных моделей бортового процессора, моделей устройств бортового комплекса управления и моделей поведения подсистем спутника;

- проведение системного тестирования непосредственно проектантами логики функционирования подсистем изделия и изделия в целом, возможное благодаря удобству пользовательского интерфейса средств тестирования;

- полное сохранение пользовательских интерфейсов средств тестирования при использовании моделей на разных вычислительных платформах;

- расширение диалоговых средств тестирования и отладки, позволяющих проектировать и программировать тесты по той же технологии, что и тестируемые программы БПО;

- организация системного тестирования в соответствии с архитектурной декомпозицией БПО.

**Средства разработки**, реализующие вышеперечисленные методы, представлены технологическим комплексом производства программ БПО (ТКПП), выполняющим функции разработки, автономного тестирования и сопровождения компонентов БПО; и наземным отладочным комплексом (НОК) БПО, применяющимся для интеграционного и системного тестирования БПО, а также изготовления и сопровождения БПО конкретных изделий.

Описание составляющих ТКПП БПО было приведено выше.

Наземный отладочный комплекс обязательно содержит БПО конкретного изделия, программную модель целевого БЦВК, модели

поведения систем изделия, средства организации модельного вычислительного комплекса и средства отладки.

Архив БПО изделий и средства изготовления носителей БПО также входят в состав НОК.

НОК является исключительно программным средством, которое функционирует в той же инструментальной среде, что и сама среда разработки.

Проблемно-ориентированный пользовательский интерфейс НОК позволяет проводить системное тестирование ПО систем и БПО в целом проектантами систем изделия.

Интерфейс НОК унифицирован с интерфейсом средств заводских испытаний и средств управления при эксплуатации изделий [3, 6].

#### **1.4 Постановка задачи**

В состав электронного архива БПО входят архивы программ БПО, изделий БПО и архивы распорядительного документооборота. Используя принцип «аналогии», архивы программ и изделий позволяют автоматизировать даже первые этапы разработки БПО – этапы определения требований и архитектурного проектирования. Для полной автоматизации этих процессов в ОАО «ИСС» предназначены работы по созданию и внедрению в состав ТКПП БПО САПР БПО (Рисунок 6) [7].

Процесс проектирования БПО упрощается благодаря возможности повторного воспроизведения информации из документации по уже выполненным проектам, а также основывается на сводных таблицах и вспомогательных документах, используемых проектантом и разрабатываемых исходя из его личного опыта.



Рисунок 6 – Роль САПР БПО на этапах разработки БПО спутников.

В процессе архитектурного формирования БПО изделия участвуют более 40 типов программных объектов, а их суммарное количество на одно изделие может достигать миллиона. Из-за тесных связей объектов друг с другом и, изменение одного объекта оказывает влияние на изделие в целом. Иерархические зависимости объектов встречаются не только в рамках одного изделия, при разработке БПО часто возникают зависимости между самими изделиями. Эти зависимости возникают при заимствовании объектов или даже целых подсистем из других изделий (при выполнении условия их однотипности или совместимости). Из-за зависимостей такого типа возникает ряд проблем – при возникновении ошибки или внесении изменений в часть объектов некоторого изделия необходимо отследить изделия, в которых было использовано и модифицировано то же самое решение, т.е. существует односторонняя зависимость между базовыми и модифицированными изделиями [8].

Для обеспечения выполнения требований к построению единого информационного пространства организации процесса разработки и

сопровождения бортового программного обеспечения САПР БПО предоставляет функции:

- структурированного представления, хранения, извлечения и изменения данных об изделиях и составляющих изделия (требованиях, наборах компонент, интерфейсах, параметрах и др.);
- проектирования на уровне изделия, системы и программного компонента;
- сопоставления взаимосвязей требований архитектуры БПО конкретного спутника с библиотеками унифицированных компонент;
- управления зависимостями при внесении изменений в архитектуру БПО;
- автоматизированного формирования отчетных статистических данных;
- автоматизированного выпуска проектной документации БПО [9].

Кроме вышеперечисленных возможностей, требуется провести расширение функционала САПР БПО с целью использования САПР БПО на этапах тестирования и верификации БПО космических аппаратов для выполнения задач:

- хранения, извлечения, добавления и модификации тестов на изделие (космический аппарат);
- хранения, извлечения, добавления и модификации тестов на систему (являющейся частью БПО космического аппарата);
- отображения матрицы трассируемости функциональных требований в тесты.

## 2 РЕАЛИЗАЦИЯ МЕХАНИЗМА ТЕСТОВ

### 2.1 Описание архитектуры САПР БПО

САПР БПО реализована согласно трёхзвенной архитектуре, в которой в системе выделяются три следующих компонента: клиентское приложение, сервер приложений и сервер базы данных (сервер БД).



Рисунок 7 – Архитектура компонентов САПР БПО.

Более подробно развёртывание САПР БПО описано ниже.

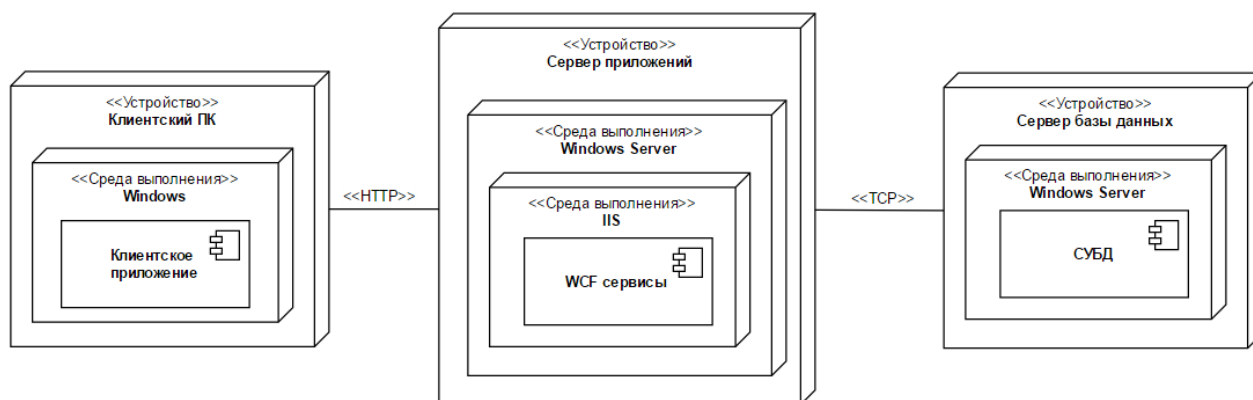


Рисунок 8 – Диаграмма развёртывания компонентов САПР БПО.

Клиентское приложение является настольным приложением для операционной системы Windows. Клиентское приложение осуществляет взаимодействие с сервером приложений по протоколу HTTP.

Сервер приложений предназначен для реализации основной бизнес-логики работы с данными. Сервером приложений может служить виртуальная или физическая машина с операционной системой Windows или Windows Server. Для развёртывания WCF сервисов, осуществляющих бизнес-логику управления данными, необходим веб-сервер IIS (Internet Information Services).

Сервер базы данных осуществляет хранение и доступ к данным. В качестве сервера базы данных может использоваться физическая или виртуальная машина с операционной системой Windows или Windows Server.



В роли СУБД служит MS SQL Server Express 2014. Сервер приложений подключается к серверу базы данных по протоколу TCP.

САПР БПО разделена на физические модули, показанные ниже.

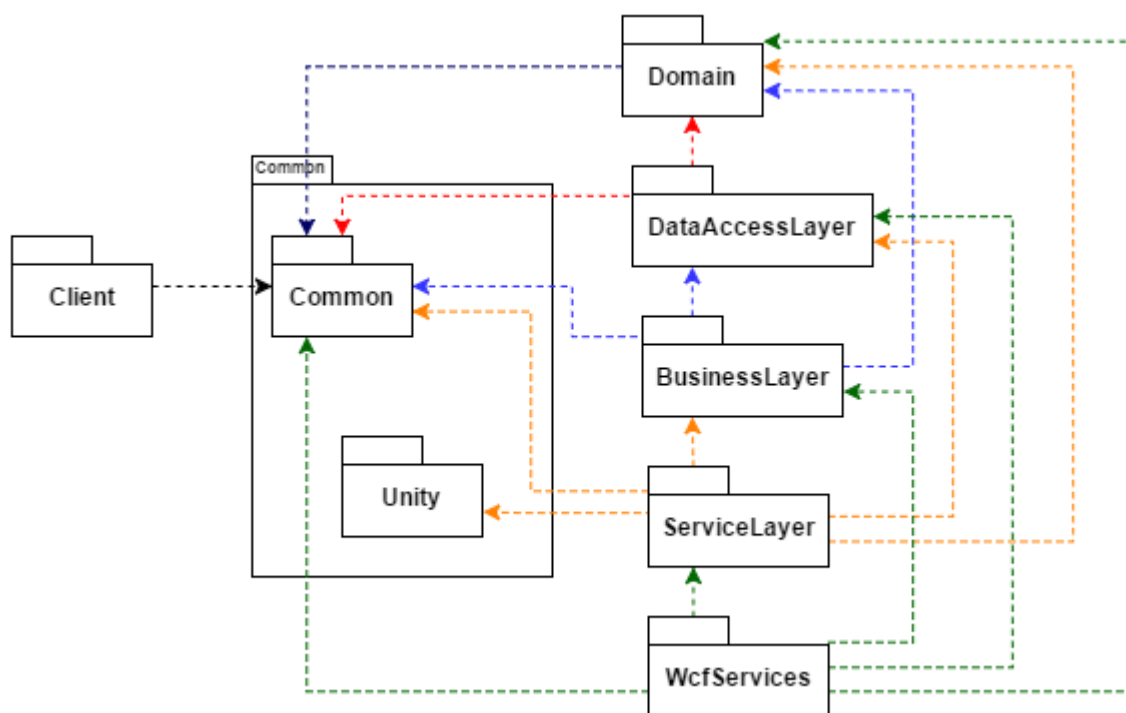


Рисунок 9 – Диаграмма пакетов САПР БПО.

Рассмотрим по отдельности каждый модуль.

Модуль Client содержит исполняемые файлы клиентского приложения.

Модуль Common включает в себя подмодули Common и Unity.

Подмодуль Common содержит описание DTO (Data Transfer Object, [10]) промежуточных сущностей, предназначенных для передачи данных из БД клиентскому приложению, и общие интерфейсы (контракты сервисов), используемые как клиентом, так и сервером. Подмодуль Unity содержит реализацию контейнера для инверсии управления Unity.

Модуль Domain содержит сущности предметной области, на основе которых формируется база данных.

Модуль DataAccessLayer содержит контекст доступа к базе данных, реализации репозитория, конфигурации сущностей и миграции.

Модуль BusinessLayer содержит реализацию всех операций сервера, объявленных в контракте сервисов.

Модуль `ServiceLayer` содержит конфигурацию точек доступа и каналов связи сервера, инициализацию фабрики контекста базы данных.

Модуль `WcfServices` содержит описание и конфигурацию набора сервисов для развертывания на веб-сервере IIS.

### **2.1.1 Сервер базы данных**

Сервер базы данных отвечает за хранение данных (в том числе их резервирование и возможное восстановление), в качестве СУБД выбран `Microsoft SQL Server Express 2014`.

Реляционный подход к хранению информации в базе данных, предлагаемый данной СУБД, является наиболее надёжным способом структурированно и эффективно управлять информацией об объектах системы.

Для сервера базы данных возможно выделить отдельную виртуальную машину и разместить её во внутренней сети организации, что позволит обеспечить безопасность, ограничив доступ к серверу базы данных извне. Кроме этого, это позволит резервировать состояние всей виртуальной машины, а не только содержимого базы данных.

В САПР БПО используется технология ORM (`Object-Relational Mapping`, [11]), что переносит проектирование базы данных на слой выше, от физической структуры базы данных к предметным классам в программном коде. Из-за этого необходимость взаимодействия разработчика напрямую с базой данных минимальна, поскольку изменение структуры таблиц осуществляется с помощью механизма миграций.

### **2.1.2 Сервер приложений**

Сервер приложений представляет собой набор WCF сервисов платформы .NET и реализует основную часть бизнес-логики.

Для сервера приложений используется многослойная архитектура, где каждый слой, расположенный выше, обладает информацией о том, как обратиться к нижестоящему слою. Доступ к слоям осуществляется только в направлении сверху вниз (Рисунок 10).



Рисунок 10 – Многослойная архитектура сервера приложений.

Слою сервисов на диаграмме пакетов (Рисунок 9) соответствуют модули `WcfServices` и `ServiceLayer`. Слой сервисов обеспечивает соединение и взаимодействие клиентского приложения с сервером приложений. Параметры развертывания каждого сервиса описываются в файле `Web.config`.

Слой бизнес-логики соответствует модуль `BusinessLayer`, содержащий реализацию операций сервера. Контракты сервисов, используемые клиентским приложением и сервером приложений и реализованные на слое бизнес-логики, объявлены в модуле `Common`.

Слою доступа к данным соответствует модуль `DataAccessLayer`, обеспечивающий взаимодействие с БД и управление физической структурой БД (с точки зрения подхода `Code First` [12]): контекст доступа к БД, реализации репозитория, конфигурации сущностей и добавление миграций.

Слою модели данных соответствует модуль `Domain`, описывающий классы предметной области.

С точки зрения логики работы программы, основные объекты системы наследуют свое поведение от абстрактного класса копируемой сущности `CopiedEntity` (Рисунок 11).

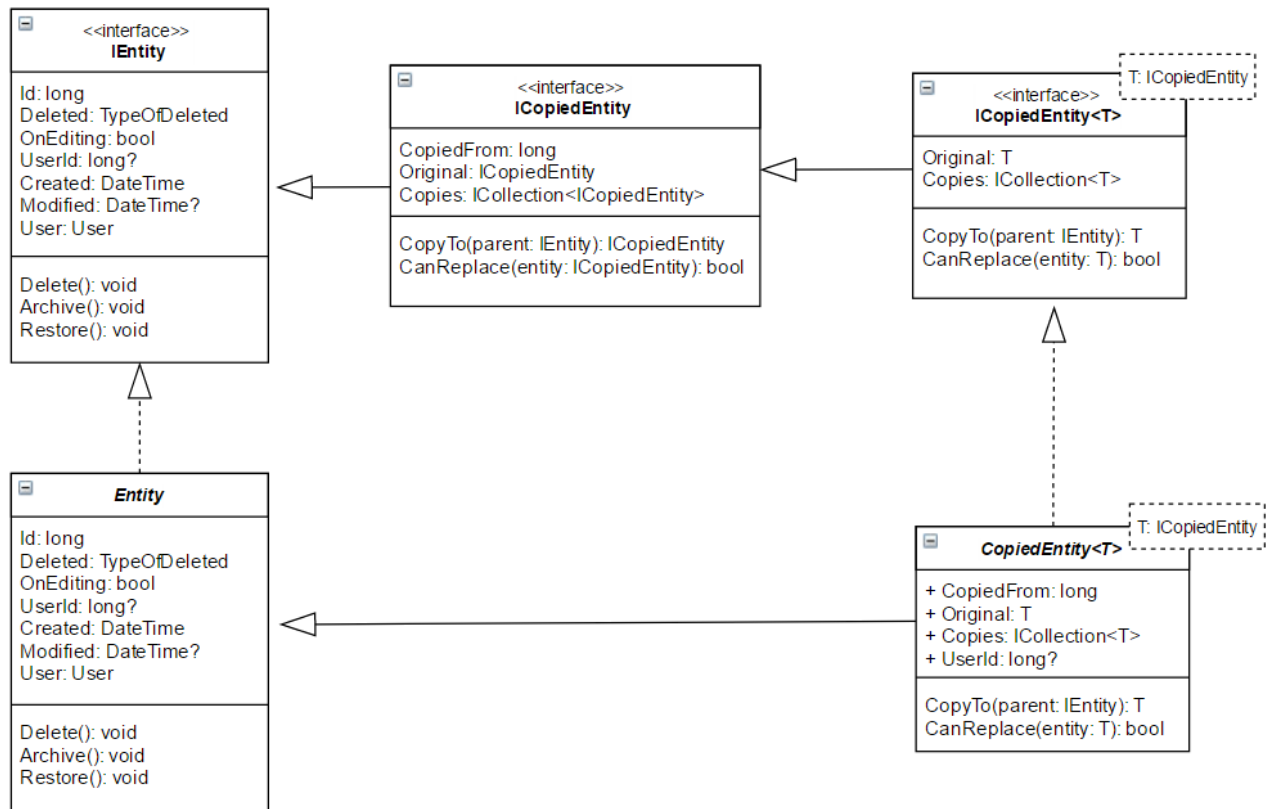


Рисунок 11 – Диаграмма классов для копируемой сущности.

Для реализации сервисов управления объектами системы предусмотрен базовый абстрактный класс `GenericService`, расширяющий возможности родительского класса `CrudService`. Класс `CrudService` содержит реализацию базовых операций добавления, чтения, редактирования и удаления сущностей в базе данных. Поскольку абстрактный класс `GenericService` наследует функционал класса `CrudService`, то и сервисы, наследующие поведение `GenericService`, также наследуют реализации базовых операций работы с БД (Рисунок 12, Рисунок 13). При необходимости расширения базового функционала, дополнительные операции приводятся в контракте сервиса.

Данные отношения между классами проиллюстрированы для сущности «Платформа» (Рисунок 14, Рисунок 15).

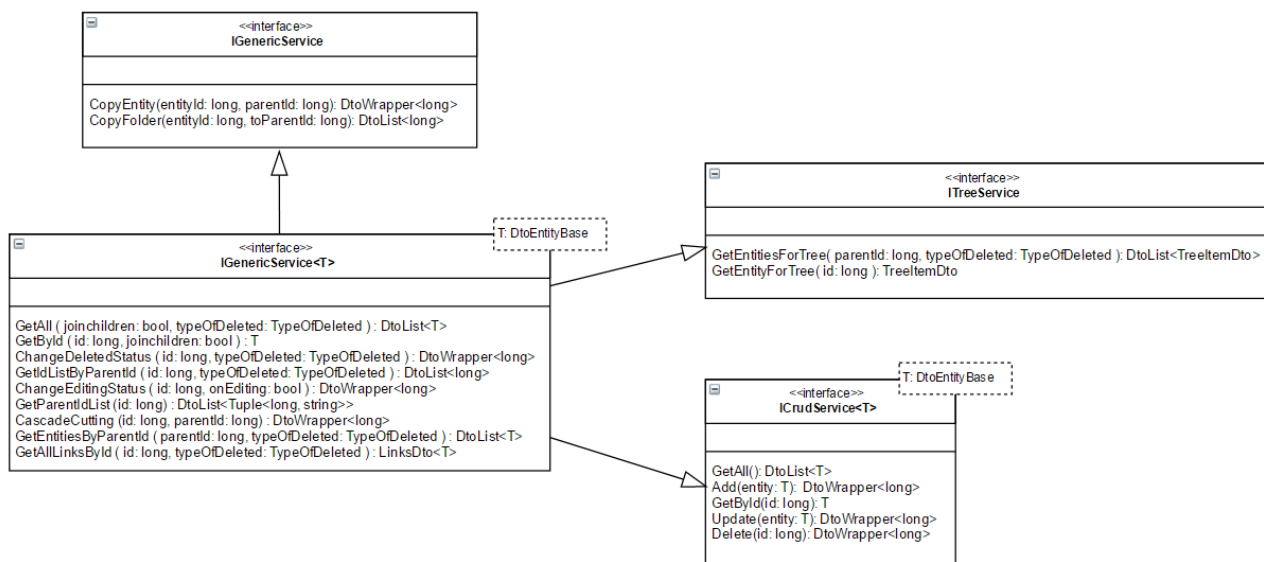


Рисунок 12 – Описание интерфейса IGenericService.

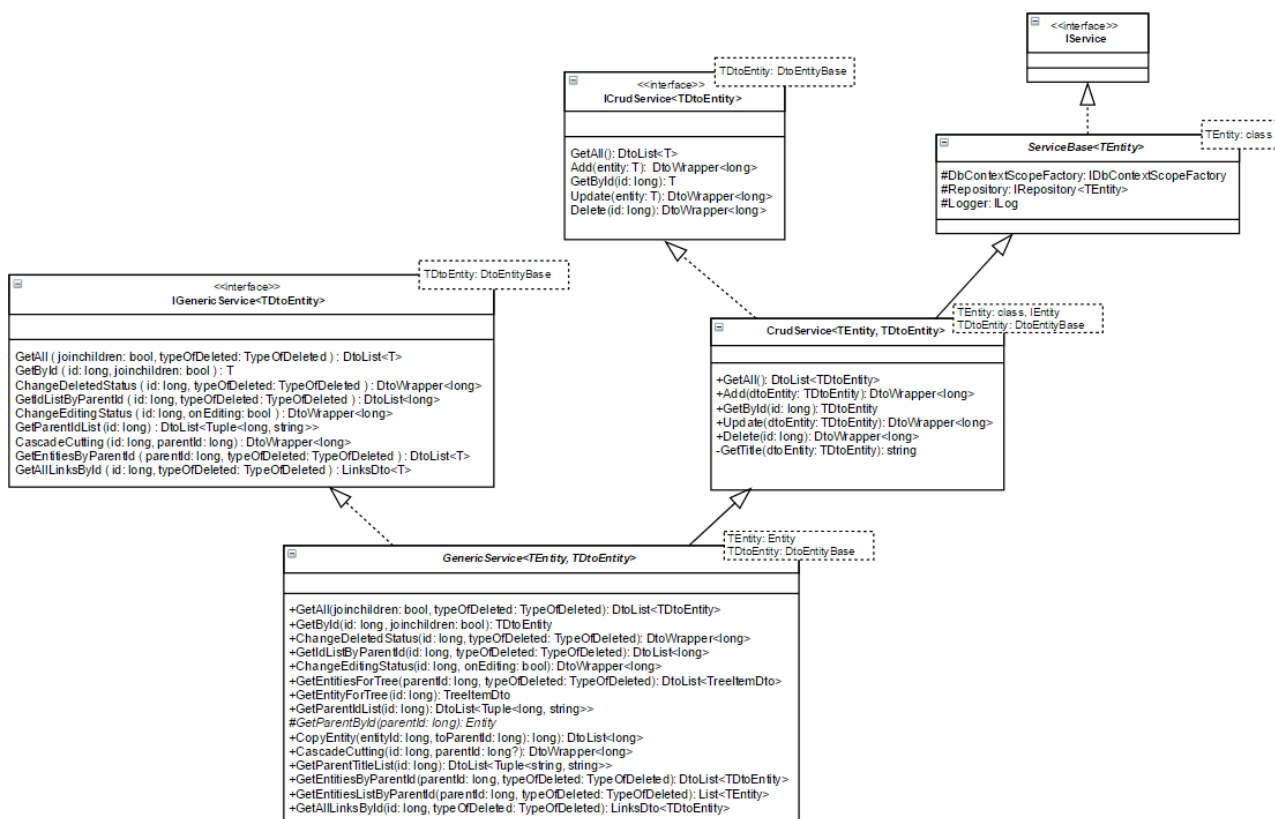


Рисунок 13 – Диаграмма классов для базового сервиса GenericService.

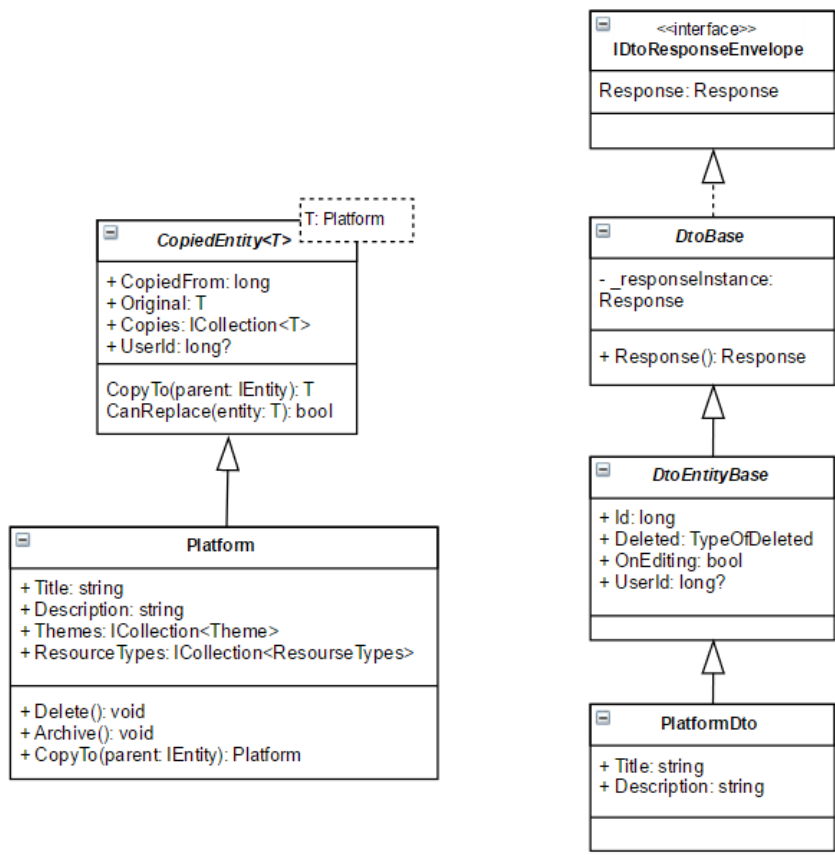


Рисунок 14 – Диаграмма классов для сущности «Платформа».

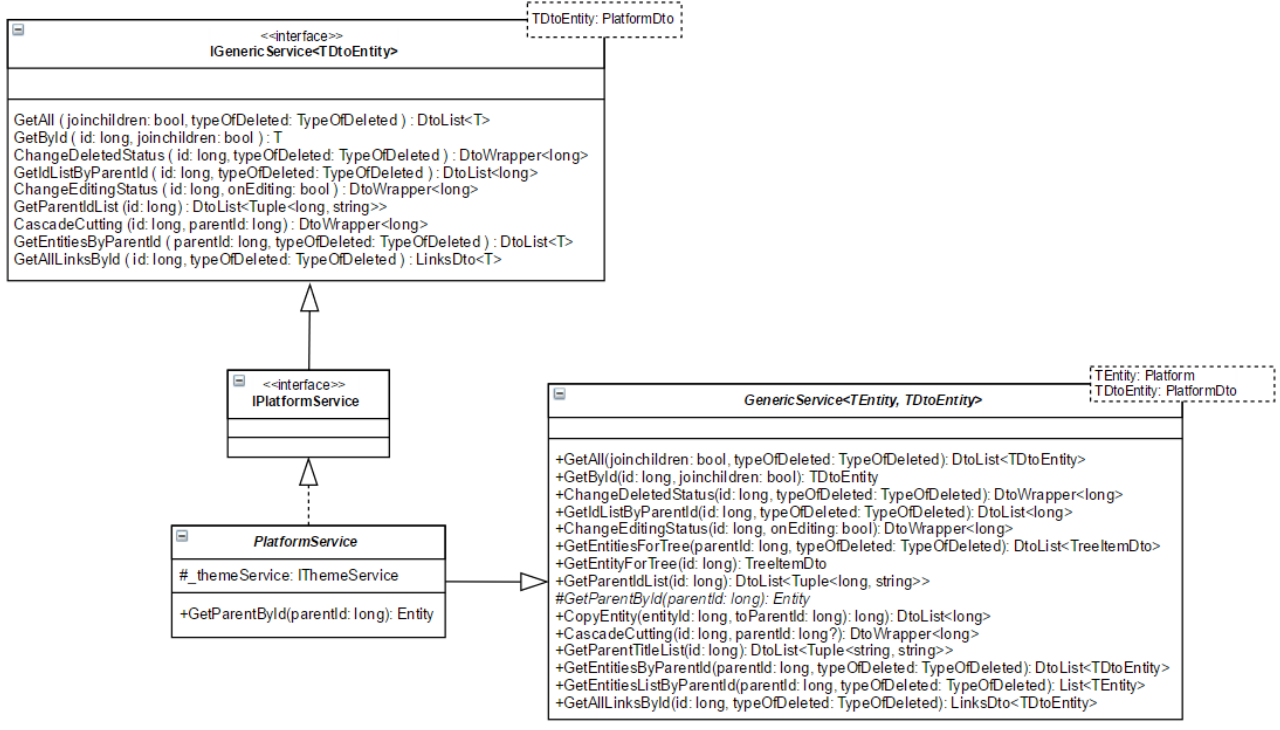


Рисунок 15 – Диаграмма классов для сервиса PlatformService.

### 2.1.3 Клиентское приложение

Клиентское приложение разработано на платформе .NET и использованием технологии WPF для графического интерфейса.

Клиентское приложение состоит из следующих модулей (Рисунок 16).

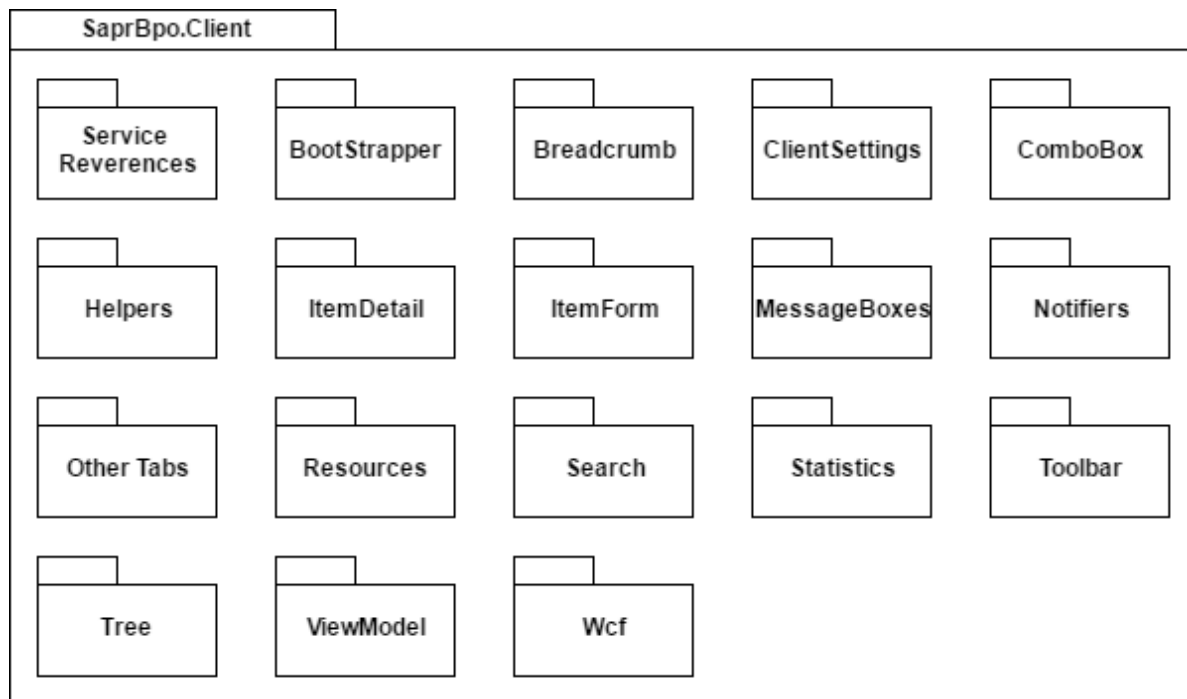


Рисунок 16 – Диаграмма пакетов клиентского приложения.

Рассмотрим по отдельности каждый модуль клиентского приложения.

Модуль `ServiceReferences` предназначен для хранения ссылок на внешние службы и сервисы.

Модуль `BootStrapper` содержит класс `ClientBootStrapper`, осуществляющий запуск приложения и обработку исключений.

Модуль `Breadcrumb` соответствует адресной строке приложения. Модуль содержит классы управления адресной строкой, отражающей положение элемента системы в дереве объектов системы.

Модуль `ClientSettings` реализует набор классов для настроек клиентского приложения, включая окна настроек, отображаемые пользователю.

Модуль `ComboBox` содержит набор классов для создания выпадающих списков, отображающихся на карточках объектов.

Модуль `Helpers` содержит набор второстепенных классов, используемых в приложении, таких как: атрибуты, конвертеры, расширения, сообщения, поведения и перечисления.

Модуль `ItemDetail` содержит набор классов для отображения базовых карточек, используемых для отображения сущностей системы.

Модуль `ItemForm` содержит набор пользовательских интерфейсов и реализацию логики для добавления новых сущностей в систему.

Модуль `MessageBoxes` содержит набор классов, реализующих модальные окна в системе.

Модуль `Notifiers` описывает модальные окна в которых отображаются сообщения с ошибками, предупреждениями и исключениями.

Модуль `Other Tabs` содержит набор классов пользовательских интерфейсов и логики для отображения дополнительных вкладок на карточке сущности.

Модуль `Resources` содержит изображения, сгруппированные по назначению, используемые в приложении.

Модуль `Search` содержит набор классов, описывающих интерфейс и логику работы поиска сущностей.

Модуль `Statistics` содержит набор классов, описывающих интерфейс модального окна для генерации файлов со статистикой.

Модуль `Toolbar` реализует интерфейс панели инструментов приложения.

Модуль `Tree` содержит набор классов, описывающих интерфейс, логику работы дерева объектов, узлы дерева (папки, объекты).

Модуль `ViewModel` содержит логику работы главного окна программы.

Модуль `Wcf` содержит набор классов для вызова удаленных сервисов, предоставляемых сервером приложений.

Параметры подключения клиентского приложения к серверу приложений описываются в файле `App.config`.



### 2.1.3.1 Подключение к WCF-сервисам

Подробные настройки вызова удаленных сервисов содержатся в модуле Wcf.

Вызов удаленных процедур осуществляется с использованием базовых классов `WcfAdapterBase`, `WcfAdapterBaseWithCrud`, `WcfAdapterBaseWithGeneric`, `WcfHandler`, `WcfProxy`, принадлежащих подмодулю `BaseWcf`.

Сопоставление операций сервисов, объявленных на сервере и доступных клиентскому приложению, осуществляется в подмодуле `PartialWcf`, например, путем наследования от класса `WcfAdapterBaseWithGeneric` и реализации соответствующего интерфейса контракта сервиса. С помощью функции `ExecuteCommand` выполняется вызов удаленной функции на сервере.

Далее показан фрагмент кода, относящийся к реализации класса `RequirementServiceProxy`, иллюстрирующий данный подход.

```
public class RequirementServiceProxy : WcfAdapterBaseWithGeneric<IRequirementService, RequirementDto>, IRequirementService
{
    public DtoList<RequirementDto> GetRequirementsBySystId(long systId, TypeOfDeleted typeOfDeleted = TypeOfDeleted.Actual)
    {
        return ExecuteCommand(proxy => proxy.GetRequirementsBySystId(systId, typeOfDeleted));
    }
    ...
}
```

Программное сопоставление между интерфейсами сервисов и реализованными прокси-классами клиентского приложения осуществляется в конструкторе класса `SimpleServiceLocator`, относящегося к модулю Wcf.

Например, для сервиса управления функциональными требованиями строка объявления имеет вид:

```
SimpleIoc.Default.Register<IRequirementService, RequirementServiceProxy>();
```

Дальнее обращение к операциям из кода происходит с использованием механизма инверсии управления.

Например, далее приведен фрагмент кода, иллюстрирующий метод `GetGridService()`, использующий механизм инверсии управления.

```
protected override IGenericService<RequirementDto> GetGridService()
{
    return SimpleServiceLocator.GetService<IRequirementService>();
}
```

### 2.1.3.2 Дерево объектов системы

Управление и отображение сущностей системы осуществляется с помощью иерархического дерева объектов системы, реализация которого располагается в модуле `Tree`.

Далее показаны базовые классы дерева объектов системы (Рисунок 17) и иерархия объектов системы (Рисунок 18).

В иерархической структуре дерева выделены два типа элементов: узлы и папки. Узлам дерева соответствуют объекты системы. Папка (директория, каталог) является контейнером для узлов дерева, принадлежащих к одному типу (например, программы) или к схожим типам (например, разные типы ресурсов системы).

Для добавления нового узла или папки в дерево объектов системы, в первую очередь, необходимо определить его место в иерархии объектов (Рисунок 18) и затем реализовать соответствующие классы, выполнив наследование от одного из базовых классов элементов дерева.

Базовыми классами элементов дерева являются:

- `TreeFolder` – каталог, содержащий внутри себя дочерние элементы;
- `FixedTreeFolder` – каталог, не содержащий дочерних элементов;
- `TreeNode` – узел дерева, который может содержать дочерние элементы;
- `FixedTreeNode` – узел дерева, который не может содержать дочерние элементы.

Кроме этого, класс `TreeNodeLocator` хранит сопоставление классов объектов передачи данных с узлами дерева. Например, для сущности «Функциональное требование» выполняется соответствующее сопоставление:

```
Register<RequirementDto>(typeof (RequirementTreeNode));
```

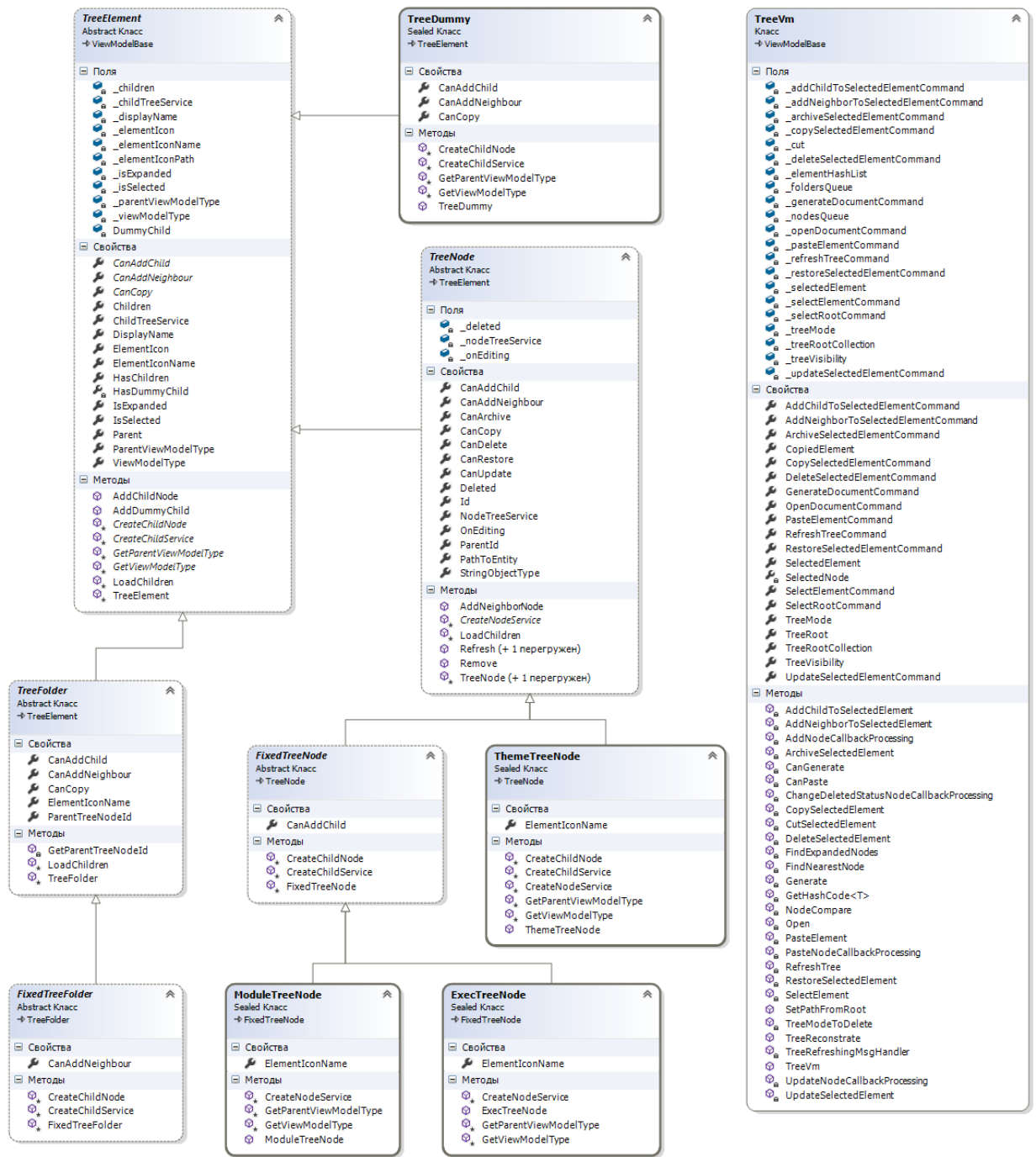


Рисунок 17 – Диаграмма классов, описывающая базовые узлы дерева.

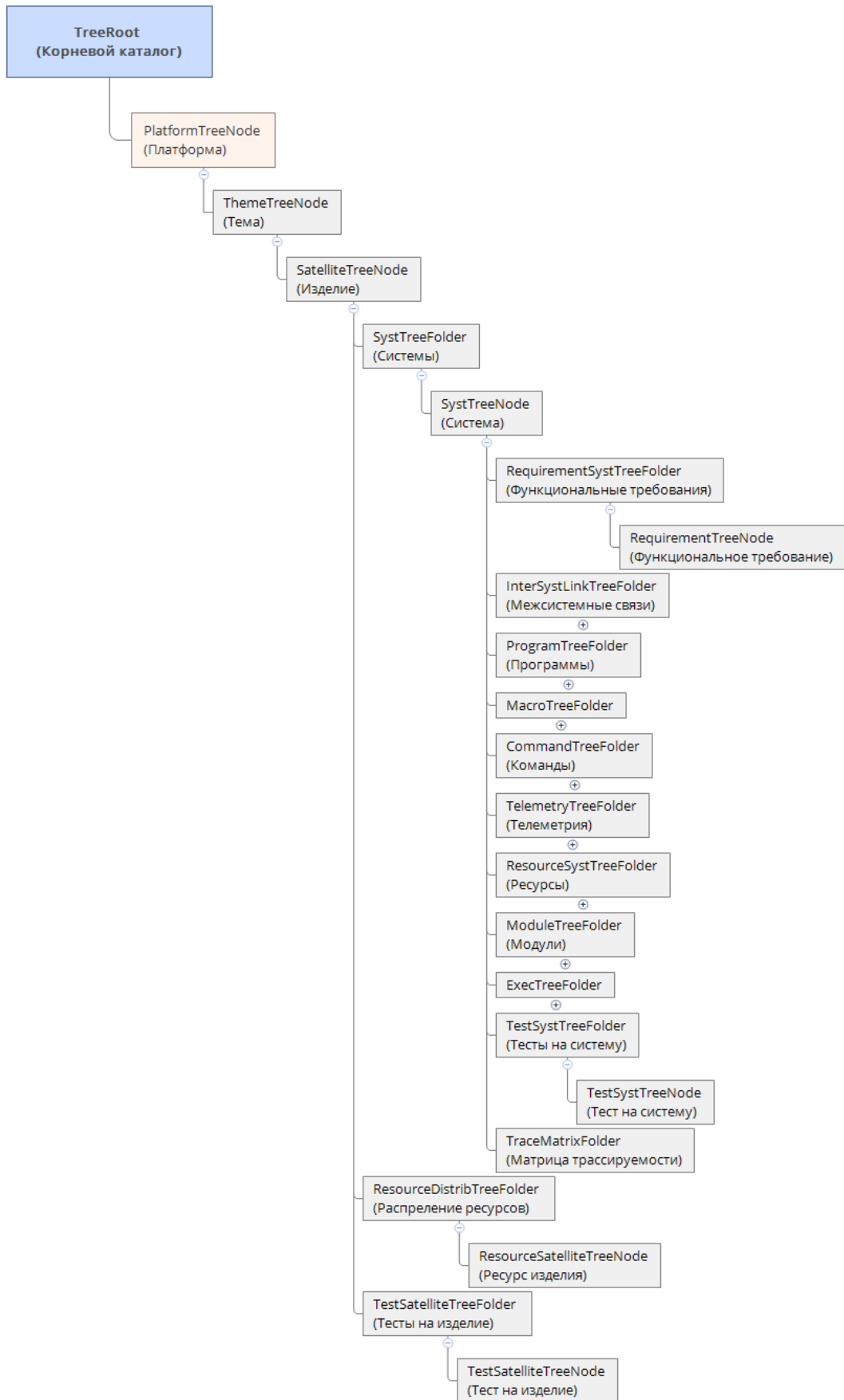


Рисунок 18 – Иерархическая структура дерева объектов.

Реализация узла дерева далее показана на примере узла «Функциональное требование» и позволяет получать информацию как о текущем узле дерева, так и о родительском элементе:

```
public class RequirementTreeNode : FixedTreeNode
{
    public RequirementTreeNode(TreeElement parent, TreeItemDto item)
        : base(parent, item, false)
    {
    }
    #region ElementIconName
    public override string ElementIconName => "Requirements2.png";
    #endregion
    protected override Type GetViewModelType()
    {
        return typeof (RequirementVm);
    }

    protected override Type GetParentViewModelType()
    {
        return typeof (RequirementSystVm);
    }

    protected override ITreeService CreateNodeService()
    {
        return SimpleServiceLocator.GetService<IRequirementService>();
    }
}
```

Дерево объектов является основным управляющим элементом пользовательского интерфейса главного окна приложения (Рисунок 19), позволяющим осуществлять навигацию между объектами, их отображение, редактирование, добавление, копирование и удаление.

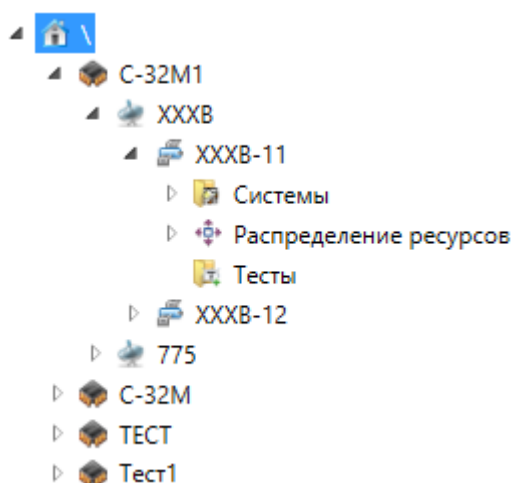


Рисунок 19 – Внешний вид дерева объектов.

Логика дерева объектов тесно связана с классами отображения элементов главного окна приложения.

### 2.1.3.3 Описание реализации оконных форм и вкладок системы.

Для реализации пользовательского интерфейса клиентского приложения используется технология WPF и фреймворк MVVMLight [13].

Шаблон MVVM подразумевает наличие трёх связанных составляющих:

- Model – класс, описывающий модель данных и используемый для привязки данных к элементам отображения;
- View – сам элемент отображения, являющийся отдельным окном или пользовательским элементом управления;
- ViewModel – класс, управляющий логикой поведения элемента отображения [14].

Для регистрации основных классов ViewModel в приложении используется класс ViewModelLocator, осуществляющий инверсию управления.

Регистрация класса MainViewModel показана ниже:

```
SimpleIoc.Default.Register<MainViewModel>();
```

Получение экземпляра MainViewModel при обращении к классу ViewModelLocator возможно, если объявлено следующее поле:

```
public static MainViewModel Main
{
    get { return ServiceLocator.Current.GetInstance<MainViewModel>(); }
}
```

Для пользовательских элементов управления сопоставление с классами ViewModel содержится в словаре ресурсов MainDataTemplateDictionary.xaml.

Пространство имен в разметке объявляется следующим образом (на примере функциональных требований):

```
xmlns:ifvm="clr-namespace:Saprbpo.Client.ItemForm.ViewModel"
xmlns:ifv="clr-namespace:Saprbpo.Client.ItemForm.View"
xmlns:reqv="clr-namespace:Saprbpo.Client.OtherTabs.Requirements.View"
xmlns:reqvm="clr-namespace:Saprbpo.Client.OtherTabs.Requirements.ViewModel"
```

Сопоставление форм отображения и логики отображения (на примере функциональных требований):

```

<DataTemplate DataType="{x:Type ifvm:RequirementVm}">
    <ifv:RequirementDetailView />
</DataTemplate>
<DataTemplate DataType="{x:Type reqvm:RequirementSystVm}">
    <ggv:EntityFolderWithDescriptionGridView />
</DataTemplate>
<DataTemplate DataType="{x:Type reqvm:RequirementImplObjectVm}">
    <reqv:RequirementImplObjectView />
</DataTemplate>
<DataTemplate DataType="{x:Type reqvm:RequirementAssignVm}">
    <reqv:RequirementAssignView />
</DataTemplate>
<DataTemplate DataType="{x:Type reqvm:RequirementTestVm}">
    <reqv:RequirementTestView />
</DataTemplate>

```

Таким образом, при вызове класса логики управления ViewModel, будет автоматически выполнено сопоставление с соответствующим элементом отображения (окном или пользовательским элементом управления). Это исключает необходимость принудительного вызова окон или пользовательских элементов в программном коде при переключении между ними, программного сопоставления классов ViewModel с элементами отображения.

Шаблоны карточек отображения находятся в модуле ItemDetails.

Сами формы отображения объектов системы расположены в модуле ItemForm.

Дополнительные вкладки, формы и элементы управления содержатся в модуле OtherTabs.

## 2.2 Модификация САПР БПО

Модификация САПР БПО в рамках добавления объектной сущности «Тест» связана со следующими изменениями:

- добавление программного класса «Тест» в описание классов предметной области, установка связей с другими сущностями и внесение соответствующих изменений в структуру таблиц БД;
- добавление и реализация сервиса управления тестами на сервере приложений;

- добавление и реализация вспомогательных классов, позволяющих передавать данные от сервера и отображать их в пользовательском интерфейсе клиентского приложения;
- подключение сервиса управления тестами в клиентском приложении;
- настройка элементов отображения тестов, реализация логики отображения и логики управления тестами в клиентском приложении.

### 2.2.1 Модификация программных классов предметной области и структуры таблиц БД

Тест – это последовательность выдачи управляющих воздействий и контроля результата, выполняемая с целью подтверждения корректности работы программных компонент, ПО систем или БПО в целом.

В САПР БПО сущность «Тест» может соотноситься с изделиями, системами или функциональными требованиями связями «многие-ко-многим». Кроме этого, возможна иерархическая зависимость между тестами –

Фрагмент БД, иллюстрирующий сущность «Тест» и её связи через вспомогательные таблицы RequirementTest, TestSyst, TestSatellite, приведён ниже (Рисунок 20).

Далее приведён фрагмент программного кода, описывающий программный класс Test (находящийся в модуле Domain, т.е. в слое модели данных).

```
public class Test : CopiedEntity<Test>
{
    [SuppressMessage("Microsoft.Usage",
"CA2214:DoNotCallOverridableMethodsInConstructors")]
    public Test()
    {
        Tests = new HashSet<Test>();
        Requirements = new HashSet<Requirement>();
        Satellites = new HashSet<Satellite>();
        Sysys = new HashSet<Syst>();
    }

    public string Title { get; set; }
    public string Identifier { get; set; }
    public string Purpose { get; set; }
    public string Condition { get; set; }
    public string Actions { get; set; }
    public long? ParentId { get; set; }
}
```



```

public virtual ICollection<Test> Tests { get; private set; }
public virtual Test ParentTest { get; set; }
public virtual ICollection<Requirement> Requirements { get; private set; }
public virtual ICollection<Satellite> Satellites { get; private set; }
public virtual ICollection<Syst> Sysys { get; private set; }
}

```

Связь «многие-ко-многим», например, между тестами и системами, достигается путем того, что каждый тест содержит коллекцию связанных с ним систем, а класс системы, в свою очередь, содержит коллекцию связанных с системой тестов.

В слое доступа к данным (в модуле `DataAccessLayer`) описывается конфигурация сущности «Тест» (класс `TestConfig`), необходимая для правильного обращения к репозиторию.

В приведенном фрагменте программного кода описывается конфигурация сущности «Тест» с учетом наличия связей «многие-ко-многим» между тестами и функциональными требованиями, тестами и системами, тестами и изделиями, а также между базовыми и модифицированными тестами.

```

public TestConfig()
{
    HasMany(e => e.Tests)
        .WithOptional(e => e.ParentTest)
        .HasForeignKey(e => e.ParentId);

    HasMany(e => e.Requirements)
        .WithMany(e => e.Tests)
        .Map(m => m.ToTable("RequirementTest").MapLeftKey("TestId").MapRightKey("RequirementId")); =>

    HasMany(e => e.Sysys)
        .WithMany(e => e.Tests)
        .Map(m => m.ToTable("TestSyst").MapLeftKey("TestId").MapRightKey("SystId"));

    HasMany(e => e.Satellites)
        .WithMany(e => e.Tests)
        .Map(m => m.ToTable("TestSatellite").MapLeftKey("TestId").MapRightKey("SatelliteId")); =>
}

```

Для внесения изменений в структуру БД создаются миграции с помощью `EntityFramework` в консоли менеджера пакетов `NuGet`.

Команда создания миграции имеет вид:

```

add-migration -name "<ИМЯ МИГРАЦИИ>" -ProjectName "Saprbpo.DataAccessLayer" -
StartupProjectName Saprbpo.DataAccessLayer

```

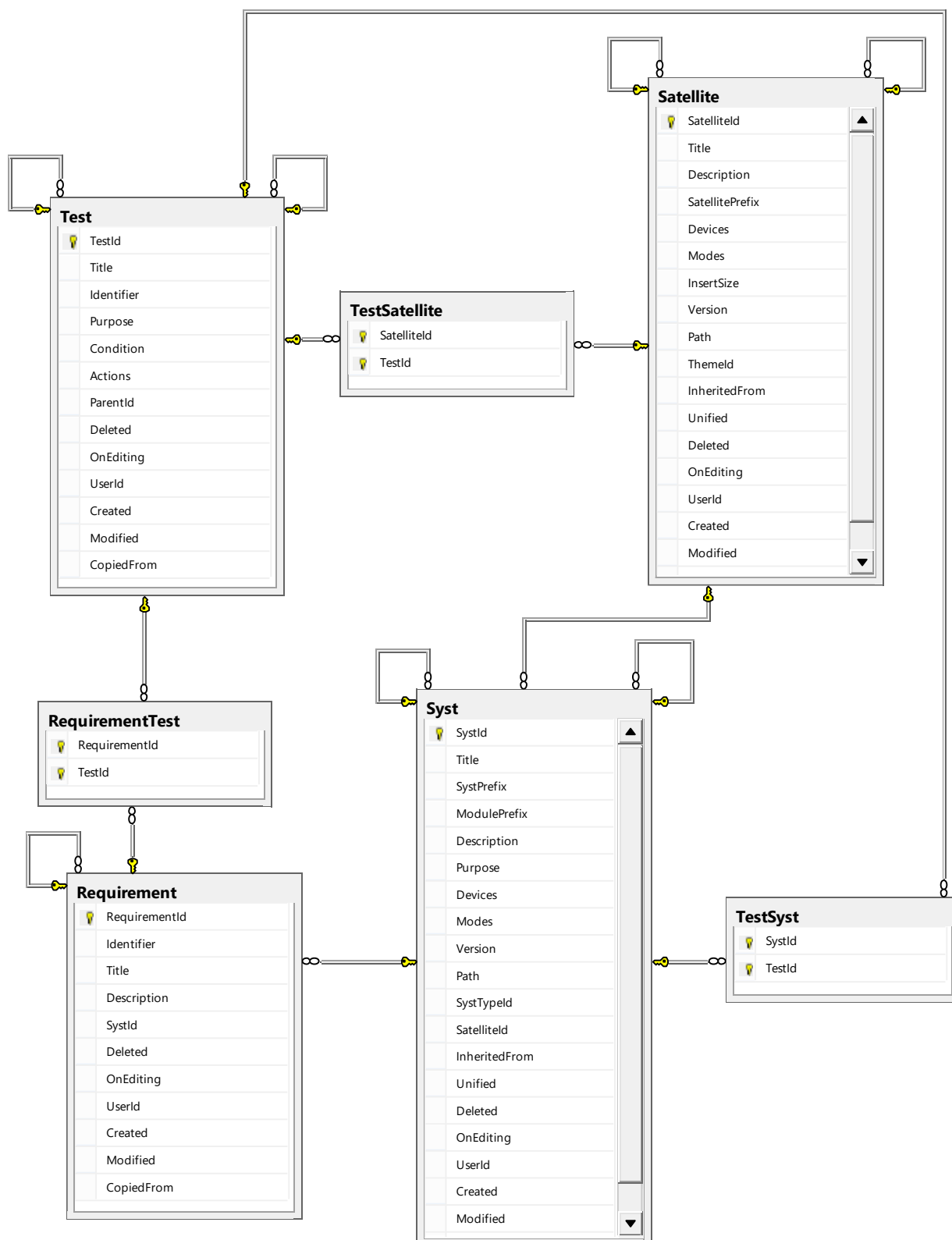


Рисунок 20 – Фрагмент БД, иллюстрирующий сущность «Тест» и её связи.

Команда обновления БД (должна быть выполнена после создания миграции) имеет вид:

```
Update-database -ProjectName SaprBpo.DataAccessLayer -StartupProjectName
SaprBpo.DataAccessLayer
```

Результатом выполнения миграции является изменение структуры таблиц в БД, расположенной на сервере БД.

### 2.2.2 Подготовка к реализации сервиса управления тестами

Перед создание сервиса с использованием средств WCF необходимо объявить контракт сервиса в модуле Common.

Контракт сервиса представляет собой интерфейс, описывающий доступные операции и использующийся клиентским приложением для обращения к операциям сервиса.

Интерфейс сервиса ITestService содержит операции получения тестов на систему, изделие или функциональное требование, а также добавления или удаления связи между тестом и системой, тестом и изделием, тестом и функциональным требованием.

```
[ServiceContract]
public interface ITestService : IGenericService<TestDto>
{
    [OperationContract]
    DtoList<TestDto> GetAllTestsAssignedToSyst(long systId);

    [OperationContract]
    DtoList<TestDto> GetAllTestsAssignedToSatellite(long satelliteId);

    [OperationContract]
    DtoList<TestDto> GetAllTestsAssignedToRequirement(long reqId);

    [OperationContract]
    DtoWrapper<bool> AssignTestToSyst(long systId, long testId, bool assign);

    [OperationContract]
    DtoWrapper<bool> AssignTestToSatellite(long satelliteId, long testId, bool assign);

    [OperationContract]
    DtoWrapper<bool> AssignTestToRequirement(long reqId, long testId, bool assign);
}
```

Кроме этого, необходимо объявить промежуточный класс TestDto, хранящий содержимое полей теста и служащий для управления логикой и привязки отображения полей теста в элементах пользовательского интерфейса.

```
public class TestDto : DtoEntityBase
{
    [TypicalProperty(Type = TypicalProperty.Title)]
    public string Title { get; set; }

    public string Identifier { get; set; }
    public string Purpose { get; set; }
    public string Condition { get; set; }
}
```

```

public string Actions { get; set; }
public long? ParentId { get; set; }
public bool IsAssignedToSatellite { get; set; } = false;
public bool IsAssignedToSyst { get; set; } = false;
public bool IsAssignedToRequirement { get; set; } = false;
}

```

### 2.2.3 Реализация сервиса управления тестами

В слое бизнес-логики (в модуле BusinessLayer) реализуется класс TestService (Рисунок 21).

Базовый сервис GenericService уже содержит реализацию общих операций работы с сущностями (в том числе наследует от класса CrudService операции чтения, добавления, редактирования и удаления сущностей), поэтому дополнительные операции (получение тестов на конкретную систему, функциональное требование или изделие, управление связями тестов с системами, требованиями и изделиями) заявлены в интерфейсе ITestService.

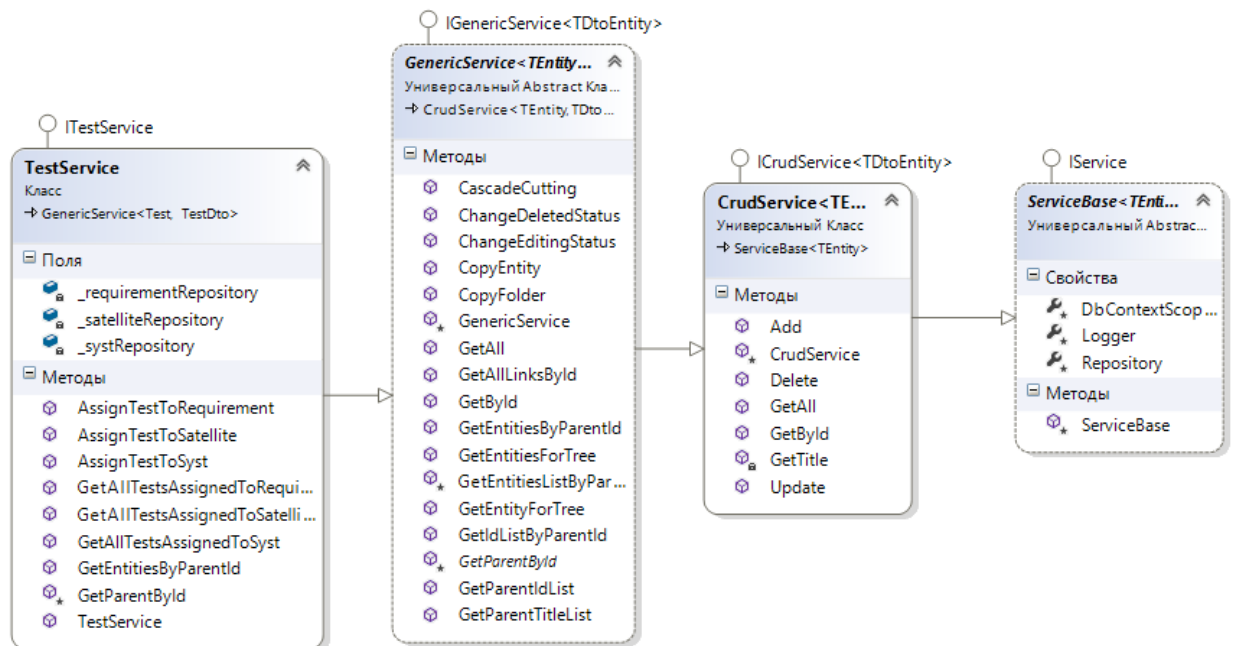


Рисунок 21 – Диаграмма классов, связанных с TestService.

Класс TestService также реализует собственные операции, предназначенные для гибкого управления тестами:

- AssignTestToRequirement – связывание теста и функционального требования;
- AssignTestToSatellite – связывание теста и изделия;
- AssignTestToSyst – связывание теста и системы;

- GetAllTestsAssignedToRequirement – получение тестов, относящихся к функциональному требованию;
- GetAllTestsAssignedToSatellite – получение тестов, относящихся к изделию;
- GetAllTestsAssignedToSyst – получение тестов, относящихся к системе.

В слое сервисов (в модуле WcfServices) объявляется конфигурация ServiceHost для сервиса TestService.

```
<%@ ServiceHost Language="C#"
    Debug="true"
    Service="SaprBpo.BusinessLayer.Services.PartialServices.TestService"
    Factory="SaprBpo.ServiceLayer.ServiceFactory" %>
```

Кроме этого, в файле Web.config объявляются настройки и способ привязки сервиса TestService.

```
<service name="SaprBpo.BusinessLayer.Services.PartialServices.TestService">
    <endpoint address="TestService" binding="basicHttpBinding"
        contract="SaprBpo.Common.ServiceContract.PartialServiceContracts.ITestService"/>
</service>
```

## 2.2.4 Доработка клиентского приложения для работы с тестами

Доработка клиентского приложения для работы с тестами включает в себя реализацию подключения к новому сервису, расширение логики дерева объектов, добавление элементов отображения тестов и реализацию их логики.

### 2.2.4.1 Подключение к новому сервису

Для настройки подключения к новому сервису в клиентском приложении добавлен класс TestServiceProxy.

```
public class TestServiceProxy : WcfAdapterBaseWithGeneric<ITestService, TestDto>,
ITestService
{
    public DtoList<TestDto> GetAllTestsAssignedToSyst(long systId)
    {
        return ExecuteCommand(proxy => proxy.GetAllTestsAssignedToSyst(systId));
    }
    ...
}
```

Также добавлено сопоставление интерфейса ITestService и прокси-класса TestServiceProxy в классе SimpleServiceLocator.

```
SimpleIoc.Default.Register<ITestService, TestServiceProxy>();
```

#### 2.2.4.2 Добавление узлов в дерево объектов

Для отображения тестов в дерево объектов добавлены новые узлы и папки:

- папка «Тесты на изделие» (класс TestSatelliteTreeFolder);
- папка «Тесты на систему» (класс TestSystTreeFolder);
- узел «Тест на изделие» (класс TestSatelliteTreeNode);
- узел «Тест на систему» (класс TestSystTreeNode).

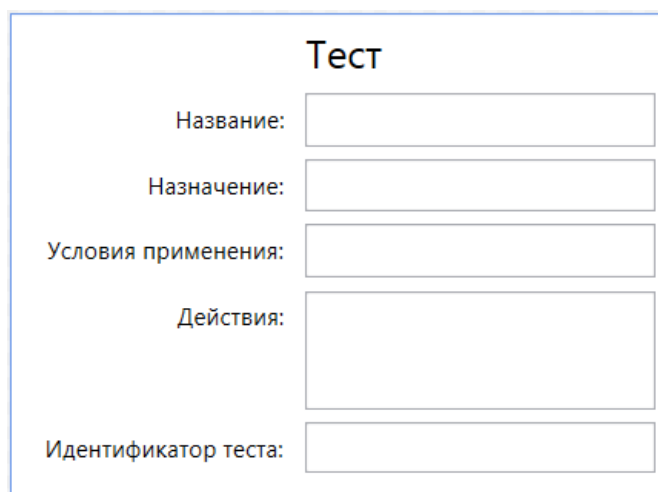
#### 2.2.4.3 Реализация логики управления тестами

В клиентском приложении логике отображения и управления тестами соответствуют следующие классы ViewModel:

- класс TestVm (соответствует карточке создания теста);
- класс TestSystVm (отображение тестов, связанных с системой);
- класс TestSatelliteVm (отображение тестов, связанных с изделием);
- класс RequirementTestVm (отображение матрицы трассируемости функциональных требований в тесты).

#### 2.2.4.4 Настройка элементов отображения тестов

Элементом отображения карточки создания теста является пользовательский элемент управления TestDetailView (Рисунок 22).



Тест

Название:

Назначение:

Условия применения:

Действия:

Идентификатор теста:

Рисунок 22 – Элемент TestDetailView.

Для отображения и управления тестами на систему используется пользовательский элемент управления TestGridView (Рисунок 23).

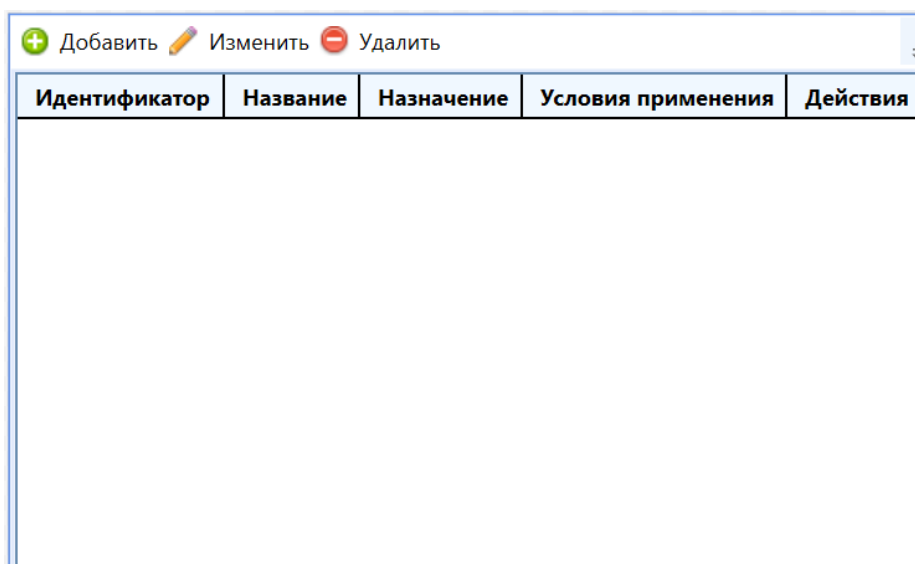


Рисунок 23 – Элемент TestGridView.

Для отображения матрицы трассируемости функциональных требований в тесты используется элемент управления RequirementTestView.

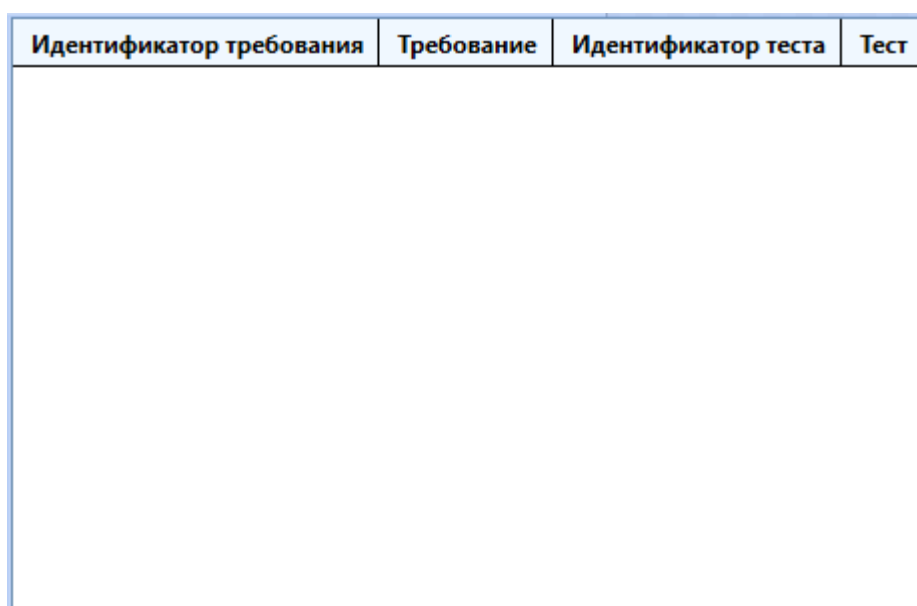


Рисунок 24 – Элемент RequirementTestView.

Настройка форм отображения осуществляется с использованием средств платформы WPF и фреймворка MVVMLight путем привязки классов, предоставляющих данные для отображения, использования стилей, триггеров и команд.

### 3 ПРОГРАММА И МЕТОДИКА ИСПЫТАНИЙ

Объектом испытаний является обновленный и расширенный функционал САПР БПО, определяемой в следующем составе:

База данных:

- база данных `saprg_bpo_iss`, под управлением СУБД MS SQL Server 2014 Express;

Серверная часть:

- библиотеки сервисов (dll-библиотеки);
- сервисы WCF;

Клиентская часть:

- пользовательский интерфейс клиентского приложения САПР БПО.

Целью испытания базы данных является проверка корректности операций добавления, чтения, редактирования данных в слое доступа к данным с использованием технологии ORM, обеспечивающейся за счет EntityFramework.

Целями испытания серверной части являются:

- проверка корректности выполнения операций добавления, чтения и редактирования данных в слое бизнес-логики для тестов и связанных с ними сущностей;
- проверка корректности выполнения операций добавления, чтения и редактирования данных в слое представления данных для тестов и связанных с ними сущностей.

Целью испытания клиентской части является проверка корректности выполнения операций добавления, чтения и редактирования данных с использованием операций логики клиентского приложения.

Испытания проводятся с использованием ручного тестирования, позволяющего провести сквозное тестирование всех слоев САПР БПО.



Входной точкой тестирования является пользовательский интерфейс клиентского приложения САПР БПО, получающий данные от сервисов WCF, расположенных на сервере приложений. Сервисы WCF для выполнения запрашиваемых операций обращаются к библиотекам бизнес-логики, которые в свою очередь используют слой доступа к данным, реализуемый набором репозитория и технологией EntityFramework. Таким образом выполняется тестирование всех слоев приложения.

### **3.1 Перечень проводимых испытаний**

Перечень проводимых испытаний содержит разделен на три группы: проверка функционала управления тестами на изделие, проверка функционала управления тестами на систему, проверка отображения матрицы трассируемости функциональных требований в тесты.

Далее приводится описание проводимых испытаний.

#### **3.1.1 Управление тестами на изделие**

В рамках управления тестами на изделие пользователю доступны возможности добавления, получения, редактирования и удаления тестов на изделие.

Для перехода к вкладке «Тесты на изделие» пользователю необходимо в дереве объектов перейти к папке изделия, внутри которой следует выбрать папку «Тесты» (Рисунок 25).

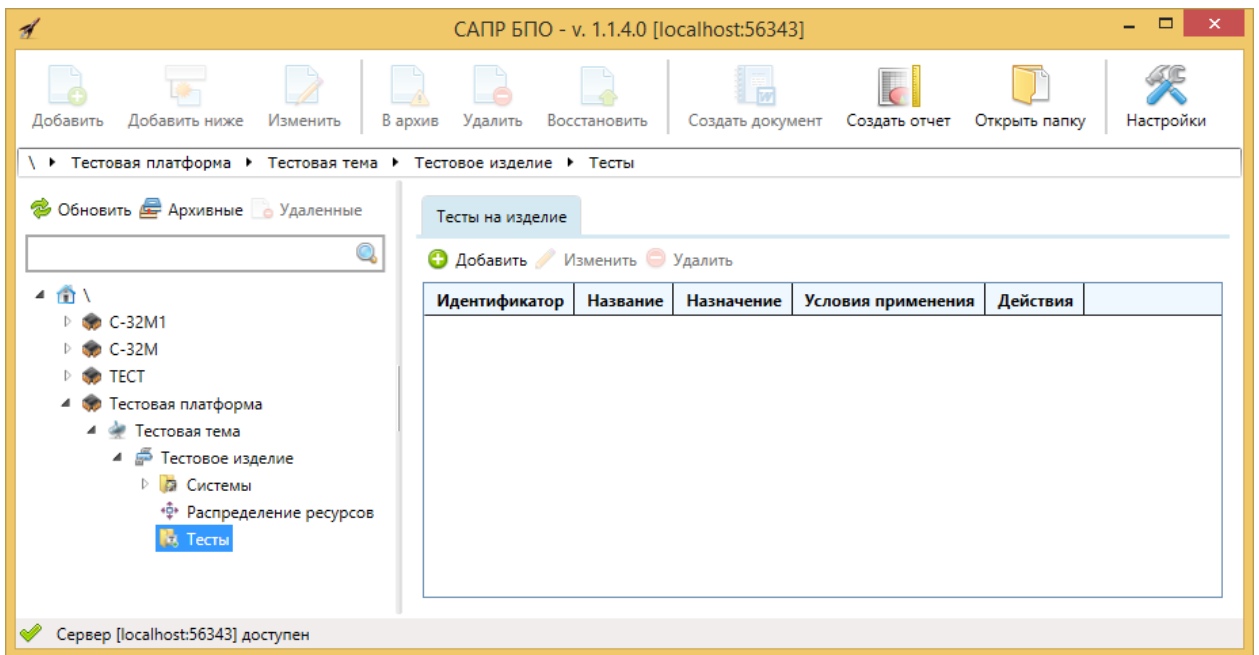


Рисунок 25 – Вкладка «Тесты на изделие».

### 3.1.1.1 Добавление теста на изделие

Для добавления теста на изделие следует нажать кнопку «Добавить», в отобразившемся окне (Рисунок 26) заполнить поля создаваемого теста и нажать кнопку «Ок».

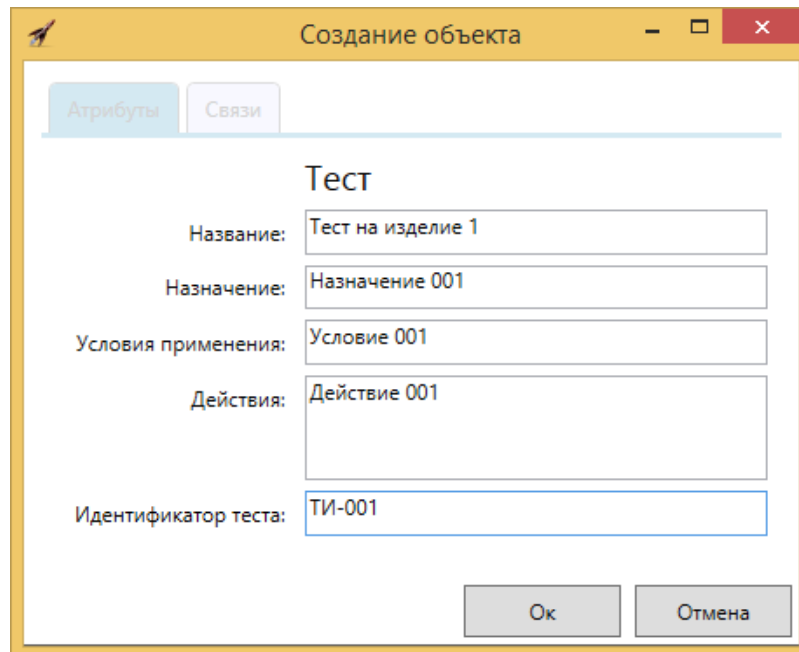


Рисунок 26 – Создание теста на изделие.

Созданный тест должен появиться в списке тестов на выбранное изделие (Рисунок 27).

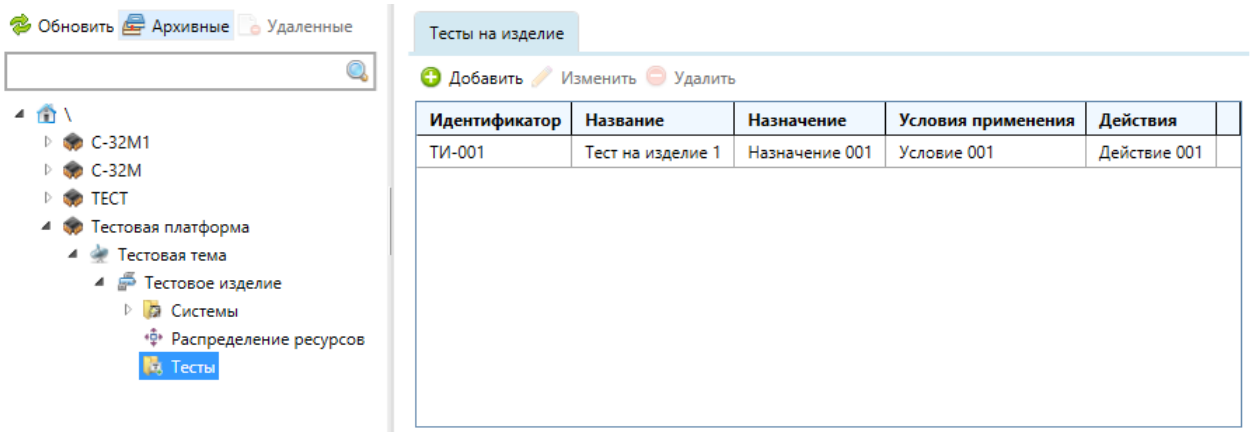


Рисунок 27 – Результат добавления теста на изделие.

Кроме этого, запись о созданном тесте должна появиться в таблице Test (Рисунок 28), запись о связи теста с изделием должна отображаться в таблице TestSatellite (Рисунок 29).

TestId	Title	Identifier	Purpose	Condition	Actions	ParentId	Deleted	OnEditing	UserId
1	Тест 1	T01	1	1	1	NULL	2	False	1
2	Тест 2-1	T02	2	2	2	NULL	0	False	NULL
3	Тест 1.1	T1-1	Применение	Условия	Действия	NULL	0	False	NULL
4	Тест 1	T11	Проверка	У1	Д1	NULL	0	False	1
5	Тест на издели...	TI1-1	Тестирование ...	У2	Д2	NULL	0	False	1
6	Тест на издели...	TI-001	Назначение 001	Условие 001	Действие 001	NULL	0	False	1
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Рисунок 28 – Запись в таблице Test.

Satelliteld	TestId
10007	1
10007	2
1	5
10009	6
*	NULL

Рисунок 29 – Запись в таблице TestSatellite.

### 3.1.1.2 Редактирование теста на изделие

Для редактирования теста на изделие следует выбрать тест в списке (Рисунок 30) и нажать кнопку «Изменить».

Тесты на изделие					
Идентификатор	Название	Назначение	Условия применения	Действия	
ТИ-001	Тест на изделие 1	Назначение 001	Условие 001	Действие 001	
ТИ-002	Тест на изделие 2	Назначение 002	Условие 002	Действие 002	

Рисунок 30 – Выбор теста в списке.

В отобразившемся окне (Рисунок 31) следует изменить содержимое необходимых полей теста и нажать кнопку «Ок».

Информация о тесте должна обновиться в списке тестов (Рисунок 32).

Изменение объекта

Атрибуты | Связи

**Тест**

Название:

Назначение:

Условия применения:

Действия:

Идентификатор теста:

Рисунок 31 – Окно редактирования теста на изделие.

Тесты на изделие					
Идентификатор	Название	Назначение	Условия применения	Действия	
ТИ-001	Тест на изделие 1	Назначение 001	Условие 001	Действие 001	
ТИ-002	Тест на изделие 2	Назначение 002	Условие 002	Действие 002, Действие 003 (отредактировано)	

Рисунок 32 – Результат редактирования теста на изделие.

Кроме этого, запись о тесте должна обновиться в таблице Test (Рисунок 33).

TestId	Title	Identifier	Purpose	Condition	Actions	Parent...	Deleted	OnEditing	Userid	Created	Modified	CopiedFrom
1	Тест 1	T01	1	1	1	NULL	2	False	1	29.05.20...	29.05.201...	NULL
2	Тест 2-1	T02	2	2	2	NULL	0	False	NULL	29.05.20...	29.05.201...	NULL
3	Тест 1.1	T1-1	Применение	Условия	Действия	NULL	0	False	NULL	29.05.20...	29.05.201...	NULL
4	Тест 1	T11	Проверка	У1	Д1	NULL	0	False	1	29.05.20...	29.05.201...	NULL
5	Тест на издели...	ТИ1-1	Тестирование ...	У2	Д2	NULL	0	False	1	29.05.20...	29.05.201...	NULL
6	Тест на издели...	ТИ-001	Назначение 001	Условие 001	Действие 001	NULL	0	False	1	30.05.20...	30.05.201...	NULL
7	Тест на издели...	ТИ-002	Назначение 002	Условие 002	Действие 002, Действие 003 (отредактир...	NULL	0	False	NULL	30.05.20...	30.05.201...	NULL
8	Тест на издели...	ТИ-003	Назначение 003	Условие 003	Действие 003	NULL	0	False	1	30.05.20...	30.05.201...	NULL
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Рисунок 33 – Отредактированная запись в таблице Test.

### 3.1.1.3 Удаление теста на изделие

Для удаления теста на изделие следует выбрать тест в списке (Рисунок 34) и нажать кнопку «Удалить».

Идентификатор	Название	Назначение	Условия применения	Действия
ТИ-001	Тест на изделие 1	Назначение 001	Условие 001	Действие 001
ТИ-002	Тест на изделие 2	Назначение 002	Условие 002	Действие 002, Действие 003 (отредактировано)
ТИ-003	Тест на изделие 3	Назначение 003	Условие 003	Действие 003

Рисунок 34 – Выбор теста в списке.

В отобразившемся окне (Рисунок 35) следует нажать кнопку «Да».

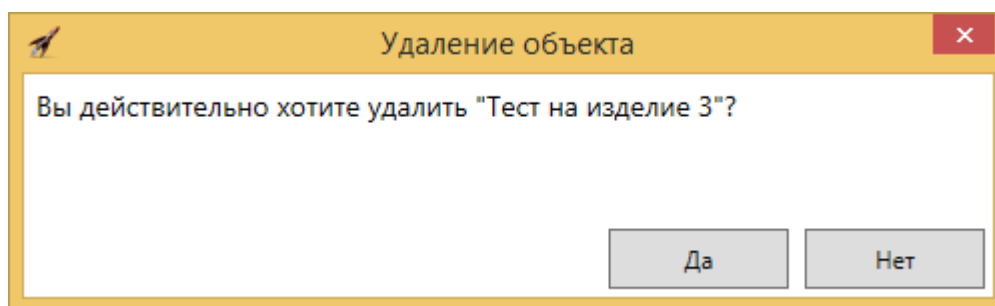


Рисунок 35 – Окно подтверждения удаления теста.

Информация о списке тестов должна обновиться (Рисунок 36).

Идентификатор	Название	Назначение	Условия применения	Действия
ТИ-001	Тест на изделие 1	Назначение 001	Условие 001	Действие 001
ТИ-002	Тест на изделие 2	Назначение 002	Условие 002	Действие 002, Действие 003 (отредактировано)

Рисунок 36 – Результат удаления теста на изделие.

Удаленный тест должен остаться в таблице Test, причем поле Deleted должно иметь значение «2» (Рисунок 37).

TestId	Title	Identifier	Purpose	Condition	Actions	Parent...	Deleted	OnEditing	UserId	Created	Modified	CopiedFrom
1	Тест 1	T01	1	1	1	NULL	2	False	1	29.05.201...	29.05.201...	NULL
2	Тест 2-1	T02	2	2	2	NULL	0	False	NULL	29.05.20...	29.05.201...	NULL
3	Тест 1.1	T1-1	Применение	Условия	Действия	NULL	0	False	NULL	29.05.20...	29.05.201...	NULL
4	Тест 1	T11	Проверка	У1	Д1	NULL	0	False	1	29.05.20...	29.05.201...	NULL
5	Тест на издели...	ТИ1-1	Тестирование ...	У2	Д2	NULL	0	False	1	29.05.20...	29.05.201...	NULL
6	Тест на издели...	ТИ-001	Назначение 001	Условие 001	Действие 001	NULL	0	False	1	30.05.20...	30.05.201...	NULL
7	Тест на издели...	ТИ-002	Назначение 002	Условие 002	Действие 002, ...	NULL	0	False	NULL	30.05.20...	30.05.201...	NULL
8	Тест на издели...	ТИ-003	Назначение 003	Условие 003	Действие 003	NULL	2	False	1	30.05.20...	30.05.201...	NULL
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Рисунок 37 – Удаленная запись в таблице Test.

### 3.1.2 Управление тестами на систему

В рамках управления тестами на систему пользователю доступны возможности добавления, получения, редактирования и удаления тестов на систему.

Для перехода к вкладке «Тесты на систему» пользователю необходимо в дереве объектов перейти к папке принадлежащей изделию системы, внутри которой следует выбрать папку «Тесты» (Рисунок 38).

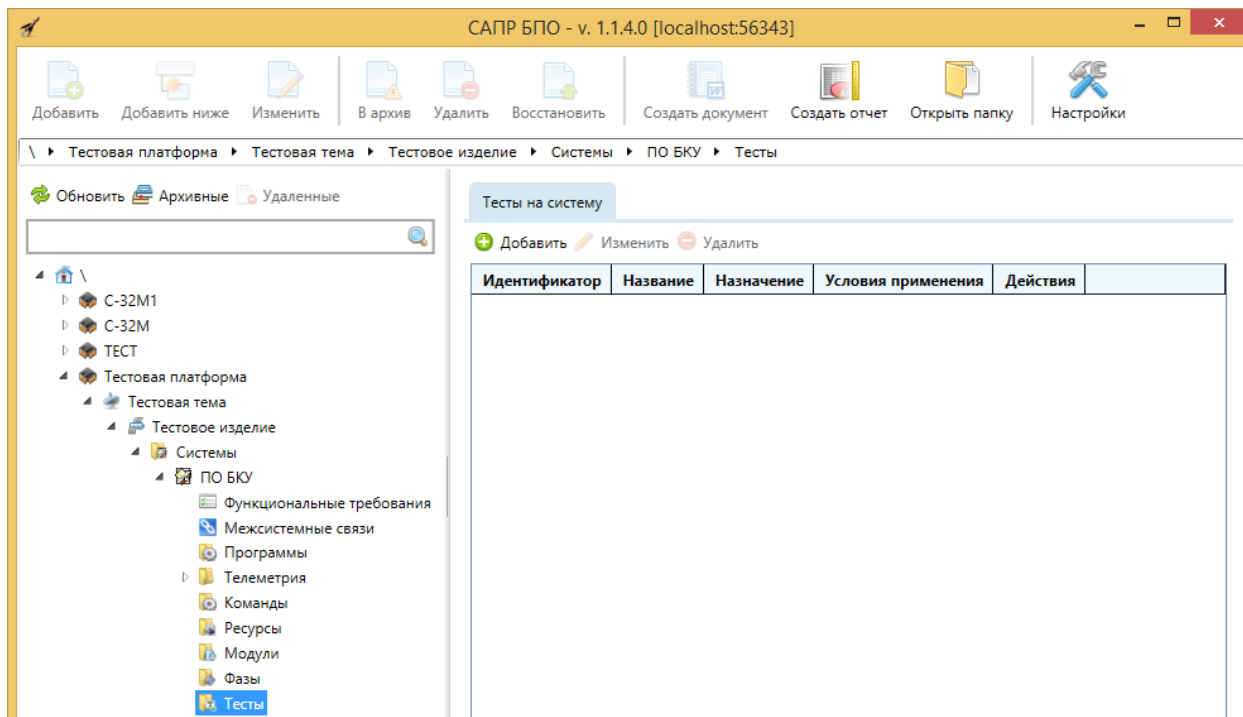


Рисунок 38 – Вкладка «Тесты на систему».

### 3.1.2.1 Добавление теста на систему

Для добавления теста на систему следует нажать кнопку «Добавить», в отобразившемся окне (Рисунок 39) заполнить поля создаваемого теста и нажать кнопку «Ок».

Созданный тест должен появиться в списке тестов на систему (Рисунок 40).

Кроме этого, запись о созданном тесте должна появиться в таблице Test (Рисунок 41), запись о связи теста с системой должна отобразиться в таблице TestSyst (Рисунок 42).

Создание объекта

Атрибуты Связи

Тест

Название: Тест на систему 101

Назначение: Назначение 101

Условия применения: Условие 101

Действия: Действие 101

Идентификатор теста: ТС-101

Ок Отмена

Рисунок 39 – Создание теста на систему.

Обновить Архивные Удаленные

Тесты на систему

+ Добавить ✎ Изменить - Удалить

Идентификатор	Название	Назначение	Условия применения	Действия
ТС-101	Тест на систему 101	Назначение 101	Условие 101	Действие 101

Рисунок 40 – Результат добавления теста на систему.

TestId	Title	Identifier	Purpose	Condition	Actions	Parent...	Deleted	OnEditing	Userld	Created	Modified	CopiedFrom
1	Тест 1	T01	1	1	1	NULL	2	False	1	29.05.20...	29.05.201...	NULL
2	Тест 2-1	T02	2	2	2	NULL	0	False	NULL	29.05.20...	29.05.201...	NULL
3	Тест 1.1	T1-1	Применение	Условия	Действия	NULL	0	False	NULL	29.05.20...	29.05.201...	NULL
4	Тест 1	T11	Проверка	У1	Д1	NULL	0	False	1	29.05.20...	29.05.201...	NULL
5	Тест на издели...	ТИ1-1	Тестирование ...	У2	Д2	NULL	0	False	1	29.05.20...	29.05.201...	NULL
6	Тест на издели...	ТИ-001	Назначение 001	Условие 001	Действие 001	NULL	0	False	1	30.05.20...	30.05.201...	NULL
7	Тест на издели...	ТИ-002	Назначение 002	Условие 002	Действие 002, ...	NULL	0	False	NULL	30.05.20...	30.05.201...	NULL
8	Тест на издели...	ТИ-003	Назначение 003	Условие 003	Действие 003	NULL	2	False	1	30.05.20...	30.05.201...	NULL
9	Тест на систем...	ТС-101	Назначение 101	Условие 101	Действие 101	NULL	0	False	1	30.05.20...	30.05.201...	NULL
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Рисунок 41 – Запись в таблице Test.

SystId	TestId
10030	3
1	4
10036	9
*	NULL

Рисунок 42 – Запись в таблице TestSyst.

### 3.1.2.2 Редактирование теста на систему

Для редактирования теста на систему следует выбрать тест в списке (Рисунок 43) и нажать кнопку «Изменить».

Тесты на систему

+ Добавить    ✎ Изменить    - Удалить

Идентификатор	Название	Назначение	Условия применения	Действия
ТС-101	Тест на систему 101	Назначение 101	Условие 101	Действие 101
ТС-102	Тест на систему 102	Назначение 102	Условие 102	Действие 102

Рисунок 43 – Выбор теста в списке.

В отобразившемся окне (Рисунок 44) следует изменить содержимое необходимых полей теста и нажать кнопку «Ок».



Рисунок 44 – Окно редактирования теста на систему.

Информация о тесте должна обновиться в списке тестов (Рисунок 45).

Тесты на систему

+ Добавить ✎ Изменить - Удалить

Идентификатор	Название	Назначение	Условия применения	Действия
ТС-101	Тест на систему 101	Назначение 101	Условие 101	Действие 101
ТС-102	Тест на систему 102	Назначение 102	Условие 102	Действие 102, Действие 103 (отредактировано)

Рисунок 45 – Результат редактирования теста на систему.

Кроме этого, запись о тесте должна обновиться в таблице Test (Рисунок 46).

dbo.TestSyst [Данные]    dbo.Test [Данные] ✕

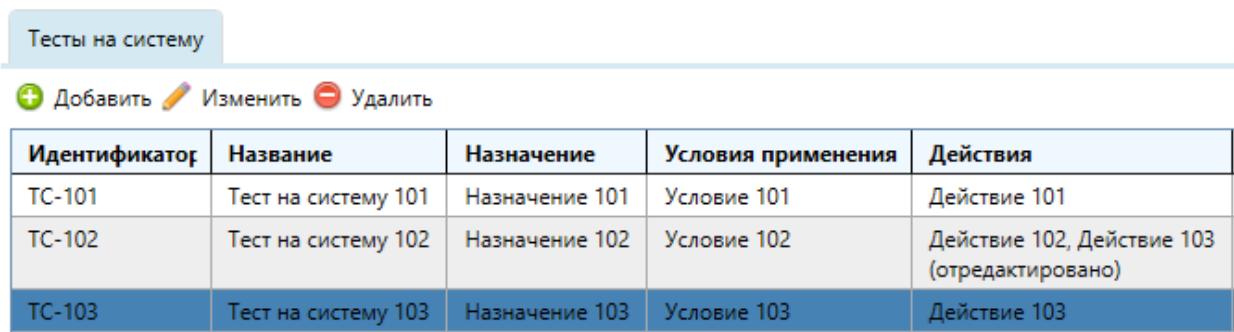
Максимальное количество строк: 1000

	Testid	Title	Identifier	Purpose	Condition	Actions	Parent...	Deleted	OnEditing	Userid	Created	Modified	CopiedFrom
1	Тест 1	T01	1	1	1	1	NULL	2	False	1	29.05.20...	29.05.201...	NULL
2	Тест 2-1	T02	2	2	2	2	NULL	0	False	NULL	29.05.20...	29.05.201...	NULL
3	Тест 1.1	T1-1	Применение	Условия	Действия		NULL	0	False	NULL	29.05.20...	29.05.201...	NULL
4	Тест 1	T11	Проверка	У1	Д1		NULL	0	False	1	29.05.20...	29.05.201...	NULL
5	Тест на издели...	ТИ1-1	Тестирование ...	У2	Д2		NULL	0	False	1	29.05.20...	29.05.201...	NULL
6	Тест на издели...	ТИ-001	Назначение 001	Условие 001	Действие 001		NULL	0	False	1	30.05.20...	30.05.201...	NULL
7	Тест на издели...	ТИ-002	Назначение 002	Условие 002	Действие 002, Действие 003(отредакти...		NULL	0	False	NULL	30.05.20...	30.05.201...	NULL
8	Тест на издели...	ТИ-003	Назначение 003	Условие 003	Действие 003		NULL	2	False	1	30.05.20...	30.05.201...	NULL
9	Тест на систем...	ТС-101	Назначение 101	Условие 101	Действие 101		NULL	0	False	1	30.05.20...	30.05.201...	NULL
▶	10	Тест на систем...	ТС-102	Назначение 102	Условие 102	Действие 102, Действие 103(отредакти...	NULL	0	False	NULL	30.05.20...	30.05.201...	NULL
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Рисунок 46 – Отредактированная запись в таблице Test.

### 3.1.2.3. Удаление теста на систему

Для удаления теста на систему следует выбрать тест в списке (Рисунок 47) и нажать кнопку «Удалить».



Тесты на систему

+ Добавить ✎ Изменить - Удалить

Идентификатор	Название	Назначение	Условия применения	Действия
ТС-101	Тест на систему 101	Назначение 101	Условие 101	Действие 101
ТС-102	Тест на систему 102	Назначение 102	Условие 102	Действие 102, Действие 103 (отредактировано)
ТС-103	Тест на систему 103	Назначение 103	Условие 103	Действие 103

Рисунок 47 – Выбор теста в списке.

В отобразившемся окне (Рисунок 48) следует нажать кнопку «Да».

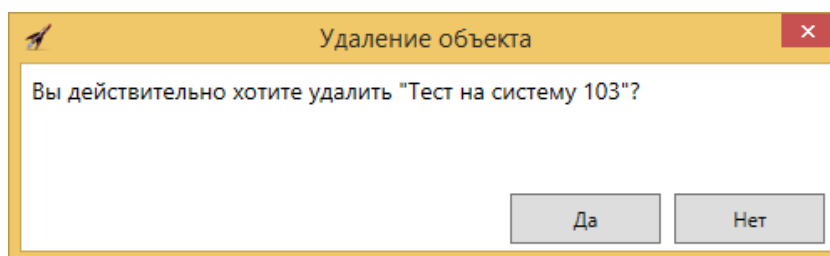
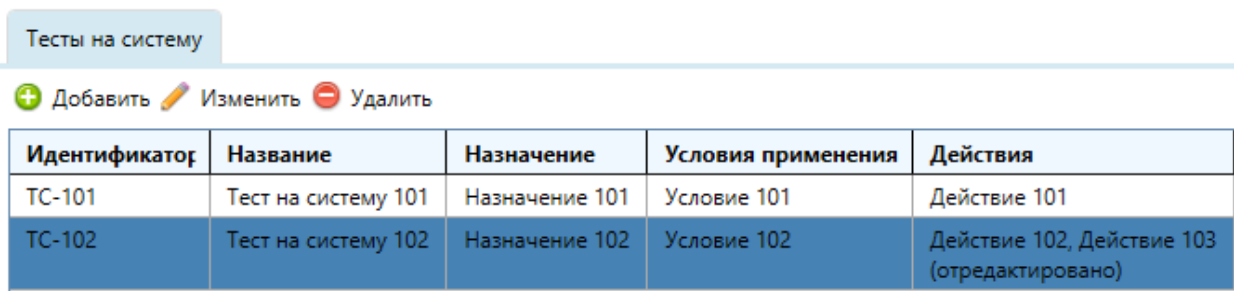


Рисунок 48 – Окно подтверждения удаления теста.

Информация о списке тестов должна обновиться (Рисунок 49).



Тесты на систему

+ Добавить ✎ Изменить - Удалить

Идентификатор	Название	Назначение	Условия применения	Действия
ТС-101	Тест на систему 101	Назначение 101	Условие 101	Действие 101
ТС-102	Тест на систему 102	Назначение 102	Условие 102	Действие 102, Действие 103 (отредактировано)

Рисунок 49 – Результат удаления теста на изделие.

Удаленный тест должен остаться в таблице Test, причем поле Deleted должно иметь значение «2» (Рисунок 50).

TestId	Title	Identifier	Purpose	Condition	Actions	Parent...	Deleted	OnEditing	UserId	Created	Modified	CopiedFrom
1	Тест 1	T01	1	1	1	NULL	2	False	1	29.05.20...	29.05.201...	NULL
2	Тест 2-1	T02	2	2	2	NULL	0	False	NULL	29.05.20...	29.05.201...	NULL
3	Тест 1.1	T1-1	Применение	Условия	Действия	NULL	0	False	NULL	29.05.20...	29.05.201...	NULL
4	Тест 1	T11	Проверка	У1	Д1	NULL	0	False	1	29.05.20...	29.05.201...	NULL
5	Тест на издели...	ТИ1-1	Тестирование ...	У2	Д2	NULL	0	False	1	29.05.20...	29.05.201...	NULL
6	Тест на издели...	ТИ-001	Назначение 001	Условие 001	Действие 001	NULL	0	False	1	30.05.20...	30.05.201...	NULL
7	Тест на издели...	ТИ-002	Назначение 002	Условие 002	Действие 002, Действие 003 (отредакти...	NULL	0	False	NULL	30.05.20...	30.05.201...	NULL
8	Тест на издели...	ТИ-003	Назначение 003	Условие 003	Действие 003	NULL	2	False	1	30.05.20...	30.05.201...	NULL
9	Тест на систем...	ТС-101	Назначение 101	Условие 101	Действие 101	NULL	0	False	1	30.05.20...	30.05.201...	NULL
10	Тест на систем...	ТС-102	Назначение 102	Условие 102	Действие 102, Действие 103 (отредакти...	NULL	0	False	NULL	30.05.20...	30.05.201...	NULL
11	Тест на систем...	ТС-103	Назначение 103	Условие 103	Действие 103	NULL	2	False	1	30.05.20...	30.05.201...	NULL
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Рисунок 50 – Удаленная запись в таблице Test.

### 3.1.3 Матрица трассируемости требований в тесты

Матрица трассируемости функциональных требований в тесты предназначена для отображения связи между списком функциональных требований на систему и тестов.

Для перехода к вкладке «Матрица трассируемости» пользователю необходимо в дереве объектов перейти к папке принадлежащей изделию системы, внутри которой следует выбрать пункт «Матрица трассируемости», затем перейти во вкладку «Трассируемость в тесты» (Рисунок 51).

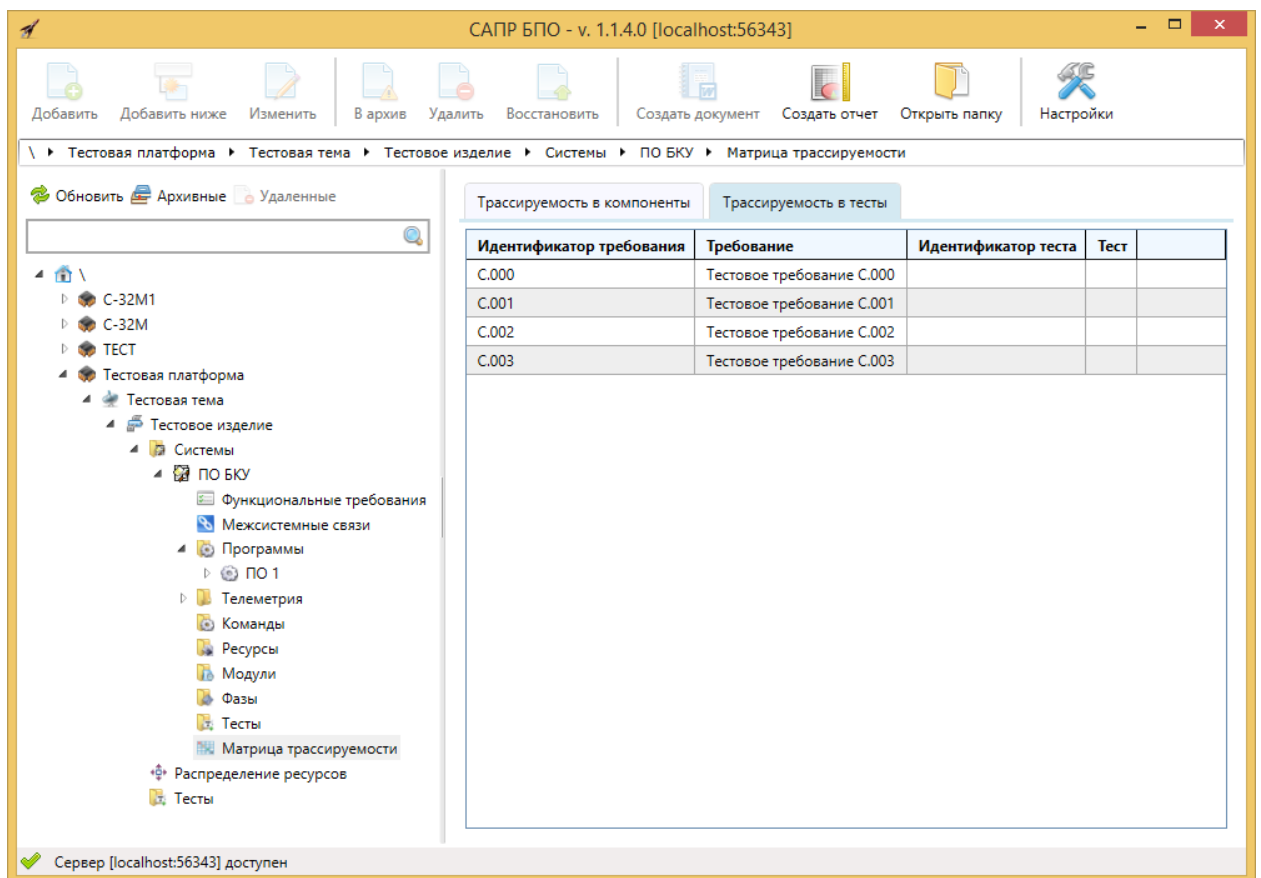
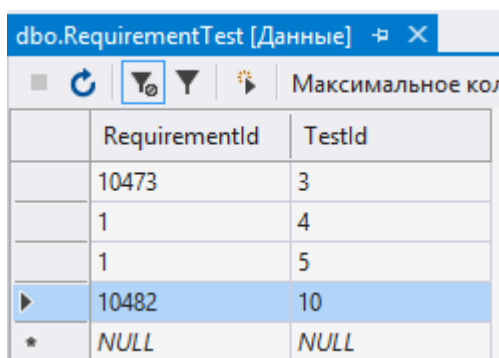


Рисунок 51 – Вкладка «Трассируемость в тесты».

### 3.1.3.1 Связывание теста и функционального требования

Связывание функциональных требований и тестов осуществляется через таблицу RequirementTest.

Для установления связи между функциональным требованием и тестом в таблицу RequirementTest необходимо добавить запись, содержащую идентификаторы требования и теста (Рисунок 52).

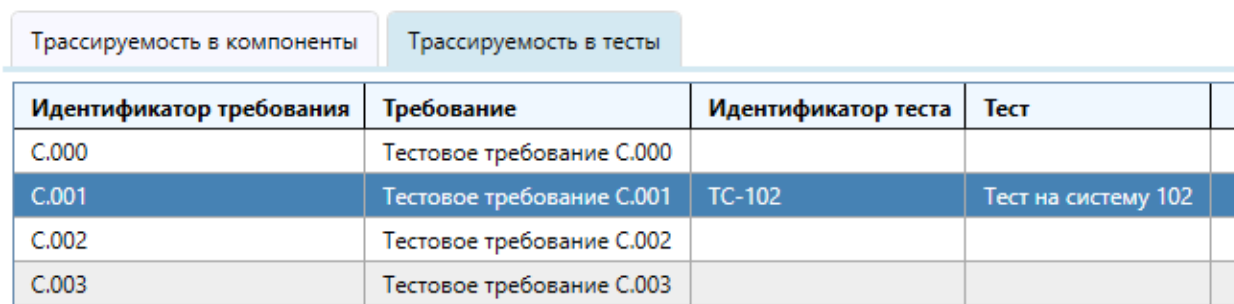


	RequirementId	TestId
	10473	3
	1	4
	1	5
▶	10482	10
*	NULL	NULL

Рисунок 52 – Таблица RequirementTest.

### 3.1.3.2 Отображение матрицы трассируемости

Отображение матрицы трассируемости функциональных требований в тесты осуществляется на основании таблицы RequirementTest. Кроме этого, удаленные функциональные требования и тесты не должны отображаться в матрице трассируемости (Рисунок 53).



Трассируемость в компоненты		Трассируемость в тесты	
Идентификатор требования	Требование	Идентификатор теста	Тест
C.000	Тестовое требование C.000		
C.001	Тестовое требование C.001	ТС-102	Тест на систему 102
C.002	Тестовое требование C.002		
C.003	Тестовое требование C.003		

Рисунок 53 – Матрица трассируемости.

### 3.1.3.3 Отображение связи «многие-ко-многим» между функциональными требованиями и тестами

Для установления связей «многие-ко-многим» между функциональными требованиями и тестами необходимо добавить записи в таблицу RequirementTest (Рисунок 54).

dbo.RequirementTest [Данные] ↗ ✕		
	RequirementId	TestId
	10473	3
	1	4
	1	5
	10482	10
	10482	9
	10483	10
	10483	9
▶*	NULL	NULL

Рисунок 54 – Таблица RequirementTest.

В матрице трассируемости требований в тесты должны отобразиться добавленные связи с учетом того, что одно требование может быть связано с несколькими тестами, а один тест может быть связан с несколькими требованиями (Рисунок 55).

Трассируемость в компоненты		Трассируемость в тесты	
Идентификатор требования	Требование	Идентификатор теста	Тест
C.000	Тестовое требование C.000		
C.001	Тестовое требование C.001	ТС-101	Тест на систему 101
C.001	Тестовое требование C.001	ТС-102	Тест на систему 102
C.002	Тестовое требование C.002	ТС-101	Тест на систему 101
C.002	Тестовое требование C.002	ТС-102	Тест на систему 102

Рисунок 55 – Матрица трассируемости.

## 4 ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСОЭФФЕКТИВНОСТЬ И РЕСУРСОСБЕРЕЖЕНИЕ

Цель раздела – комплексное описание и анализ финансово-экономических аспектов выполненной работы. Оцениваются полные денежные затраты на исследование (проект), а также дать хотя бы приближенную экономическую оценку результатов ее внедрения. Это в свою очередь позволит с помощью традиционных показателей эффективности инвестиций оценить экономическую целесообразность осуществления работы. Раздел завершается комплексной оценкой научно-технического уровня ВКР на основе экспертных данных.

### 4.1 Организация и планирование работ

При организации процесса реализации конкретного проекта необходимо рационально планировать занятость каждого из его участников и сроки проведения отдельных работ.

Данный пункт содержит полный перечень проводимых работ, их исполнители и рациональную продолжительность. Для наглядности и в виду наличия только двух исполнителей результат планирования представлен как линейный график реализации проекта. Для его построения, хронологически упорядоченные вышеуказанные данные должны быть сведены в таблицу 4.1, приведённую ниже.

Таблица 4.1 – Перечень работ и продолжительность их выполнения

Этапы работы	Исполнители	Загрузка исполнителей
Постановка целей и задач, получение исходных данных	НР	НР – 100%
Составление и утверждение технического задания	НР, И	НР – 60% И – 40%
Подбор и изучение материалов по тематике	НР, И	НР – 20% И – 100%
Разработка календарного плана	НР, И	НР – 50% И – 50%
Обсуждение литературы	НР, И	НР – 30% И – 100%

Этапы работы	Исполнители	Загрузка исполнителей
Проектирование пользовательских сценариев	НР, И	НР – 40% И – 80%
Проектирование базы данных	НР, И	НР – 10% И – 100%
Проектирование пользовательских интерфейсов	И	И – 100%
Разработка информационной системы	И	И – 100%
Тестирование информационной системы	И	И – 100%
Оформление пояснительной записки	НР, И	НР – 30% И – 100%
Публикация проекта	И	И – 100%

#### 4.1.1 Продолжительность этапов работ

Расчёт продолжительности этапов работ осуществляется опытно-статистическим методом, реализуемый экспертным способом.

Экспертный способ предполагает генерацию необходимых количественных оценок специалистами конкретной предметной области, опирающимися на их профессиональный опыт и эрудицию. Для определения вероятных (ожидаемых) значений продолжительности работ  $t_{ож}$  используется следующая формула.

$$t_{ож} = \frac{3 \cdot t_{min} + 2 \cdot t_{max}}{5} \quad (4.1)$$

где  $t_{min}$  – минимальная продолжительность работы, дн.;

$t_{max}$  – максимальная продолжительность работы, дн.;

Для выполнения перечисленных в таблице 4.1 работ требуются специалисты:

- инженер – в его роли действует исполнитель НИР (ВКР);
- научный руководитель.

Для построения линейного графика необходимо рассчитать длительность этапов в рабочих днях, а затем перевести её в календарные дни. Расчёт продолжительности выполнения каждого этапа в рабочих днях ( $T_{РД}$ ) ведётся по формуле.

$$T_{РД} = \frac{t_{ож}}{K_{ВН}} \cdot K_{Д} \quad (4.2)$$

где  $t_{ож}$  – продолжительность работы, дн.;

$K_{ВН}$  – коэффициент выполнения работ, учитывающий влияние внешних факторов на соблюдение предварительно определённых длительностей в данной работе  $K_{ВН} = 0.9$ .

$K_{Д}$  – коэффициент, учитывающий дополнительное время на компенсацию непредвиденных задержек и согласование работ, в данном случае  $K_{Д} = 1$ .

Расчёт продолжительности этапа в календарных днях ведётся по формуле.

$$T_{КД} = T_{РД} \cdot T_{К} \quad (4.3)$$

где  $T_{КД}$  – продолжительность выполнения этапа в календарных днях;

$T_{К}$  – коэффициент календарности, позволяющий перейти от длительности работ в рабочих днях к их аналогам в календарных днях, и рассчитываемый по формуле 4.4.

$$T_{К} = \frac{T_{КАЛ}}{T_{КАЛ} - T_{ВД} - T_{ПД}} \quad (4.4)$$

где  $T_{КАЛ}$  – календарные дни ( $T_{КАЛ} = 366$ );

$T_{ВД}$  – выходные и праздничные дни;

Так как исполнитель и научный руководитель работают по разному количеству дней, то необходимо вычислить два коэффициента, для пятидневной рабочей недели и шестидневной.

Для пятидневной:  $T_{ВД} = 118$ .

$$T_{К} = \frac{365}{365 - 118} = 1,48$$

Для шестидневной:  $T_{ВД} = 66$ .

$$T_{К} = \frac{365}{365 - 66} = 1,22$$

В таблице 4.2 приведён пример определения продолжительности этапов работ и их трудоёмкости по исполнителям, занятым на каждом этапе.



В столбцах (3–5) реализован экспертный способ по формуле 4.1. Столбцы 6 и 7 содержат величины трудоёмкости этапа для каждого из двух участников проекта (научный руководитель и инженер) с учётом коэффициента  $K_d = 1$ . Каждое из них в отдельности не может превышать соответствующее значение  $t_{ож} \cdot K_d$ . Столбцы 8 и 9 содержат те же трудоёмкости, выраженные в календарных днях путём дополнительного умножения на  $T_k$ . Итог по столбцу 5 даёт общую ожидаемую продолжительность работы над проектом в рабочих днях, итоги по столбцам 8 и 9 – общие трудоёмкости для каждого из участников проекта. Две последних величины далее будут использованы для определения затрат на оплату труда участников и прочие затраты. Величины трудоёмкости этапов по исполнителям  $T_{kd}$  (данные столбцов 8 и 9 кроме итогов) позволяют построить линейный график осуществления проекта.

Таблица 4.2 – Трудозатраты на выполнение проекта

Этап	Исполнители	Продолжительность работ, дни			Трудоемкость работ по исполнителям чел.- дн.			
					$T_{рд}$		$T_{кд}$	
		$t_{min}$	$t_{max}$	$t_{ож}$	НР	И	НР	И
1	2	3	4	5	6	7	8	9
Постановка целей и задач, получение исходных данных	НР	2	4	2,8	3,11	–	3,8	–
Составление и утверждение технического задания (ТЗ)	НР, И	4	6	4,8	3,2	2,13	3,9	3,16
Подбор и изучение материалов по тематике	НР, И	2	4	2,8	0,62	3,11	0,76	4,61
Разработка календарного плана	НР, И	2	4	2,8	1,56	1,56	1,9	2,3
Обсуждение литературы	НР, И	1	3	1,8	0,6	2	0,73	2,96
Проектирование пользовательских сценариев	НР, И	11	13	11,8	5,24	10,49	6,4	15,54
Проектирование базы данных	НР, И	5	7	5,8	0,64	6,44	0,79	9,55
Проектирование пользовательских интерфейсов	И	7	9	7,8	–	8,67	–	12,84
Разработка информационной системы	И	22	25	23,2	–	25,78	–	38,2
Тестирование информационной системы	И	5	6	5,4	–	6	–	8,89
Оформление пояснительной записка	НР, И	18	20	18,8	6,27	20,89	7,65	30,95
Публикация проекта	И	2	4	2,8	–	3,11	–	4,61
<b>Итого:</b>				<b>90,6</b>	<b>21,24</b>	<b>90,18</b>	<b>25,95</b>	<b>133,62</b>

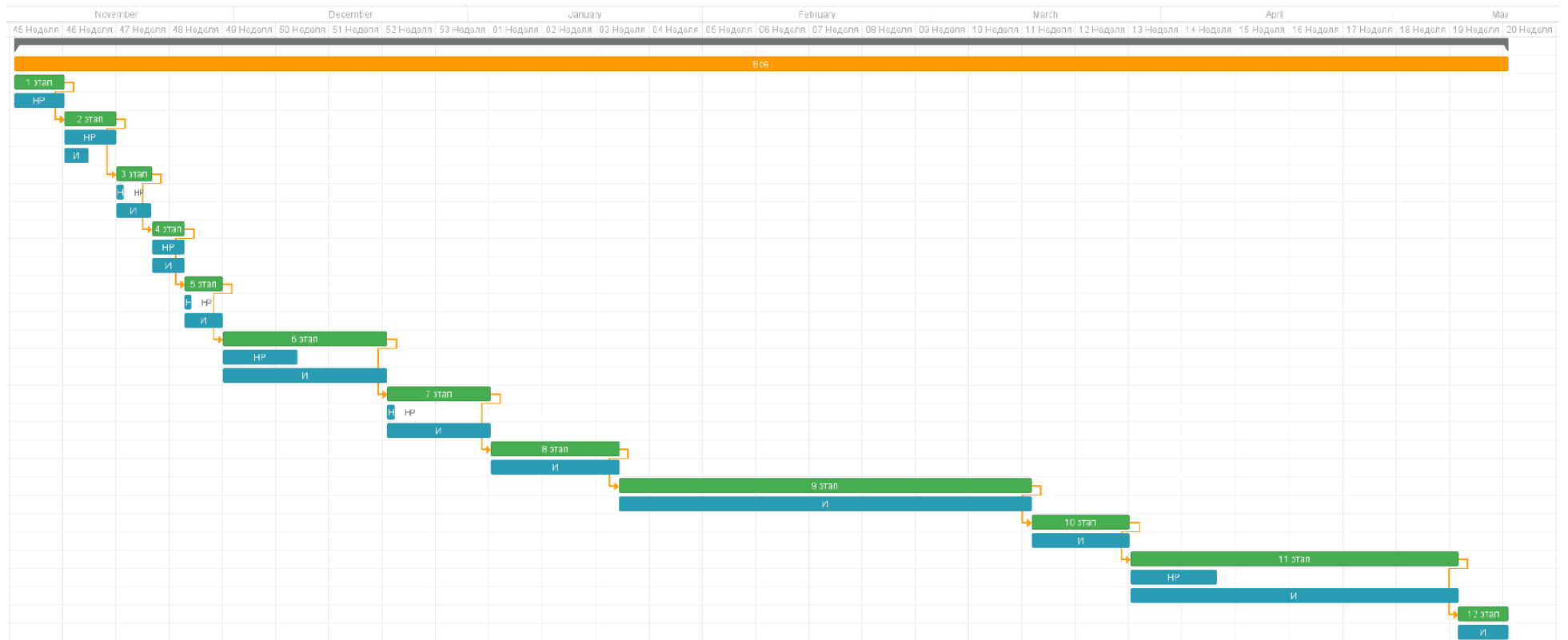


Рисунок 56 – Линейный график осуществления проекта

#### 4.1.2 Расчет накопления готовности проекта

Цель данного пункта – оценка текущих состояний (результатов) работы над проектом. Величина накопления готовности работы показывает, на сколько процентов по окончании текущего (*i*-го) этапа выполнен общий объём работ по проекту в целом.

Далее представлены обозначения:

- $TP_{\text{общ.}}$  – общая трудоёмкость проекта;
- $TP_i(TP_k)$  – трудоёмкость *i*-го (*k*-го) этапа проекта,  $i = \overline{1, I}$ ;
- $TP_{iH}$  – накопленная трудоёмкость *i*-го этапа проекта по его завершении;
- $TP_{ij}(TP_{kj})$  – трудоёмкость работ, выполняемых *j*-м участником на *i*-м этапе, здесь  $j = \overline{1, m}$  – индекс исполнителя, в данной работе  $m = 2$ .

Степень готовности определяется формулой 4.5.

$$CG_i = \frac{TP_i^H}{TP_{\text{общ.}}} = \frac{\sum_{k=1}^i TP_k}{TP_{\text{общ.}}} = \frac{\sum_{k=1}^i \sum_{j=1}^m TP_{kj}}{\sum_{k=1}^I \sum_{j=1}^m TP_{kj}}. \quad (4.5)$$

Применительно к таблице 4.2 величины  $TP_{ij}(TP_{kj})$  находятся в столбцах (6,  $j = 1$ ) и (7,  $j = 2$ ).  $TP_{\text{общ.}}$  равна сумме чисел из итоговых клеток этих столбцов. Расчёт  $TP_i(\%)$  и  $CG_i(\%)$  на основе этих данных содержится в таблице 4.3.

Таблица 4.3 – Трудозатраты на выполнение проекта

Этап	$TP_i, \%$	$CG_i, \%$
Постановка целей и задач, получение исходных данных	2,79	2,79
Составление и утверждение технического задания (ТЗ)	4,79	7,58
Подбор и изучение материалов по тематике	3,35	10,93
Разработка календарного плана	2,79	13,72
Обсуждение литературы	2,33	16,06
Проектирование пользовательских сценариев	14,12	30,18
Проектирование базы данных	6,36	36,54
Проектирование пользовательских интерфейсов	7,78	44,32
Разработка информационной системы	23,14	67,45

Этап	ТР <sub>i</sub> , %	СГ <sub>i</sub> , %
Тестирование информационной системы	5,38	72,84
Оформление пояснительной записки	24,37	97,21
Подведение итогов	2,79	100,0

## 4.2 Расчет сметы затрат на выполнение проекта

В состав затрат на создание проекта включается величина всех расходов, необходимых для реализации комплекса работ, составляющих содержание данной разработки. Расчёт сметной стоимости её выполнения производится по следующим статьям затрат:

- заработная плата;
- социальный налог;
- расходы на электроэнергию (без освещения);
- амортизационные отчисления;
- прочие, накладные расходы.

### 4.2.1 Расчет заработной платы

Данная статья расходов включает заработную плату научного руководителя и инженера (в его роли выступает исполнитель проекта), а также премии, входящие в фонд заработной платы. Расчёт основной заработной платы выполняется на основе трудоёмкости выполнения каждого этапа и величины месячного оклада исполнителя. Оклад инженера принимается равным окладу соответствующего специалиста низшей квалификации в ТПУ.

Среднедневная тарифная заработная плата (ЗП<sub>дн-т</sub>) рассчитывается по формуле.

$$\text{ЗП}_{\text{дн-т}} = \text{МО} / 20,5 \quad (4.6)$$

учитывающей, что в году 246 рабочих дней и, следовательно, в месяце в среднем 20,5 рабочих дня.

Для случая шестидневной рабочей недели в году 299 рабочих дней, и, следовательно, в месяце в среднем 24,92 рабочих дней.

$$ЗП_{\text{дн-т}} = MO/24,92 \quad (4.7)$$

Расчёты затрат на полную заработную плату приведены в таблице 4.4. Затраты времени по каждому исполнителю в рабочих днях с округлением до целого взяты из таблицы 4.2. Для учёта в её составе премий, дополнительной зарплаты и районной надбавки используется следующий ряд коэффициентов:  $K_{\text{ПР}} = 1,1$ ;  $K_{\text{доп.ЗП}} = 1,113$ ;  $K_{\text{р}} = 1,3$ . Таким образом, для перехода от тарифной (базовой) суммы заработка исполнителя, связанной с участием в проекте, к соответствующему полному заработку (зарплатной части сметы) необходимо первую умножить на интегральный коэффициент  $K_{\text{и}} = 1,1 \cdot 1,113 \cdot 1,3 = 1,59$ . Данный показатель рассчитан для пятидневной рабочей недели. Для шестидневной применяется другой коэффициент дополнительной заработной платы, равный 1,188, соответственно  $K_{\text{и}} = 1,1 \cdot 1,188 \cdot 1,3 = 1,70$

Таблица 4.4 – Расходы на заработную плату

Исполнитель	Оклад, руб./мес.	Среднедневная ставка, руб./раб.день	Затраты времени, раб.дни	Коэффициент	Фонд з/платы, руб.
НР	23 264,86	930,5944	22	1,7	34 780,48
И	7 864,11	382,1239067	91	1,59	55 344,8
<b>Итого:</b>					<b>90125,28</b>

#### 4.2.2 Расчет затрат на социальный налог

Затраты на единый социальный налог (ЕСН), включающий в себя отчисления в пенсионный фонд, на социальное и медицинское страхование, составляют 30% от полной заработной платы по проекту, т.е.  $C_{\text{соц}} = K_{\text{соц}} \cdot C_{\text{осн}} \cdot C_{\text{соц}} = C_{\text{зп}} \cdot 0,3$ . Итак, в нашем случае  $C_{\text{соц.}} = 90125,28 \cdot 0,3 = 27037,58$  руб.

#### 4.2.3 Расчет затрат на электроэнергию

Данный вид расходов включает в себя затраты на электроэнергию, потраченную в ходе выполнения проекта на работу используемого оборудования, рассчитываемые по формуле.

$$C_{\text{эл.об.}} = P_{\text{об}} \cdot t_{\text{об}} \cdot Ц_{\text{Э}} \quad (4.8)$$

где  $P_{\text{об}}$  – мощность, потребляемая оборудованием, кВт;

$\text{Цэ}$  – тариф на 1 кВт·час;

$t_{\text{об}}$  – время работы оборудования, час.

Для ТПУ  $\text{Цэ} = 5,257$  руб./кВт·час (с НДС).

Время работы оборудования вычисляется на основе итоговых данных таблицы 2 для инженера ( $T_{\text{рд}}$ ) из расчёта, что продолжительность рабочего дня равна 8 часов.

$$t_{\text{об}} = T_{\text{рд}} * K_t, \quad (4.9)$$

где  $K_t \leq 1$  – коэффициент использования оборудования по времени, равный отношению времени его работы в процессе выполнения проекта к  $T_{\text{рд}}$ , в данном случае приравнивается 0.6.

Мощность, потребляемая оборудованием, определяется по формуле 4.10.

$$P_{\text{об}} = P_{\text{ном.}} * K_c \quad (4.10)$$

где  $P_{\text{ном.}}$  – номинальная мощность оборудования, кВт;

$K_c \leq 1$  – коэффициент загрузки, зависящий от средней степени использования номинальной мощности. Для технологического оборудования малой мощности  $K_c = 1$ .

Расчёт затраты на электроэнергию для технологических целей приведён в таблице 4.5.

Таблица 4.5 – Затраты на электроэнергию технологическую

Наименование оборудования	Время работы оборудования $t_{\text{об}}$ , час	Потребляемая мощность $P_{\text{об}}$ , кВт	Затраты $\text{Эоб}$ , руб.
Персональный компьютер	728*0,6	0,3	574,06
<b>Итого:</b>			<b>574,06</b>

#### 4.2.4 Расчет амортизационных расходов

В данном пункте рассчитывается амортизация используемого оборудования за время выполнения проекта.

Используется формула 4.11.

$$C_{AM} = \frac{N_A * C_{OB} * t_{pf} * n}{F_D}, \quad (4.11)$$

где  $N_A$  – годовая норма амортизации единицы оборудования;

$C_{OB}$  – балансовая стоимость единицы оборудования с учётом ТЗР. При невозможности получить соответствующие данные из бухгалтерии она может быть заменена действующей ценой, содержащейся в ценниках, прейскурантах и т.п.;

$F_D$  – действительный годовой фонд времени работы соответствующего оборудования, берётся из специальных справочников или фактического режима его использования в текущем календарном году.

$t_{pf}$  – фактическое время работы оборудования в ходе выполнения проекта, учитывается исполнителем проекта;

$n$  – число задействованных однотипных единиц оборудования.

$$C_{AM} = \frac{0,33 * 52000 * 728 * 1}{1976} = 6322,11$$

#### 4.2.5 Расчет прочих расходов

В данном пункте отражены расходы на выполнение проекта, которые не учтены в предыдущих статьях, их следует принять равными 10% от суммы всех предыдущих расходов, т.е.

$$C_{проч.} = (C_{мат} + C_{зп} + C_{соц} + C_{эл.об.} + C_{ам} + C_{нп}) * 0,1 \quad (4.12)$$

Для данного примера это

$$C_{проч.} = (90125,28 + 27037,58 + 574,06 + 6322,11) * 0,1 = 12405,9 \text{ руб.}$$

#### 4.2.8 Расчет общей себестоимости разработки

Проведя расчёт по всем статьям сметы затрат на разработку, можно определить общую себестоимость научно-исследовательской работы.

Таблица 4.6 – Смета затрат на разработку проекта

Статья затрат	Условное обозначение	Сумма, руб.
Основная заработная плата	$C_{зп}$	90125,28
Отчисления в социальные фонды	$C_{соц}$	27037,58
Расходы на электроэнергию	$C_{эл.}$	574,06
Амортизационные отчисления	$C_{ам}$	6322,11
Прочие расходы	$C_{проч}$	124059,03
<b>Итого:</b>		<b>124059,03</b>



Таким образом, затраты на разработку составили  $C = 124059,03$  руб.

#### **4.2.9 Расчет прибыли**

Прибыль от реализации проекта в зависимости от конкретной ситуации (масштаб и характер получаемого результата, степень его определённости и коммерциализации, специфика целевого сегмента рынка и т.д.) может определяться различными способами. Из-за недостатка данных прибыль следует принять в размере от 5 до 20 % от полной себестоимости проекта. В данном случае она принята за 27292,99 руб. (20 %) от расходов на разработку проекта.

#### **4.2.10 Расчет НДС**

НДС составляет 18% от суммы затрат на разработку и прибыли. В данном примере это  $(124059,03 + 27292,99) \cdot 0,18 = 29476,43$  руб.

#### **4.2.11 Цена разработки НИР**

Цена равна сумме полной себестоимости, прибыли и НДС, в данном случае

$$C_{\text{НИР(КР)}} = 124059,03 + 27292,99 + 29476,43 = 193234,3 \text{ руб.}$$

### **4.3 Оценка экономической эффективности проекта**

Рассматриваемая работа разрабатывалась специально для применения в АО «Информационные спутниковые системы» имени академика М.Ф. Решетнёва».

Количественная оценка невозможна ввиду недостатка информации и компетенции. Данная работа позволяет сократить временные затраты на проектирование и подготовку производства бортового программного обеспечения и обеспечит поддержку на ранних этапах жизненного цикла. Она позволит уменьшить время работы высококлассных и высокооплачиваемых специалистов за счёт возможности использовать наработки уже сформированной ранее документации и обнаружения специфичных для такой сферы ошибок (неправильно указанные диапазоны выделяемых ресурсов, попадание излишнего количества ресурсов в резерв и так далее).

В виду характера работы возможно сокращение цикла производства и уменьшение количества ошибок. Это достигается за счёт того, что пользователю проще формировать документы, и он меньше тратит время на необходимые бюрократические издержки, например, на соби́рание подписей в случае изменения документов из-за какой-либо ошибки, выявленной на более поздних этапах проектирования бортового программного обеспечения.

Можно сказать, что в данной работе оценка экономического эффекта приобретает форму экономии ресурсов и носит в первую очередь внутрисистемный характер, внесистемный характер оценить возможно, но это потребует проведения большого количества сложных исследований.

#### **4.3.1 Оценка научно-технического уровня НИР**

Научно-технический уровень характеризует влияние проекта на уровень и динамику обеспечения научно-технического прогресса в данной области. Для оценки научной ценности, технической значимости и эффективности, планируемых и выполняемых НИР, используется метод балльных оценок. Балльная оценка заключается в том, что каждому фактору по принятой шкале присваивается определённое количество баллов. Обобщённую оценку проводят по сумме баллов по всем показателям. На её основе делается вывод о целесообразности НИР.

Сущность метода заключается в том, что на основе оценок признаков работы определяется интегральный показатель (индекс) её научно-технического уровня по формуле 4.13.

$$K_{НТУ} = \sum_{i=1}^3 R_i \cdot n_i, \quad (4.13)$$

где  $I_{нту}$  – интегральный индекс научно-технического уровня;

$R_i$  – весовой коэффициент  $i$ -го признака научно-технического эффекта;

$n_i$  – количественная оценка  $i$ -го признака научно-технического эффекта, в баллах.

Для используемого в пособии примера частные оценки уровня  $n_i$  и их краткое обоснование даны в таблице 4.7.

Таблица 4.7 – Оценки научно-технического уровня НИР

Значимость	Фактор НТУ	Уровень фактора	Выбранный балл	Обоснование выбранного балла
0,4	Уровень новизны	Относительно новая	4	Уменьшит потери времени пользователям на выполнение рутинных задач
0,1	Теоретический уровень	Разработка программы	1	Создания нового вида приложения для управления документами
0,5	Возможность реализации	В течение первых лет	10	Возможность реализации в виде приложения в течение 1-2 лет

Отсюда интегральный показатель научно-технического уровня для текущего проекта составляет:

$$K_{НТУ} = 0,5 \cdot 4 + 0,3 \cdot 6 + 0,2 \cdot 12 = 6,27$$

По полученному интегральному показателю данный проект имеет средний уровень научно-технического эффекта.

## 5 СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ

### Аннотация

Представление о понятии «Социальная ответственность» возможно получить из международного стандарта ИС CSR-08260008000:2011 «Социальная ответственность организации. Требования» [15], который был утвержден и введен в действие Международным Комитетом по корпоративной социальной ответственности (ИС CSR), протокол №2 от 03 марта 2011 г.

В настоящем стандарте используются термины и определения, такие как:

**социальная ответственность** (social responsibility) – ответственность организации за воздействие ее решений и деятельности на общество и окружающую среду через прозрачное и этическое поведение, которое:

- содействует устойчивому развитию, включая здоровье и благосостояние общества;
- учитывает ожидания заинтересованных сторон;
- соответствует применяемому законодательству и согласуется с международными нормами поведения;
- интегрировано в деятельность всей организации и применяется в ее взаимоотношениях.

### Введение

Научно-исследовательская работа выполнялась в помещении, далее офис, находящемся на кафедре «Автоматики и компьютерных систем», десятого корпуса Томского Политехнического Университета, в аудитории 117а.

Площадь, приходящаяся на одно рабочее место пользователя ПК с ЭЛТ-монитором должна составлять не менее 6 м<sup>2</sup>, с монитором на базе плоских дискретных экранов – 4,5 м<sup>2</sup>, что позволяет расположить технические средства на безопасном расстоянии до пользователя.

Для данной рабочей зоны необходимо проанализировать следующие факторы. К вредным факторам относятся: микроклимат, шум, электромагнитные поля, освещение.

К опасным факторам рабочей зоны относятся: опасность возникновения пожара и опасность поражения электрическим током.

Чрезвычайные ситуации характерные для данного объекта: пожар.

Вопросы, относящиеся к организации и охране труда при работе за компьютером, регулируются:

- Трудовым кодексом Российской Федерации,
- СанПиН 2.2.2/2.4.1340-03 «Гигиенические требования к ПЭВМ и организации работы»,
- Инструкцией по охране труда при работе на ПК.

### 5.1. Производственная безопасность

Описание выявленных вредных и опасных факторов приведено в таблице 5.1.

Таблица 5.1. Опасные и вредные факторы

Источник фактора, наименование видов работ	Факторы (по ГОСТ 12.0.003-74)		Нормативные документы
	Вредные	Опасные	
1.. Выполнение работ по разработке информационной системы. 2. Эксплуатация информационной системы проектантом бортового программного обеспечения.	1. Отклонение показателей микроклимата в помещении. 2. Недостаточное освещение рабочего места 3. Превышение уровней шума.	1. Поражение электрическим током. 2. Возникновение пожара.	СанПиН 2.2.4.548-96. «Гигиенические требования к персональным электронно-вычислительным машинам и организации работы» [16] СНиП 23-05-95. «Естественное и искусственное освещение» [17] ГОСТ 12.1.003-83. «ССБТ. Шум. Общие требования безопасности» [19] ГОСТ 12.1.009-76. «Электробезопасность. Термины и определения» [20] ГОСТ 12.1.019-79. ССБТ «Электробезопасность. Общие требования и номенклатура видов защиты» [21] СНиП 21-01-97. «Пожарная безопасность зданий и сооружений» [22] НПБ 105-2003. «Определение категорий помещений, зданий

			и наружных установок по взрывопожарной и пожарной опасности» [23]
--	--	--	---

### 5.1.1. Микроклимат рабочего помещения.

**Микроклимат производственных (рабочих) помещений** – климат внутренней среды этих помещений, который определяется действующими на организм человека сочетаниями температуры, влажности и скорости движения воздуха, а также интенсивности теплового излучения от нагретых поверхностей [18].

Мероприятия по доведению микроклиматических показателей до нормативных значений включаются в комплексные планы предприятий по охране труда. Для создания благоприятных условий работы, соответствующих физиологическим потребностям человеческого организма, санитарные нормы устанавливают оптимальные и допустимые метеорологические условия в рабочей зоне помещения (таблицы 5.2–5.3) [16]. Выполняемая работа относится к категории **легкая** (1б).

Таблица 5.2 – Оптимальные величины показателей микроклимата на рабочих местах производственных помещений (СанПиН 2.2.4.548-96)

Период года	Температура воздуха, °С	Температура поверхностей, °С	Относительная влажность воздуха, %	Скорость движения воздуха, м/с
Холодный	21 - 23	20 – 24	60-40	0,1
Теплый	23-25	22-26	60-40	0,1

Таблица 5.3 – Допустимые величины показателей микроклимата на рабочих местах производственных помещений (СанПиН 2.2.4.548-96)

Период года	Температура воздуха, °С		Температура поверхностей, °С	Относительная влажность воздуха, %	Скорость движения воздуха, м/с	
	диапазон ниже оптимальных величин	диапазон выше оптимальных величин			для диапазона температур воздуха ниже оптимальных величин, не более	для диапазона температур воздуха выше оптимальных величин, не более
Холодный	19,0 - 20,9	23,1 - 24,0	18,0 - 25,0	15 - 75	0,1	0,2

Период года	Температура воздуха, °С		Температура поверхностей, °С	Относительная влажность воздуха, %	Скорость движения воздуха, м/с	
	диапазон ниже оптимальных величин	диапазон выше оптимальных величин			для диапазона температур воздуха ниже оптимальных величин, не более	для диапазона температур воздуха выше оптимальных величин, не более
Теплый	20,0 - 21,9	24,1 - 28,0	19,0 - 29,0	15 - 75	0,1	0,3

В данном случае температура воздуха и температура поверхностей составляют 21<sup>0</sup>С и 20<sup>0</sup>С при относительной влажности 50% в холодный период года; 24<sup>0</sup>С и 23<sup>0</sup>С при относительной влажности воздуха 55% в теплый период года, что соответствует нормам СанПиН 2.2.4.548-96.

### 5.1.2. Производственное освещение

Освещение – получение, распределение и использование световой энергии для обеспечения благоприятных условий видения предметов и объектов. Оно влияет на настроение и общее самочувствие, определяет эффективность труда. Нерационально организованное освещение может явиться причиной травматизма: плохо освещенные опасные зоны, слепящие источники света и блики от них, резкие тени и пульсации освещенности ухудшают видимость и могут вызвать неадекватное восприятие наблюдаемого объекта [17].

Длина рассматриваемого помещения составляет 7 метров, ширина – 4 м, высота – 3,5 м. Высота рабочей поверхности 0,7 м.

В помещении установлен светильник типа ОД 2-30, характеристики которого приведены в таблице 5.4.

Таблица 5.4 – Характеристика светильника ОД 2-30

Мощность, Вт	Размеры, мм			Световой поток, лм
	Длина	Ширина	Высота	
2 x 30	933	204	156	1800

План светильников показан ниже.

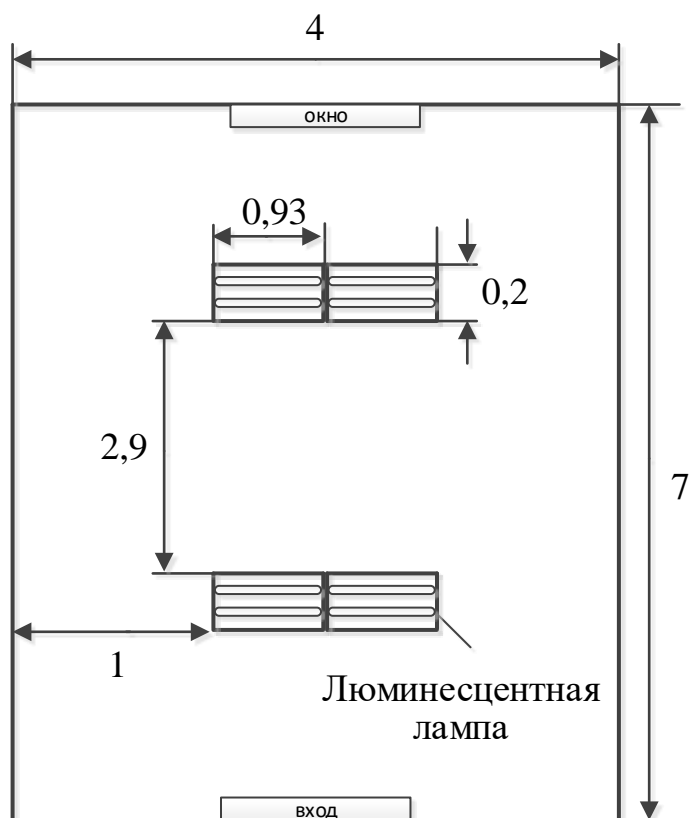


Рисунок 57 – План размещения общего освещения (вид сверху)

Расчёт общего равномерного искусственного освещения горизонтальной рабочей поверхности выполняется методом коэффициента светового потока, учитывающим световой поток, отражённый от потолка и стен.

Коэффициент  $Z$  (отношение средней освещённости к минимальной) примем равным  $Z = 1.1$ . Коэффициент запаса определяется по таблице [17] в зависимости от запылённости помещения, в нашем случае  $K = 1.5$  (коэффициент запаса). Коэффициент использования, выражается отношением светового потока, падающего на расчетную поверхность, к суммарному потоку всех ламп и исчисляется в долях единицы. Он зависит от характеристик светильника, размеров помещения, окраски стен и потолка, характеризуемых коэффициентами отражения от стен ( $R_c$ ) и потолка ( $R_{п}$ ), значение коэффициентов  $R_c$  и  $R_{п}$  определим по таблице из СНиП 23-05-95 [17],  $R_c = 70\%$ ,  $R_{п} = 50\%$ .



Значение  $\eta$  определим по таблице коэффициентов использования различных светильников из СНиП 23-05-95 [17]. Для этого вычислим индекс помещения по формуле:

$$I = \frac{S}{h \cdot (a+b)}, \quad (5.1)$$

Площадь помещения составляет 28 м<sup>2</sup>.

$$I = \frac{28}{2,64 \cdot (7+4)} = 0,96$$

Зная индекс помещения  $I$ ,  $R_c$  и  $R_p$ , определим коэффициент использования светового потока из таблицы, взятой из СНиП 23-05-95,  $\eta = 0,49$  [17]. Стандартный световой поток возьмём из таблицы, зная мощность и тип люминесцентной лампы, получаем  $\Phi_{ст} = 1060$ .

$$E_f = (N * n * \Phi_{ст} * \eta) / (S * k * z) \quad (5.2)$$

$$E_f = (8 * 2 * 1060 * 0,49) / (28 * 1,5 * 1,1) = 180 \text{ лк}$$

По нормам, установленным СНиП 23-05-95, минимальная освещённость рабочих поверхностей в офисных помещениях для работ средней точности при общем освещении должна быть равна 200 лк [17]. Отклонение от минимального значения освещенности составляет 10%.

### 5.1.3. Производственные шумы

Шум – это совокупность различных звуков, возникающих в процессе производства и неблагоприятно воздействующих на организм [18].

Шум может привести к нарушениям слуха (в случае постоянного нахождения при шуме более 85 децибел(dB)), может являться фактором стресса и повысить систолическое кровяное давление.

Дополнительно, он может способствовать несчастным случаям, маскируя предупреждающие сигналы и мешая сконцентрироваться.

Для рассматриваемого помещения основными источниками шума являются персональные компьютеры, кондиционер и вытяжные вентиляторы на окнах.

Нормативным документом, регламентирующим уровни шума для различных категорий рабочих мест служебных помещений, является ГОСТ 12.1.003-83 «ССБТ. Шум. Общие требования безопасности» [19].

Помещения, в которых для работы используются ПК не должны граничить с помещениями, в которых уровни шума превышают нормируемые значения.

В помещениях, оборудованных ПК, которые являются основным источником шума при выполнении данных видов работ, уровень шума на рабочем месте не должен превышать 50 дБА [19].

#### **5.1.4. Электробезопасность**

Электробезопасность – система организационных и технических мероприятий и средств, обеспечивающих защиту людей от вредного и опасного для жизни воздействия электрического тока, электрической дуги, электромагнитного поля и статического электричества [20].

Опасное и вредное воздействия на людей электрического тока и электрической дуги проявляются в виде электротравм и профессиональных заболеваний.

Помещение, где расположены персональные вычислительные машины, относится к помещениям без повышенной опасности [21], так как отсутствуют следующие факторы:

- сырость;
- токопроводящая пыль;
- токопроводящие полы;
- высокая температура;
- возможность одновременного прикосновения человека к имеющим соединение с землёй металлоконструкциям зданий, технологическим аппаратам и механизмам, металлическим корпусам электрооборудования.

К мероприятиям по предотвращению возможности поражения электрическим током следует отнести [21]:

- при производстве монтажных работ необходимо использовать только исправный инструмент, аттестованный службой КИПиА;
- с целью защиты от поражения электрическим током, возникающим между корпусом приборов и инструментом при пробое сетевого напряжения на корпус, корпуса приборов и инструментов должны быть заземлены;
- при включенном сетевом напряжении работы на задней панели должны быть запрещены;
- все работы по устранению неисправностей должен производить квалифицированный персонал;
- необходимо постоянно следить за исправностью электропроводки.

Перед началом работы следует убедиться в отсутствии свешивающихся со стола или висящих под столом проводов электропитания, в целостности вилки и провода электропитания, в отсутствии видимых повреждений аппаратуры и рабочей мебели.

Токи статического электричества, наведенные в процессе работы компьютера на корпусах монитора, системного блока и клавиатуры, могут приводить к разрядам при прикосновении к этим элементам. Такие разряды опасности для человека не представляют, но могут привести к выходу из строя компьютера. Для снижения величин токов статического электричества используются нейтрализаторы, местное и общее увлажнение воздуха, использование покрытия полов с антистатической пропиткой [18].

### **5.1.5. Пожарная безопасность**

Пожарная безопасность – комплекс организационных и технических мероприятий, направленных на обеспечение безопасности людей, на предотвращение пожара, ограничение его распространения, а также на создание условий для успешного тушения пожара [22].

Рабочее помещение, в котором производится работа по выполнению ВКР по пожарной и взрывной опасности относят к категории В [23].

К противопожарным мероприятиям в помещении относят следующие мероприятия [18]:

1) помещение должно быть оборудовано: средствами тушения пожара (огнетушителями, ящиком с песком, стендом с противопожарным инвентарем); средствами связи; должна быть исправна электрическая проводка осветительных приборов и электрооборудования;

2) каждый сотрудник должен знать место нахождения средств пожаротушения и средств связи; помнить номера телефонов для сообщения о пожаре; уметь пользоваться средствами пожаротушения.

Помещение обеспечено средствами пожаротушения в соответствии с нормами [22]:

1) пенный огнетушитель ОП-10 – 1 шт.

2) углекислотный огнетушитель ОУ-5 – 1 шт.

При невозможности самостоятельно потушить пожар необходимо вызвать пожарную команду, после чего поставить в известность о случившемся инженера по техники безопасности.

Вынужденная эвакуация при пожаре протекает в условиях нарастающего действия опасных факторов пожара. Кратковременность процесса вынужденной эвакуации достигается устройством эвакуационных путей и выходов, число, размеры и конструктивно-планировочные решения которых регламентированы строительными нормами СНиП 2.01.02-85.

Помещение и этаж оборудованы следующими средствами оповещения:

- световая индикация в коридорах этажа;
- звуковая индикация в виде громкоговорителя;
- пассивными датчиками задымленности.

## **5.2. Экологическая безопасность**

Охрана окружающей среды сводится к устранению отходов бытового мусора и отходам жизнедеятельности человека. В случае выхода из строя ПК,

они списываются и отправляются на специальный склад, который при необходимости принимает меры по утилизации списанной техники и комплектующих [24].

На сегодняшний день одним из самых распространенных источников ртутного загрязнения являются вышедшие из эксплуатации люминесцентные лампы. Каждая такая лампа, кроме стекла и алюминия, содержит около 60 мг ртути. Поэтому отслужившие свой срок люминесцентные лампы, а также другие приборы, содержащие ртуть, представляют собой опасный источник токсичных веществ.

В целом, утилизация ламп предполагает передачу использованных ламп предприятиям – переработчикам, которые с помощью специального оборудования перерабатывают вредные лампы в безвредное сырье – сорбент, которое в последующем используют в качестве материала для производства, например, тротуарной плитки.

Под хранением отходов понимается временное размещение их в специально отведенных для этого местах или объектах до их утилизации. Отработанные люминесцентные лампы, согласно Классификатору отходов ДК 005-96, утвержденному приказом Госстандарта № 89 от 29.02.96 г., относятся к отходам, которые сортируются и собираются отдельно, поэтому утилизация люминесцентных ламп и их хранение должны отвечать определенные требованиям [25].

### **5.3. Безопасность в чрезвычайных ситуациях**

В данном случае на объекте (офис) могут возникать чрезвычайные ситуации (ЧС) следующего характера:

- техногенные;
- экологические;
- природные.

Наиболее типичной ЧС для помещения, котором производится выполнение работы, является пожар. Данная ЧС может произойти в случае

замыкания электропроводки оборудования, обрыву проводов, не соблюдению мер пожаробезопасности и т.д.

Для того чтобы избежать возникновения пожара необходимо проводить следующие профилактические работы, направленные на устранение возможных источников возникновения пожара:

- периодическая проверка проводки;
- отключение оборудования при покидании рабочего места;
- проведение инструктажа работников о пожаробезопасности.

Чтобы увеличить устойчивость офисного помещения к ЧС необходимо устанавливать системы противопожарной сигнализации, реагирующие на дым и другие продукты горения, установка огнетушителей, обеспечить офис и проинструктировать рабочих о плане эвакуации из офиса, а также назначить ответственных за эти мероприятия. Два раза в год (в летний и зимний период) проводить учебные тревоги для отработки действий при пожаре. В ходе осмотра офисного помещения были выявлены системы, сигнализирующие о наличии пожара или задымленности помещения и наличие огнетушителей.

В случае возникновения ЧС как пожар, необходимо предпринять меры по эвакуации персонала из офисного помещения в соответствии с планом эвакуации (Рисунок 58). При отсутствии прямых угроз здоровью и жизни произвести попытку тушения возникшего возгорания огнетушителем. В случае потери контроля над пожаром, необходимо эвакуироваться вслед за сотрудниками по плану эвакуации и ждать приезда специалистов, пожарников. При возникновении пожара должна сработать система пожаротушения, издав предупредительные сигналы, и передав на пункт пожарной станции сигнал о ЧС, в случае если система не сработала, по каким-либо причинам, необходимо самостоятельно произвести вызов пожарной службы по телефону 101, сообщить место возникновения ЧС и ожидать приезда специалистов.

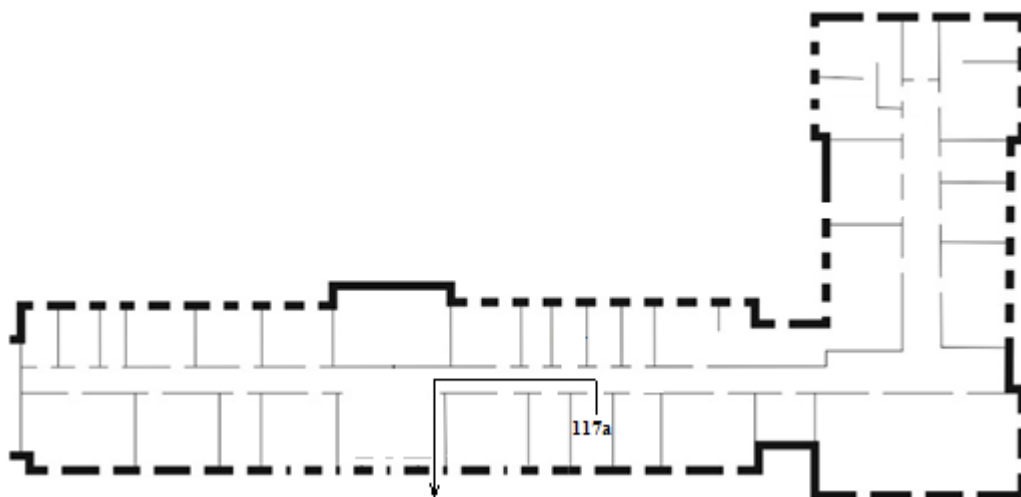


Рисунок 58 – План эвакуации при пожаре и других ЧС из помещений учебного корпуса № 10, пр. Ленина, 1 этаж.

#### **5.4. Правовые и организационные вопросы обеспечения безопасности**

Требования к организации рабочих мест пользователей:

- рабочее место должно быть организовано с учетом эргономических требований согласно ГОСТ 12.2.032-78 «ССБТ. Рабочее место при выполнении работ сидя. Общие эргономические требования» [26] и ГОСТ 12.2.061-81 «ССБТ. Оборудование производственное. Общие требования безопасности к рабочим местам» [27];

- конструкция рабочей мебели (рабочий стол, кресло, подставка для ног) должна обеспечивать возможность индивидуальной регулировки соответственно росту пользователя и создавать удобную позу для работы. Вокруг ПК должно быть обеспечено свободное пространство не менее 60-120см;

- на уровне экрана должен быть установлен оригинал-держатель.

Далее представлены требования к рабочему месту (Рисунок 59).

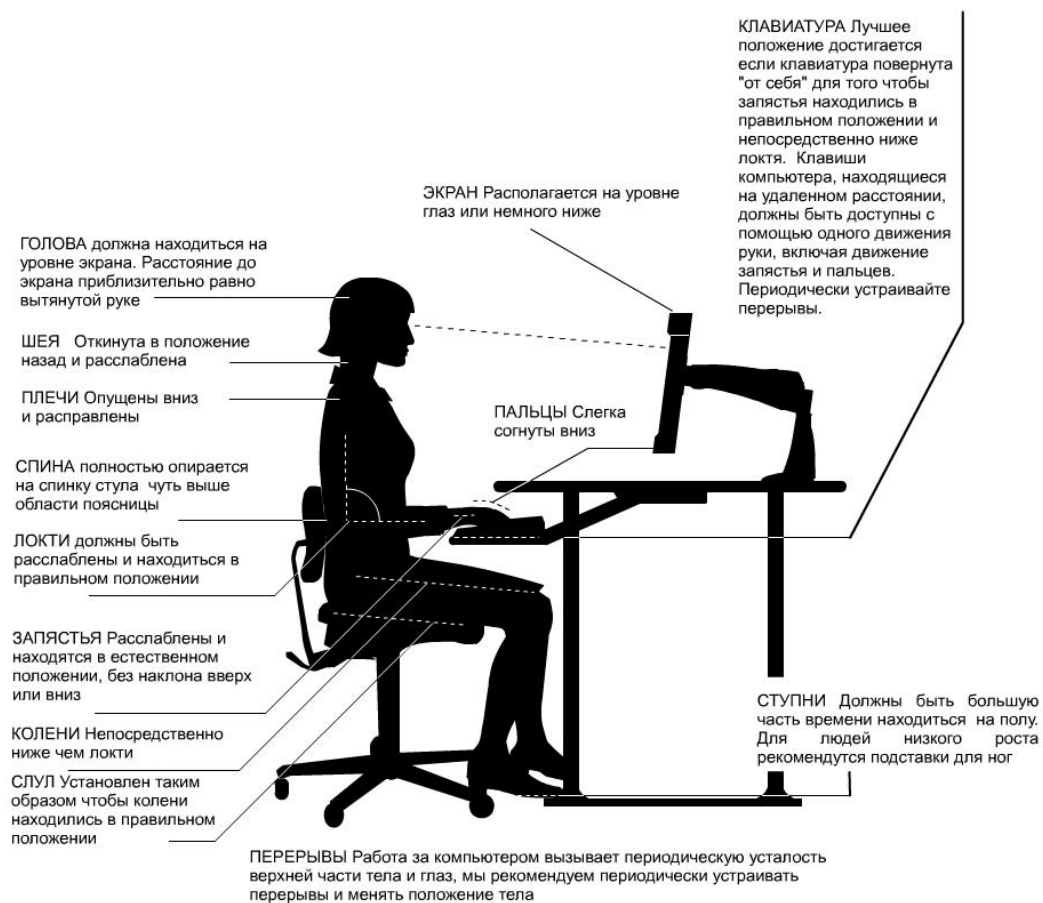


Рисунок 59 – Организация рабочего места.

В соответствии с государственными стандартами и правовыми нормами обеспечения безопасности предусмотрена рациональная организация труда в течение смены, которая предусматривает:

- длительность рабочей смены не более 8 часов;
- установление двух регламентируемых перерывов (не менее 20 минут после 1-2 часов работы, не менее 30 минут после 2 часов работы);
- обеденный перерыв не менее 40 минут.

Обязательно предусмотрен предварительный медосмотр при приеме на работу и периодические медосмотры.

Каждый сотрудник должен пройти инструктаж по технике безопасности перед приемом на работу и в дальнейшем, должен быть пройден инструктаж по электробезопасности и охране труда.



## **ЗАКЛЮЧЕНИЕ**

В результате выполнения магистерской диссертации был разработан новый функционал, расширяющий возможности информационной системы, а также была подробно описана архитектура системы с учетом механизмов и логики работы ее составных частей.

Реализованные возможности САПР БПО по работе с тестами являются базовым началом для пересмотра иерархической структуры всех объектов системы, установления их связей с тестами и определения функционала, необходимого специалистам на этапах тестирования и верификации БПО космических аппаратов.

Наиболее актуальными задачами по доработке САПР БПО на данный момент являются определение предметных связей между тестами и всеми объектами системы, формулирование соответствующих пользовательских сценариев и реализация по вышеописанному шаблону последующих доработок.

Вопрос подготовки к внедрению САПР БПО в состав ТКПП БПО возможен только после завершения процесса интеграции тестов и сбора обратной связи, может происходить параллельно с определением новых направлений усовершенствования САПР БПО.

Включение САПР БПО в состав ТКПП БПО позволит сделать еще один шаг к созданию единого информационного пространства, автоматизирующего процессы создания, контроля и сопровождения БПО.

## СПИСОК ПУБЛИКАЦИЙ СТУДЕНТА

1. Готовится к публикации: Баглаева Е. А., Цапко С. Г. Система автоматизированного проектирования бортового программного обеспечения космического аппарата //Инженерия для освоения космоса: сборник научных трудов V Всероссийского молодежного форума с международным участием, г. Томск, 18-20 апреля 2017 г.—Томск, 2017. – 2017.

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Микрин Е. А. и др. Принципы построения бортовых комплексов управления автоматических космических аппаратов //Проблемы управления. – 2004. – №. 3. – С. 62–66.
2. Бровкин А. Г. и др. Бортовые системы управления космическими аппаратами./Под ред. проф //АС Сырова М.: Изд-во МАИ-ПРИНТ. – 2010.
3. Антамошкин А. Н., Колташев А. А. Технологические аспекты создания бортового программного обеспечения спутников связи //Вестник Сибирского государственного аэрокосмического университета им. академика МФ Решетнева. – 2005. – №. 3. – С. 93–95.
4. Колташев А.А., Кочура С.Г. Технология создания и сопровождения бортового программного обеспечения спутников связи, навигации и геодезии: современное состояние // Научные технологии. – 2014. – № 9. – С. 24–27.
5. Антамошкин А. Н. и др. Технология разработки бортового программного обеспечения: управление работами и объектами //Вестник Сибирского государственного аэрокосмического университета им. академика МФ Решетнева. – 2008. – №. 1. – С. 4–8.
6. Колташев А. А. Основные принципы системного тестирования и подтверждения бортового программного обеспечения спутников //Вестник Сибирского государственного аэрокосмического университета им. академика МФ Решетнева. – 2010. – №. 1. – С. 4–7.
7. Дьяченко А.Б., Ерохов П.М., Колташев А.А. Прагматичный подход к автоматизации проектирования и тестирования бортового программного обеспечения спутников // Системный анализ, управление и навигация: Тезисы докладов 17-й Междунар. науч. конф. М.: Изд-во МАИ. – 2012. – С. 76–77.
8. Система менеджмента качества. Обеспечение бортовое программное. Управление проектированием, изготовлением и

сопровождением // СТП 154-123–2014. – Железногорск: АО «ИСС» им. акад. М.Ф. Решетнева, 2014. – 72 с.

9. Цапко Г. П., Мартынов Я. А. Единая информационная среда создания и сопровождения бортового программного обеспечения спутников навигации и связи // Доклады ТУСУРа. – 2015. – Т. 37. – №. 3. С. 97–102.

10. Misheski D. J., Nock C. M. Object oriented framework mechanism for data transfer between a data source and a data target : пат. 5915252 США. – 1999.

11. O'Neil E. J. Object/relational mapping 2008: hibernate and the entity data model (edm) // Proceedings of the 2008 ACM SIGMOD international conference on Management of data. – ACM, 2008. – С. 1351-1356.

12. Mateos C. et al. Detecting WSDL bad practices in code–first Web Services // International Journal of Web and Grid Services. – 2011. – Т. 7. – №. 4. – С. 357-387.

13. Bugnion L. MVVM Light Toolkit // GalaSoft <http://www.galasoft>. – 2013.

14. Sorensen E., Mikailesc M. Model-View-ViewModel (MVVM) Design Pattern using Windows Presentation Foundation (WPF) Technology // MegaByte Journal. – 2010. – Т. 9. – №. 4. – С. 1-19.

15. Международный стандарт «Социальная ответственность организации. Требования». 2011. URL: [www.ksovok.com/doc/ic\\_csr\\_08260008000\\_ru.doc](http://www.ksovok.com/doc/ic_csr_08260008000_ru.doc) (дата обращения 11.03.2017)

16. СанПиН 2.2.4.548-96. «Гигиенические требования к персональным электронно-вычислительным машинам и организации работы».

17. СНиП 23-05-95. «Естественное и искусственное освещение».

18. Назаренко, Ольга Брониславовна. Безопасность жизнедеятельности : учебное пособие / О. Б. Назаренко, Ю. А. Амелькович; Национальный исследовательский Томский политехнический университет (ТПУ). — 3-е изд., перераб. и доп. — Томск: Изд-во ТПУ, 2013. — 177 с

19. ГОСТ 12.1.003-83. «ССБТ. Шум. Общие требования безопасности».

20. ГОСТ 12.1.009-76. «Электробезопасность. Термины и определения»
21. ГОСТ 12.1.019-79. ССБТ «Электробезопасность. Общие требования и номенклатура видов защиты».
22. СНиП 21-01-97. «Пожарная безопасность зданий и сооружений».
23. НПБ 105-2003. «Определение категорий помещений, зданий и наружных установок по взрывопожарной и пожарной опасности».
24. ГОСТ 17.4.3.04-85. «Охрана природы. Почвы. Общие требования к контролю и охране от загрязнения».
25. СанПиН 2.1.7.2790-10 «Санитарно-эпидемиологические требования к обращению с медицинскими отходами».
26. ГОСТ 12.2.032-78. «ССБТ. Рабочее место при выполнении работ сидя. Общие эргономические требования».
27. ГОСТ 12.2.061-81. «ССБТ. Оборудование производственное. Общие требования безопасности к рабочим местам»

## Приложение А

### Раздел 3

#### Программа и методика испытаний

---

Студент:

Группа	ФИО	Подпись	Дата
8ВМ5Г	Баглаева Елена Алексеевна		

Консультант кафедры АиКС:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Ассистент	Ефремов Александр Александрович	-		

Консультант – лингвист кафедры ИяИК:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Старший преподаватель	Денико Роман Викторович	-		

### 3 THE TESTING PROGRAMME AND PROCEDURE

The testing object is updated and expanded functionality of SAPR BPO, defined in the following composition:

Database:

- sapr\_bpo\_iss database, managed by MS SQL Server 2014 Express;

Server side:

- service libraries (dll-libraries);
- WCF services;

Client side:

- the user interface of the SAPR BPO client application.

The objective of the database testing is the verification of the correctness of the create, read and update operations in the data access layer, using ORM technology provided by EntityFramework.

The objectives of the server side testing are:

- checking the correctness of the create, read and update operations in the business logic layer for tests and related entities;
- checking the correctness of the create, read and update operations in the data presentation layer for tests and related entities.

The purpose of the client side testing is to check the correctness of the create, read and update operations using the functional operations of the client application.

Tests are being carried out using manual testing, which provides end-to-end testing of all SAPR BPO layers.

The entry point of testing is the user interface of the SAPR BPO client application, which receives data from WCF services located on the application server. In order to perform the requested operations, WCF services call the business logic libraries, which use the data access layer implemented by the repository set and the EntityFramework technology. Thus, all layers of the application are being tested.

### 3.1 The list of performed tests

The list of performed tests is divided into three groups: checking the functionality of satellite test management, checking the functionality of system test management, checking the traceability matrix of the functional requirements and the tests.

The description of the performed tests is given below.

#### 3.1.1 Satellite test management

As a part of the satellite test management, the user is provided create, read, update and delete operations for satellite tests.

To navigate to the "Satellite tests" tab, the user should open the satellite folder in the object tree and select the "Tests" folder inside the satellite folder. (Figure 60).

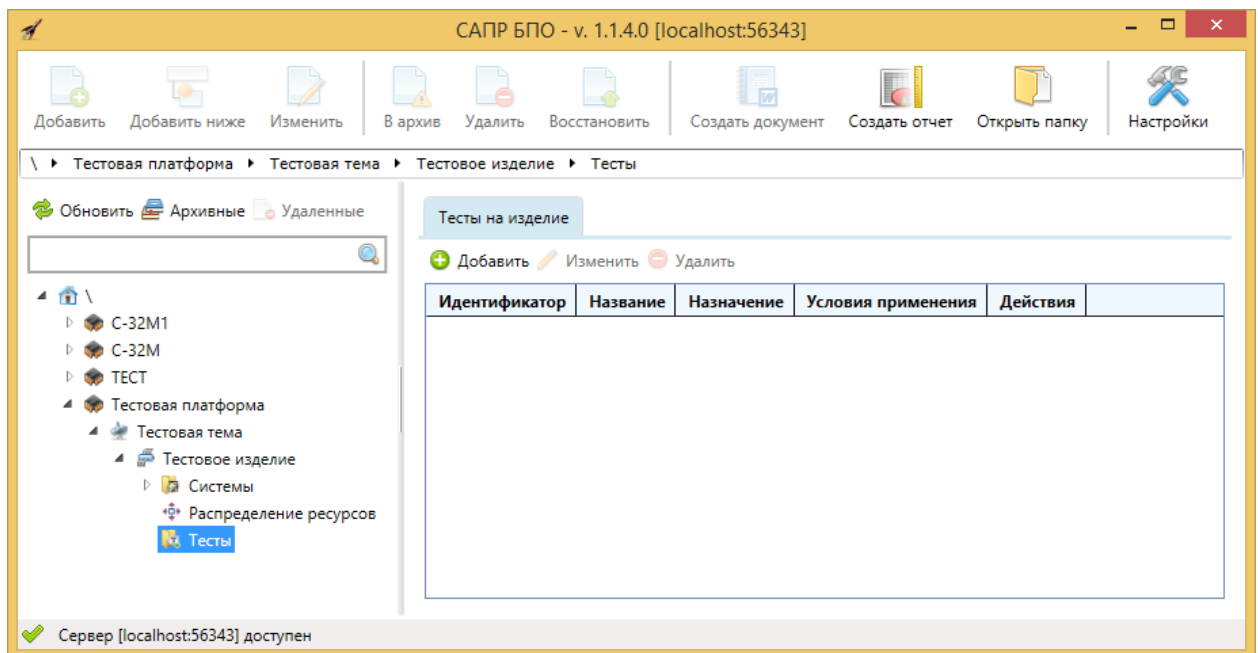


Figure 60 – The "Satellite tests" tab.

##### 3.1.1.1 Creating a satellite test

To create a satellite test, the user should click the "Add" button, fill the fields of the test in the displayed window (Figure 61) and click the "Ok" button.



Figure 61 – Creating a satellite test.

The created test should appear in the list of tests for the selected satellite (Figure 62).

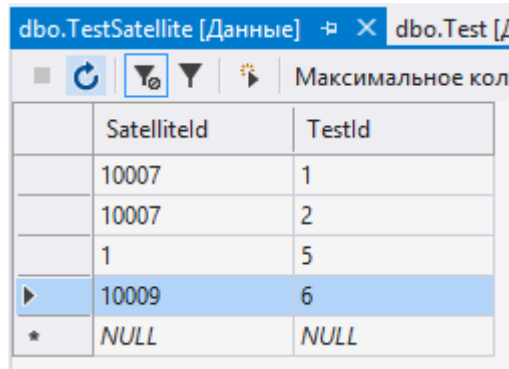
Идентификатор	Название	Назначение	Условия применения	Действия
ТИ-001	Тест на изделие 1	Назначение 001	Условие 001	Действие 001

Figure 62 – The result of the satellite test creation.

Besides, the entry of the created test should appear in the Test table (Figure 63), the entry about the links between the test and the satellite should be displayed in the TestSatellite table (Figure 64).

Testid	Title	Identifier	Purpose	Condition	Actions	ParentId	Deleted	OnEditing	UserId
1	Тест 1	T01	1	1	1	NULL	2	False	1
2	Тест 2-1	T02	2	2	2	NULL	0	False	NULL
3	Тест 1.1	T1-1	Применение	Условия	Действия	NULL	0	False	NULL
4	Тест 1	T11	Проверка	У1	Д1	NULL	0	False	1
5	Тест на издели...	ТИ1-1	Тестирование ...	У2	Д2	NULL	0	False	1
6	Тест на издели...	ТИ-001	Назначение 001	Условие 001	Действие 001	NULL	0	False	1
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Figure 63 – The entry in the Test table.



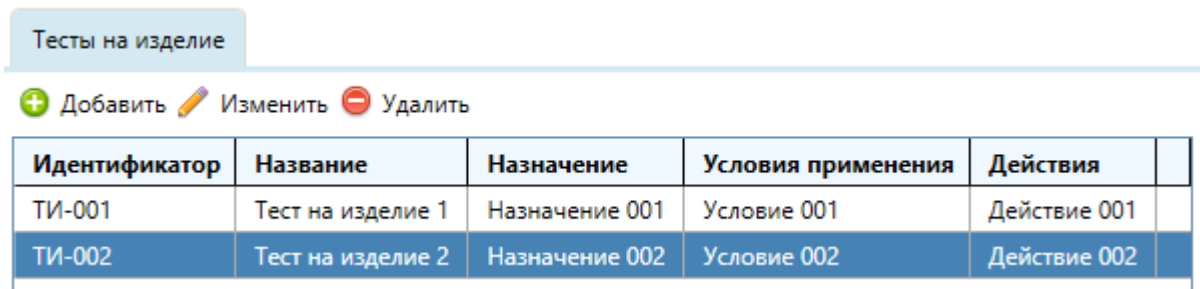
The screenshot shows a window titled 'dbo.TestSatellite [Данные]' and 'dbo.Test [Д...]' with a toolbar containing icons for refresh, filter, and sort. Below the toolbar is a table with two columns: 'Satelliteld' and 'Testld'. The table contains five rows of data, with the last row highlighted in blue.

Satelliteld	Testld
10007	1
10007	2
1	5
10009	6
NULL	NULL

Figure 64 – The entry in the TestSatellite table.

### 3.1.1.2 Updating the satellite test

For updating the satellite test, the user should select the test in the list (Figure 65) and click the "Edit" button.

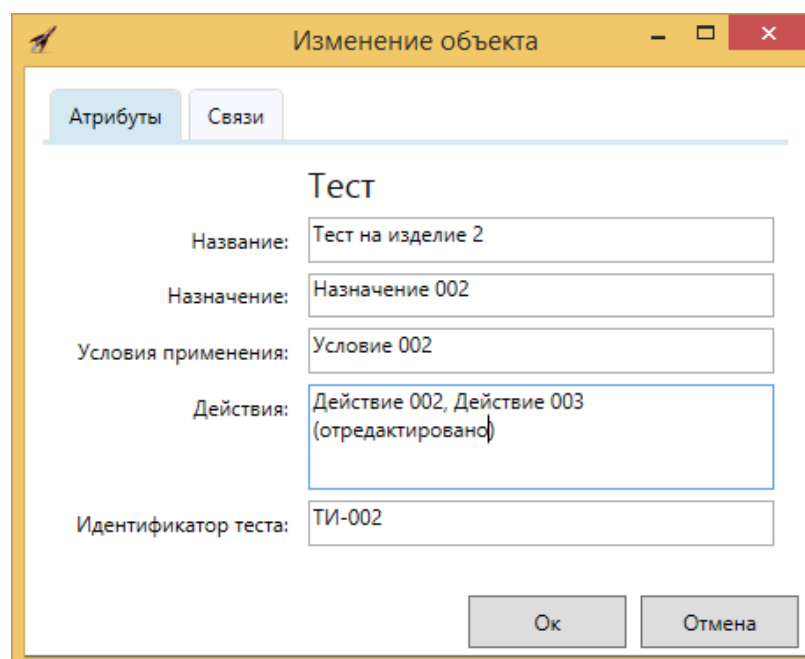


The screenshot shows a web application interface with a tab labeled 'Тесты на изделие'. Below the tab are three buttons: 'Добавить' (Add), 'Изменить' (Edit), and 'Удалить' (Delete). Below the buttons is a table with five columns: 'Идентификатор', 'Название', 'Назначение', 'Условия применения', and 'Действия'. The table contains two rows of data, with the second row highlighted in blue.

Идентификатор	Название	Назначение	Условия применения	Действия
ТИ-001	Тест на изделие 1	Назначение 001	Условие 001	Действие 001
ТИ-002	Тест на изделие 2	Назначение 002	Условие 002	Действие 002

Figure 65 – Selecting the test in the list.

In the displayed window (Figure 66), the user needs to edit the contents of the test fields and click the "Ok" button.



The screenshot shows a dialog box titled 'Изменение объекта' (Edit Object) with two tabs: 'Атрибуты' (Attributes) and 'Связи' (Relationships). The 'Атрибуты' tab is active, and the title of the dialog is 'Тест' (Test). The dialog contains several text input fields for editing test attributes:

- Название: Тест на изделие 2
- Назначение: Назначение 002
- Условия применения: Условие 002
- Действия: Действие 002, Действие 003 (отредактированс)
- Идентификатор теста: ТИ-002

At the bottom of the dialog are two buttons: 'Ок' (Ok) and 'Отмена' (Cancel).

Figure 66 – The window of satellite test editing.

After that, the test information should be updated in the list of tests (Figure 67).

Идентификатор	Название	Назначение	Условия применения	Действия
ТИ-001	Тест на изделие 1	Назначение 001	Условие 001	Действие 001
ТИ-002	Тест на изделие 2	Назначение 002	Условие 002	Действие 002, Действие 003 (отредактировано)

Figure 67 – The result of satellite test updating.

Besides, the test entry should be updated in the Test table (Figure 68).

TestId	Title	Identifier	Purpose	Condition	Actions	Parent...	Deleted	OnEditing	UserId	Created	Modified	CopiedFrom
1	Тест 1	T01	1	1	1	NULL	2	False	1	29.05.20...	29.05.201...	NULL
2	Тест 2-1	T02	2	2	2	NULL	0	False	NULL	29.05.20...	29.05.201...	NULL
3	Тест 1.1	T1-1	Применение	Условия	Действия	NULL	0	False	NULL	29.05.20...	29.05.201...	NULL
4	Тест 1	T11	Проверка	У1	Д1	NULL	0	False	1	29.05.20...	29.05.201...	NULL
5	Тест на издели...	ТИ1-1	Тестирование ...	У2	Д2	NULL	0	False	1	29.05.20...	29.05.201...	NULL
6	Тест на издели...	ТИ-001	Назначение 001	Условие 001	Действие 001	NULL	0	False	1	30.05.20...	30.05.201...	NULL
7	Тест на издели...	ТИ-002	Назначение 002	Условие 002	Действие 002, Действие 003 (отредактир...	NULL	0	False	NULL	30.05.20...	30.05.201...	NULL
8	Тест на издели...	ТИ-003	Назначение 003	Условие 003	Действие 003	NULL	0	False	1	30.05.20...	30.05.201...	NULL
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Figure 68 – The updated entry in the Test table.

### 3.1.1.3 Deleting the satellite test

To delete the satellite test, the user should select the test in the list (Figure 69) and click the "Delete" button.

Идентификатор	Название	Назначение	Условия применения	Действия
ТИ-001	Тест на изделие 1	Назначение 001	Условие 001	Действие 001
ТИ-002	Тест на изделие 2	Назначение 002	Условие 002	Действие 002, Действие 003 (отредактировано)
ТИ-003	Тест на изделие 3	Назначение 003	Условие 003	Действие 003

Figure 69 – Selecting the test in the list.

In the displayed window (Figure 70) the user should click the "Yes" button.

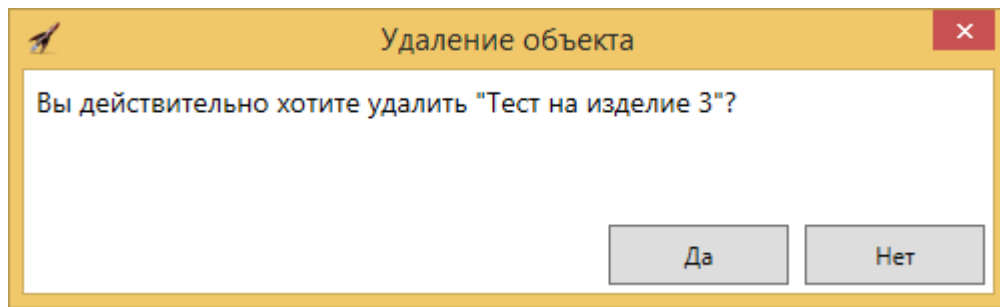


Figure 70 – Confirmation window of deleting the test.

After that, the information in the list of tests should be updated (Figure 71).

Идентификатор	Название	Назначение	Условия применения	Действия
ТИ-001	Тест на изделие 1	Назначение 001	Условие 001	Действие 001
ТИ-002	Тест на изделие 2	Назначение 002	Условие 002	Действие 002, Действие 003 (отредактировано)

Figure 71 – The result of deleting the satellite test.

The deleted test should remain in the Test table, with the Deleted field set to "2" (Figure 72).

TestId	Title	Identifier	Purpose	Condition	Actions	Parent...	Deleted	OnEditing	UserId	Created	Modified	CopiedFrom
1	Тест 1	T01	1	1	1	NULL	2	False	1	29.05.20...	29.05.201...	NULL
2	Тест 2-1	T02	2	2	2	NULL	0	False	NULL	29.05.20...	29.05.201...	NULL
3	Тест 1.1	T1-1	Применение	Условия	Действия	NULL	0	False	NULL	29.05.20...	29.05.201...	NULL
4	Тест 1	T11	Проверка	У1	Д1	NULL	0	False	1	29.05.20...	29.05.201...	NULL
5	Тест на издели...	ТИ1-1	Тестирование ...	У2	Д2	NULL	0	False	1	29.05.20...	29.05.201...	NULL
6	Тест на издели...	ТИ-001	Назначение 001	Условие 001	Действие 001	NULL	0	False	1	30.05.20...	30.05.201...	NULL
7	Тест на издели...	ТИ-002	Назначение 002	Условие 002	Действие 002, ...	NULL	0	False	NULL	30.05.20...	30.05.201...	NULL
8	Тест на издели...	ТИ-003	Назначение 003	Условие 003	Действие 003	NULL	2	False	1	30.05.20...	30.05.201...	NULL
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Figure 72 – The deleted entry in the Test table.

### 3.1.2 System test management

As a part of the system test management, the user is provided create, read, update and delete operations for system tests.

To navigate to the "System tests" tab, the user should open the folder of the system, belonging to the satellite, in the object tree and select the "Tests" folder inside the system folder (Figure 73).

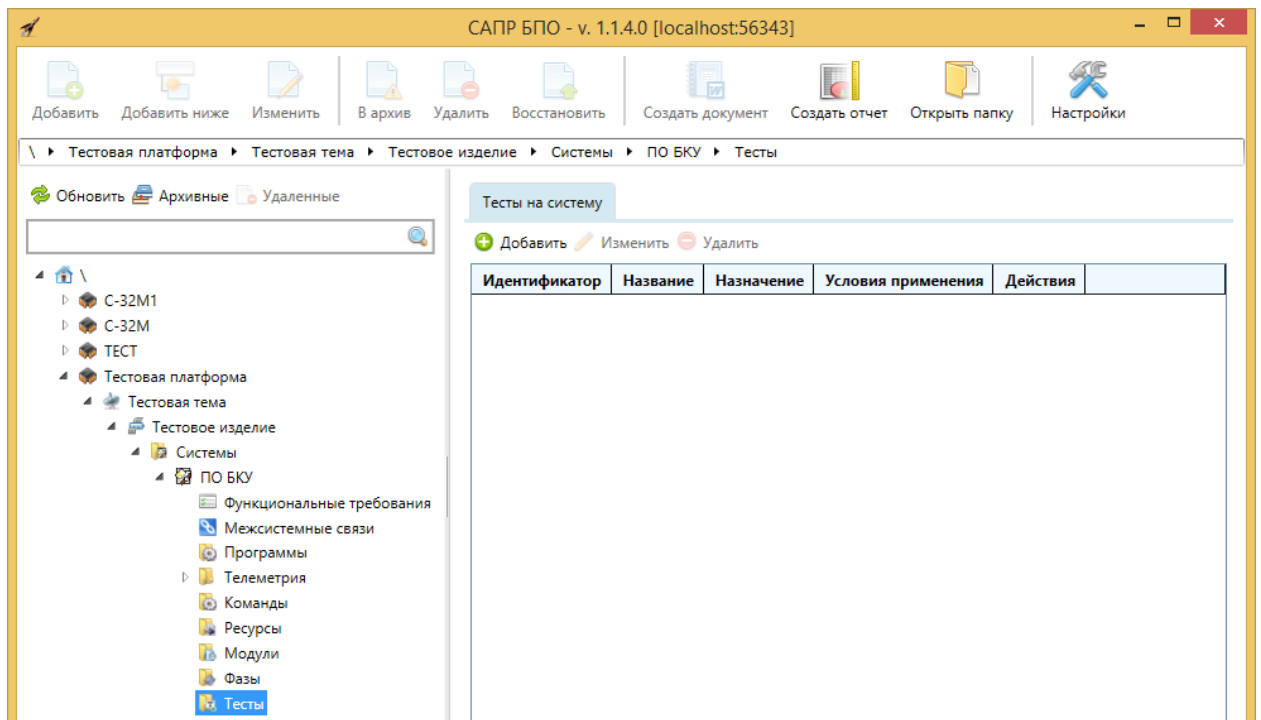


Figure 73 – The "System tests" tab.

### 3.1.2.1 Creating a system test

To create a system test, the user should click the "Add" button, fill the fields of the test in the displayed window (Figure 74) and click the "Ok" button.

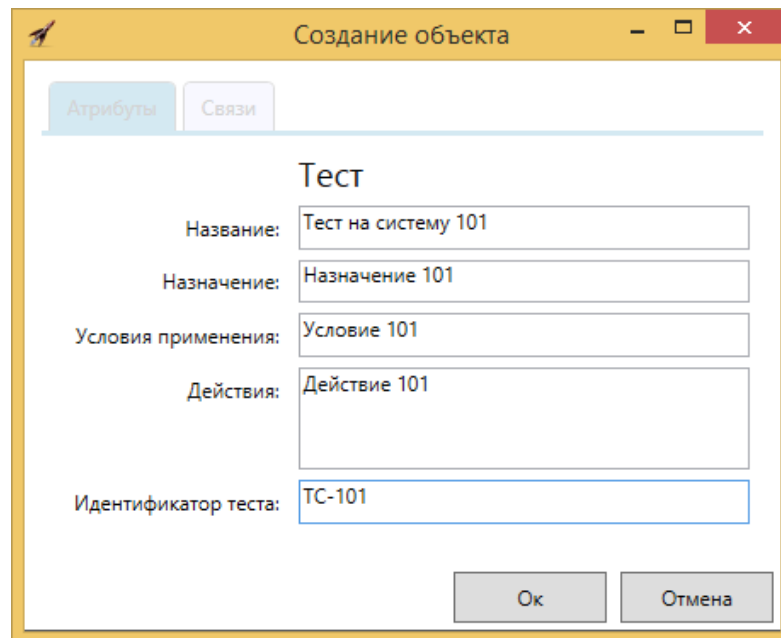


Figure 74 – Creating a system test.

The created test should appear in the list of tests for the selected system (Figure 75).

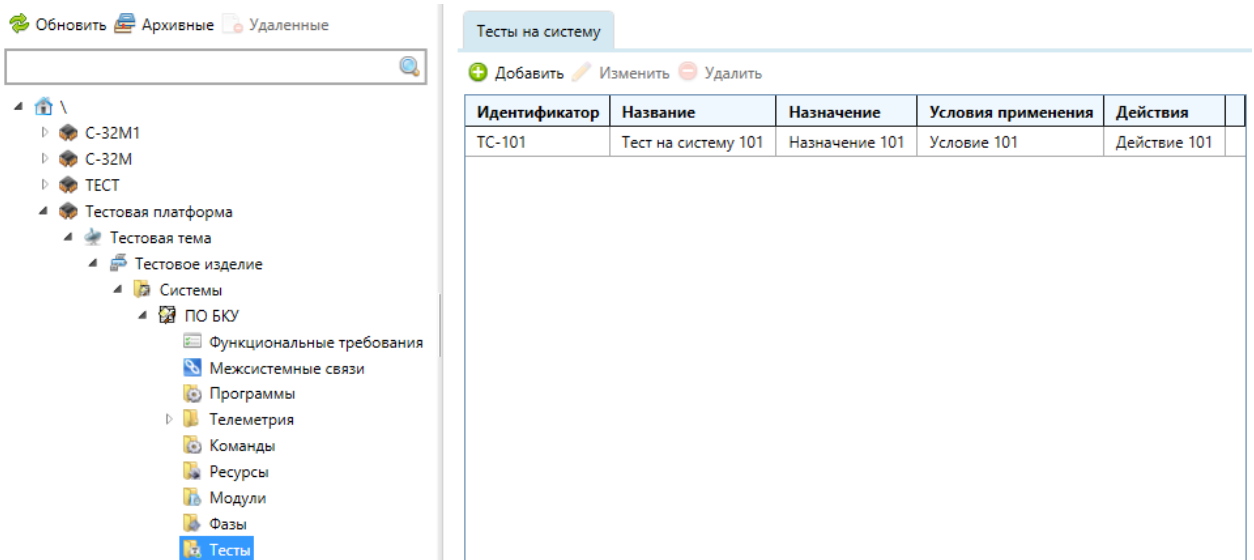


Figure 75 – The result of the system test creation.

Besides, the entry of the created test should appear in the Test table (Figure 76), the entry about the links between the test and the system should be displayed in the TestSatellite table (Figure 77).

TestId	Title	Identifier	Purpose	Condition	Actions	Parent...	Deleted	OnEditing	UserId	Created	Modified	CopiedFrom
1	Тест 1	T01	1	1	1	NULL	2	False	1	29.05.20...	29.05.201...	NULL
2	Тест 2-1	T02	2	2	2	NULL	0	False	NULL	29.05.20...	29.05.201...	NULL
3	Тест 1.1	T1-1	Применение	Условия	Действия	NULL	0	False	NULL	29.05.20...	29.05.201...	NULL
4	Тест 1	T11	Проверка	У1	Д1	NULL	0	False	1	29.05.20...	29.05.201...	NULL
5	Тест на издели...	ТИ1-1	Тестирование ...	У2	Д2	NULL	0	False	1	29.05.20...	29.05.201...	NULL
6	Тест на издели...	ТИ-001	Назначение 001	Условие 001	Действие 001	NULL	0	False	1	30.05.20...	30.05.201...	NULL
7	Тест на издели...	ТИ-002	Назначение 002	Условие 002	Действие 002, ...	NULL	0	False	NULL	30.05.20...	30.05.201...	NULL
8	Тест на издели...	ТИ-003	Назначение 003	Условие 003	Действие 003	NULL	2	False	1	30.05.20...	30.05.201...	NULL
9	Тест на систем...	ТС-101	Назначение 101	Условие 101	Действие 101	NULL	0	False	1	30.05.20...	30.05.201...	NULL
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Figure 76 – The entry in the Test table.

SystId	TestId
10030	3
1	4
10036	9
*	NULL

Figure 77 – The entry in the TestSystem table.

### 3.1.2.2 Updating the system test

For updating the system test, the user should select the test in the list (Figure 78) and click the "Edit" button.

Тесты на систему					
<span>+ Добавить</span> <span>✎ Изменить</span> <span>− Удалить</span>					
Идентификатор	Название	Назначение	Условия применения	Действия	
TC-101	Тест на систему 101	Назначение 101	Условие 101	Действие 101	
TC-102	Тест на систему 102	Назначение 102	Условие 102	Действие 102	

Figure 78 – Selecting the test in the list.

In the displayed window (Figure 79), the user needs to edit the contents of the test fields and click the "Ok" button.

Figure 79 – The window of system test editing.

After that, the test information should be updated in the list of tests (Figure 80).

Тесты на систему					
<span>+ Добавить</span> <span>✎ Изменить</span> <span>− Удалить</span>					
Идентификатор	Название	Назначение	Условия применения	Действия	
TC-101	Тест на систему 101	Назначение 101	Условие 101	Действие 101	
TC-102	Тест на систему 102	Назначение 102	Условие 102	Действие 102, Действие 103 (отредактировано)	

Figure 80 – The result of system test updating.

Besides, the test entry should be updated in the Test table (Figure 81).

TestId	Title	Identifier	Purpose	Condition	Actions	Parent...	Deleted	OnEditing	Userld	Created	Modified	CopiedFrom
1	Тест 1	T01	1	1	1	NULL	2	False	1	29.05.20...	29.05.201...	NULL
2	Тест 2-1	T02	2	2	2	NULL	0	False	NULL	29.05.20...	29.05.201...	NULL
3	Тест 1.1	T1-1	Применение	Условия	Действия	NULL	0	False	NULL	29.05.20...	29.05.201...	NULL
4	Тест 1	T11	Проверка	У1	Д1	NULL	0	False	1	29.05.20...	29.05.201...	NULL
5	Тест на издели...	ТИ1-1	Тестирование...	У2	Д2	NULL	0	False	1	29.05.20...	29.05.201...	NULL
6	Тест на издели...	ТИ-001	Назначение 001	Условие 001	Действие 001	NULL	0	False	1	30.05.20...	30.05.201...	NULL
7	Тест на издели...	ТИ-002	Назначение 002	Условие 002	Действие 002, Действие 003(отредакти...	NULL	0	False	NULL	30.05.20...	30.05.201...	NULL
8	Тест на издели...	ТИ-003	Назначение 003	Условие 003	Действие 003	NULL	2	False	1	30.05.20...	30.05.201...	NULL
9	Тест на систем...	ТС-101	Назначение 101	Условие 101	Действие 101	NULL	0	False	1	30.05.20...	30.05.201...	NULL
10	Тест на систем...	ТС-102	Назначение 102	Условие 102	Действие 102, Действие 103(отредакти...	NULL	0	False	NULL	30.05.20...	30.05.201...	NULL
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Figure 81 – The updated entry in the Test table.

### 3.1.2.3. Deleting the system test

To delete the system test, the user should select the test in the list (Figure 82) and click the "Delete" button.

Идентификатор	Название	Назначение	Условия применения	Действия
ТС-101	Тест на систему 101	Назначение 101	Условие 101	Действие 101
ТС-102	Тест на систему 102	Назначение 102	Условие 102	Действие 102, Действие 103 (отредактировано)
ТС-103	Тест на систему 103	Назначение 103	Условие 103	Действие 103

Figure 82 – Selecting the test in the list.

In the displayed window (Figure 83) the user should click the "Yes" button.

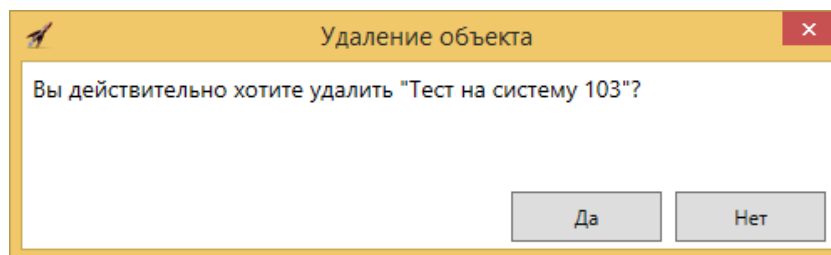


Figure 83 – Confirmation window of deleting the test.

After that, the information in the list of tests should be updated (Figure 84).

Идентификатор	Название	Назначение	Условия применения	Действия
ТС-101	Тест на систему 101	Назначение 101	Условие 101	Действие 101
ТС-102	Тест на систему 102	Назначение 102	Условие 102	Действие 102, Действие 103 (отредактировано)

Figure 84 – The result of deleting the system test.



The deleted test should remain in the Test table, with the Deleted field set to "2" (Figure 85).

Testid	Title	Identifier	Purpose	Condition	Actions	Parent...	Deleted	OnEditing	Userid	Created	Modified	CopiedFrom
1	Тест 1	T01	1	1	1	NULL	2	False	1	29.05.20...	29.05.201...	NULL
2	Тест 2-1	T02	2	2	2	NULL	0	False	NULL	29.05.20...	29.05.201...	NULL
3	Тест 1.1	T1-1	Применение	Условия	Действия	NULL	0	False	NULL	29.05.20...	29.05.201...	NULL
4	Тест 1	T11	Проверка	У1	Д1	NULL	0	False	1	29.05.20...	29.05.201...	NULL
5	Тест на издели...	ТИ1-1	Тестирование ...	У2	Д2	NULL	0	False	1	29.05.20...	29.05.201...	NULL
6	Тест на издели...	ТИ-001	Назначение 001	Условие 001	Действие 001	NULL	0	False	1	30.05.20...	30.05.201...	NULL
7	Тест на издели...	ТИ-002	Назначение 002	Условие 002	Действие 002, Действие 003 (отредакти...	NULL	0	False	NULL	30.05.20...	30.05.201...	NULL
8	Тест на издели...	ТИ-003	Назначение 003	Условие 003	Действие 003	NULL	2	False	1	30.05.20...	30.05.201...	NULL
9	Тест на систем...	ТС-101	Назначение 101	Условие 101	Действие 101	NULL	0	False	1	30.05.20...	30.05.201...	NULL
10	Тест на систем...	ТС-102	Назначение 102	Условие 102	Действие 102, Действие 103 (отредакти...	NULL	0	False	NULL	30.05.20...	30.05.201...	NULL
11	Тест на систем...	ТС-103	Назначение 103	Условие 103	Действие 103	NULL	2	False	1	30.05.20...	30.05.201...	NULL
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Figure 85 – The deleted entry in the Test table.

### 3.1.3 The traceability matrix of the requirements and the tests

The traceability matrix of the functional requirements and the tests is designed to show the relationship between the list of system functional requirements and the tests.

To navigate to the "Traceability matrix" tab, the user should open the folder of the system, belonging to the satellite, in the object tree, select the "Traceability matrix" item inside the system folder and click the "Traceability in tests" tab (Figure 86).

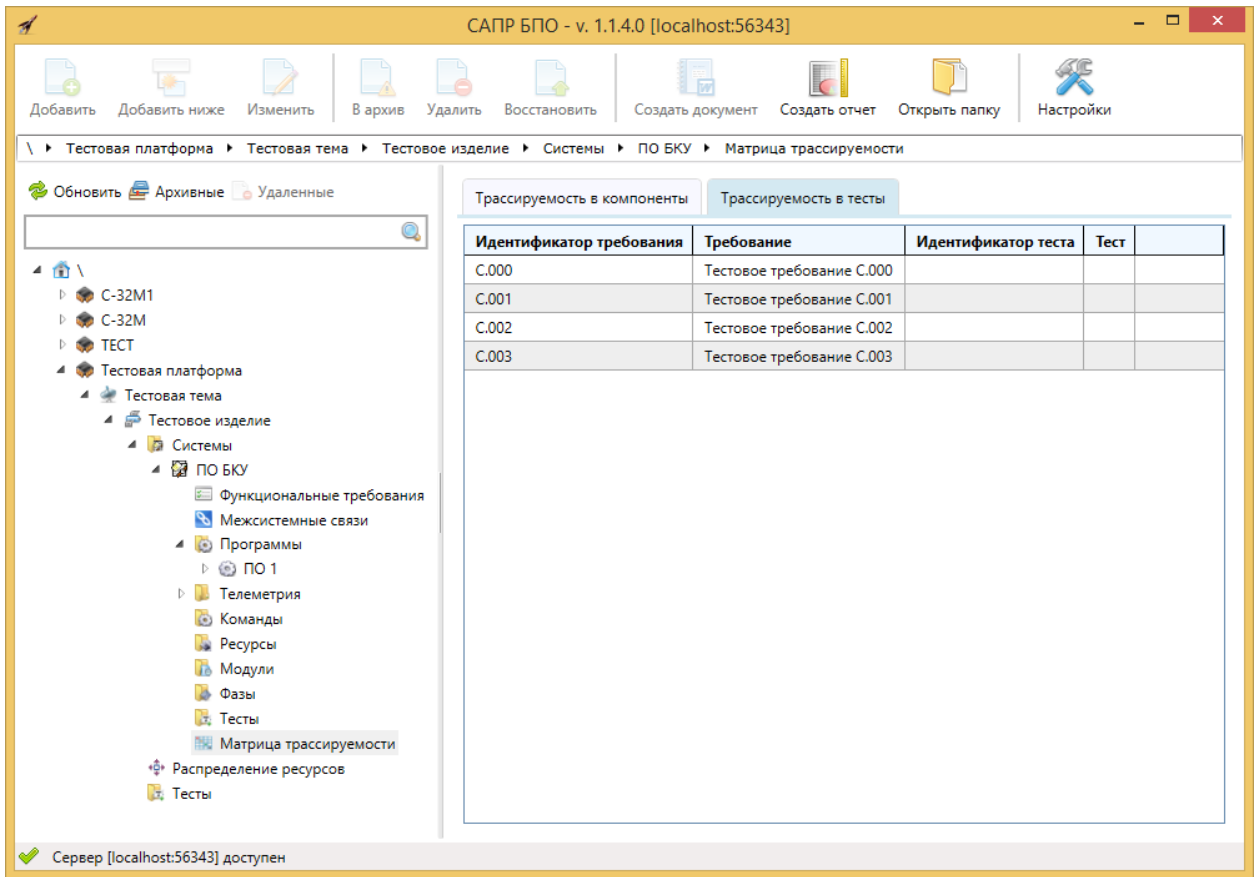


Figure 86 – The "Traceability in tests" tab.

### 3.1.3.1 Linking the test to the functional requirement

Linking the functional requirements to the tests is carried out using the RequirementTest table.

In order to set the link between the functional requirement and the test, it is required to add an entry containing the requirement and test identifiers to the RequirementTest table (Figure 87).

RequirementId	TestId
10473	3
1	4
1	5
10482	10
NULL	NULL

Figure 87 – The RequirementTest table.

### 3.1.3.2 Displaying the traceability matrix

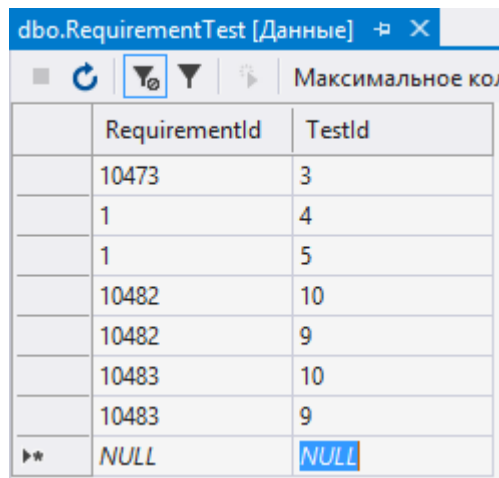
Displaying the traceability matrix of functional requirements and tests is performed using the RequirementTest table. Besides, deleted functional requirements and tests should not be displayed in the traceability matrix (Figure 88)

Трассируемость в компоненты		Трассируемость в тесты	
Идентификатор требования	Требование	Идентификатор теста	Тест
C.000	Тестовое требование C.000		
C.001	Тестовое требование C.001	TC-102	Тест на систему 102
C.002	Тестовое требование C.002		
C.003	Тестовое требование C.003		

Figure 88 – The traceability matrix.

### 3.1.3.3 Displaying the "many-to-many" link between functional requirements and test

To establish many-to-many relationships between functional requirements and tests, it is required to add entries to the RequirementTest table (Figure 89).



The screenshot shows a SQL Server Enterprise Manager window titled 'dbo.RequirementTest [Данные]'. The table has two columns: 'RequirementId' and 'TestId'. The data is as follows:

RequirementId	TestId
10473	3
1	4
1	5
10482	10
10482	9
10483	10
10483	9
NULL	NULL

Figure 89 – The RequirementTest table.

After that, in the traceability matrix of requirements and tests, the added links should appear, according to the rule that one requirement can be associated with several tests, and one test can be associated with several requirements (Figure 90).

Трассируемость в компоненты		Трассируемость в тесты		
Идентификатор требования	Требование	Идентификатор теста	Тест	
C.000	Тестовое требование C.000			
C.001	Тестовое требование C.001	TC-101	Тест на систему 101	
C.001	Тестовое требование C.001	TC-102	Тест на систему 102	
C.002	Тестовое требование C.002	TC-101	Тест на систему 101	
C.002	Тестовое требование C.002	TC-102	Тест на систему 102	

Figure 90 – The traceability matrix.