

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Локтионова Оксана Геннадьевна

Должность: проректор по учебной работе

Дата подписания: 15.06.2023 10:11:51

Уникальный программный ключ:

0b817ca911e6668abb13a5d426d39e5f1c11eabbf73e943df4a4851fda56d089

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное

образовательное учреждение высшего образования

«Юго-Западный государственный университет»

(ЮЗГУ)

Кафедра программной инженерии

УТВЕРЖДАЮ

Проректор по учебной работе

О.Г. Локтионова

« 25 » 10

2022 г.



## ВЫПОЛНЕНИЕ ВЫПУСКНОЙ КВАЛИФИКАЦИОННОЙ РАБОТЫ БАКАЛАВРА

Методические указания по выполнению выпускной квалификационной работы бакалавра для студентов направления подготовки бакалавров 09.03.03 «Прикладная информатика»



## Содержание

МИНОБРНАУКИ РОССИИ.....	1
1 Цель и задачи выпускной квалификационной работы.....	4
2 Выполнение выпускной квалификационной работы .....	5
2.1 Структура выпускной квалификационной работы.....	6
2.2 Требования к содержанию основной части выпускной квалификационной работы.....	7
3 Структура содержания основной части ВКРБ .....	11
4 Приложения .....	13
4.1 Приложение А. Графический материал.....	13
4.2 Приложение Б. Исходный текст программного продукта.....	15
5 Подготовка ВКРБ для проверки на объем заимствованного текста.....	17
6 Подготовка ВКРБ для передачи в библиотеку.....	18
7 Документы, вкладываемые в ВКРБ.....	19
8 Брошюрование ВКРБ .....	25
9 Подготовка электронного варианта ВКРБ.....	25
9.1 Содержание диска в приклеенном конверте .....	25

## **Термины и определения**

Выпускная квалификационная работа – выполненная обучающимся (несколькими обучающимися совместно) работа, демонстрирующая уровень подготовленности выпускника к самостоятельной профессиональной деятельности.

Дипломная работа – самостоятельно выполненное теоретическое или экспериментальное исследование на заданную тему, посвященное решению актуальной научной или научно-практической проблемы, определяемой спецификой направления подготовки и направленностью (профилем) образовательной программы.

Дипломный проект – самостоятельно выполненная работа, направленная на решение конкретной научно-технической, производственной или практической задачи с помощью создания нового объекта или реконструкции, модернизации имеющегося объекта (сооружения, механизма, машины, устройства, программы, технологии, метода, методики, услуги, изделия, продукта и т.п. или их составных частей), включающая расчетно-графическую часть (чертежи, макеты, схемы, стенды, фотографии, рисунки и т.п.).

### **1 Цель и задачи выпускной квалификационной работы**

Целью ВКР как формы ГИА является определение соответствия результатов освоения обучающимися основных образовательных программ соответствующим требованиям ФГОС ВО.

Основными задачами при выполнении ВКР являются:

- подтверждение способности применять знания, умения, навыки и компетенции, сформированные при освоении ОП ВО, для решения профессиональных задач;

- выявление способности к поиску новых решений (конструкторских, технологических, экономических, организационных, правовых и др.);

- выявление навыков самостоятельного анализа и синтеза при решении профессиональных задач с применением достижений в науке, технике, технологии, экономической и социальной сферах;

- проверка и оценка владения современными методиками научных исследований и эксперимента при решении профессиональных задач;

- определение готовности к выполнению видов профессиональной деятельности, установленных ФГОС ВО и определенных осваиваемой ОП ВО.

## **2 Выполнение выпускной квалификационной работы**

В соответствии с Федеральным государственным образовательным стандартом высшего образования направления подготовки (специальности) 09.03.04«Программная инженерия», утвержденного приказом Министерства образования и науки Российской Федерации № 229 от 12.03.2015, выполнение выпускной квалификационной работы проводится во время преддипломной практики, которая является обязательной.

ВКР выполняются на русском языке.

Допускается выполнение ВКР на иностранном языке, если обучающийся проходит обучение по ОП ВО на иностранном языке и (или) это установлено заданием на выполнение ВКР.

При выполнении ВКР выпускник должен подтвердить обладание следующими общепрофессиональными компетенциями:

владением основными концепциями, принципами, теориями и фактами, связанными с информатикой (ОПК-1);

владением архитектурой электронных вычислительных машин и систем (ОПК-2);

готовностью применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов (ОПК-3);

способностью осуществлять поиск, хранение, обработку и анализ информации из различных источников и баз данных, представлять ее в требуемом формате с использованием информационных, компьютерных и сетевых технологий (ОПК-4).

## **2.1 Структура выпускной квалификационной работы**

Выпускная квалификационная работа выполняется в виде пояснительной записки.

В соответствии со стандартом университета СТУ 04.02.030-2017 «Курсовые работы (проекты). Выпускные квалификационные работы. Общие требования к структуре и оформлению» структурными элементами пояснительной записки ВКР являются:

- титульный лист;
- задание;
- реферат;
- содержание;
- обозначения и сокращения (при необходимости);
- введение;
- основная часть;
- заключение;
- список использованных источников;
- приложения (при необходимости).

Оформление и содержание всех этих структурных элементов за исключением: основной части и приложений изложено в стандарте.

Выполнение, оформление и содержание основной части и приложений излагается в данных методических указаниях. Объем ВКР не менее 70 страниц (без приложений) на одной стороне листа А4.

## **2.2 Требования к содержанию основной части выпускной квалификационной работы**

В соответствии с Федеральным государственным образовательным стандартом высшего образования направления подготовки (специальности) 09.03.04«Программная инженерия», утвержденного приказом Министерства образования и науки Российской Федерации № 229 от 12.03.2015, в основной части ВКРБ должны быть отражены в соответствии с будущей:

### **- производственно-технологической деятельностью:**

- применение средств автоматизированного проектирования, разработки, тестирования и сопровождения программного обеспечения;

- применение методов и инструментальных средств управления инженерной деятельностью и процессами жизненного цикла программного обеспечения;

- использование типовых методов для контроля, оценки и обеспечения качества программной продукции;

- соответствие разрабатываемого программного обеспечения и технической документации российским и международным стандартам, техническим условиям, ведомственным нормативным документам и стандартам предприятия;

- взаимодействие с заказчиком в процессе выполнения программного проекта;

- процессы разработки программного обеспечения;

- создание технической документации по результатам выполнения работ;

### **- организационно-управленческой деятельностью:**

- составление технической документации (графиков работ, инструкций, планов, смет, заявок на материалы, оборудование, программное обеспечение) и установленной отчетности по утвержденным формам;

- планирование и организация собственной работы;

- планирование и координация работ по настройке и сопровождению программного продукта;

- планирование и организация работы малых коллективов исполнителей программного проекта;

- проведение технико-экономического обоснования программных проектов;

**-сервисно-эксплуатационной деятельностью:**

- ввод в эксплуатацию программного обеспечения (инсталляция, настройка параметров, адаптация, администрирование);

- профилактическое и корректирующее сопровождение программного продукта в процессе эксплуатации;

- составление частного технического задания на разработку программного продукта;

**- научно-исследовательской деятельностью:**

- проведение научных исследований (экспериментов, наблюдений и количественных измерений), связанных с объектами профессиональной деятельности (программными продуктами, проектами, процессами, методами и инструментами программной инженерии), в соответствии с утвержденными заданиями и методиками;

- построение моделей объектов профессиональной деятельности с использованием инструментальных средств компьютерного моделирования;

- составление описания проводимых исследований, подготовка данных для составления обзоров и отчетов;

**- аналитической деятельностью:**

- сбор и анализ требований заказчика к программному продукту;

- формализация предметной области программного проекта по результатам технического задания и экспресс-обследования;

- содействие заказчику в оценке и выборе вариантов программного обеспечения;

- составление коммерческого предложения заказчику, подготовка презентации и согласование пакета договорных документов;

**- проектной деятельностью:**

- проектирование компонентов программного продукта в объеме, достаточном для их конструирования в рамках поставленного задания;

- создание компонент программного обеспечения (кодирование, отладка, модульное и интеграционное тестирование);

- выполнение измерений и рефакторинг кода в соответствии с планом;

- интеграция компонент программного продукта;

- разработка тестового окружения, создание тестовых сценариев;

- разработка и оформление эскизной, технической и рабочей проектной документации;

**- педагогической деятельностью:**

- разработка методик обучения технического персонала и пособий по применению программных систем.

Исходя из профессиональных компетенций, которыми должен обладать выпускник, освоивший программу бакалавриата, установленных для преддипломной практики и государственной аттестации учебным планом направления подготовки (специальности) 09.03.04«Программная инженерия», одобренного Ученым советом университета протокол № 09 «25» мая 2015 г. в основной части ВКРБ должны быть отражены в соответствии с будущей:

**производственно-технологической деятельностью:**

основные методы и инструменты разработки программного обеспечения (ПК-1);

использование операционных систем, сетевых технологий, средств разработки программного интерфейса, применения языков и методов формальных спецификаций, систем управления базами данных (ПК-2);

использование различных технологий разработки программного обеспечения (ПК-3);

концепции и атрибуты качества программного обеспечения (надежности, безопасности, удобства использования), в том числе роли людей, процессов, методов, инструментов и технологий обеспечения качества (ПК-4);

стандарты и модели жизненного цикла (ПК-5);

**организационно-управленческой деятельностью:**

классические концепции и модели менеджмента в управлении проектами (ПК-6);

методы управления процессами разработки требований, оценки рисков, приобретения, проектирования, конструирования, тестирования, эволюции и сопровождения (ПК-7);

основы групповой динамики, психологии и профессионального поведения, специфичных для программной инженерии (ПК-8);

методы контроля проекта и контроля версий (ПК-9);

**сервисно-эксплуатационной деятельностью:**

основные концепции и модели эволюции и сопровождения программного обеспечения (ПК-10);

особенности эволюционной деятельности как с технической точки зрения, так и с точки зрения бизнеса (работы с унаследованными системами, возвратного проектирования, реинженеринга, миграции и рефакторинга) (ПК-11);

**научно-исследовательской деятельностью:**

формализация в своей предметной области с учетом ограничений используемых методов исследования (ПК-12);

использование методов и инструментальных средств исследования объектов профессиональной деятельности (ПК-13);

обоснование принимаемых проектных решений, осуществление постановки и выполнение экспериментов по проверке их корректности и эффективности (ПК-14);

оформление презентаций, научно-технических отчетов по результатам выполненной работы, опубликование результатов исследований в виде статей и докладов на научно-технических конференциях (ПК-15);

**аналитической деятельностью:**

формализация предметной области программного проекта и разработка спецификаций для компонентов программного продукта (ПК-16);

начальная оценка степени трудности, рисков, затрат и формирование рабочего графика (ПК-17);

коммерческие предложения с вариантами решения (ПК-18);

**проектной деятельностью:**

моделирование, анализ и использование формальных методов конструирования программного обеспечения (ПК-19);

оценки временной и емкостной сложности программного обеспечения (ПК-20);

чтение, понимание и выделение главной идеи прочитанного исходного кода, документации (ПК-21);

создание программных интерфейсов (ПК-22);

оформление методических материалов и пособий по применению программных систем (ПК-24).

### **3 Структура содержания основной части ВКРБ**

Структура ВКРБ следующая:

1 Техническое задание

1.1 Основание для разработки

1.2 Назначение разработки

1.3 Требования к программно-информационной системе

1.3.1 Требования к данным программно-информационной системы

1.3.2 Функциональные требования к программно-информационной системе

1.3.2.1 Вариант использования 1 ...

1.3.2.2 Вариант использования 2 ...

1.3.2.3 Вариант использования 3 ...

1.3.3 Требования к пользовательскому интерфейсу программно-информационной системы

- 1.3.4 Нефункциональные требования к программно-информационной системе
- 1.4 Требования к оформлению документации
- 2 Технический проект
  - 2.1 Общие сведения о программно-информационной системе
  - 2.2 Проект данных программно-информационной системы
    - 2.2.1 Структура базы данных
    - 2.2.2 Описание структуры данных для ...
    - 2.2.3 Описание формата данных ...
  - 2.3 Проектирование архитектуры программно-информационной системы
    - 2.3.1 Компоненты программно-информационной системы
    - 2.3.2 Архитектура программно-информационной системы
  - 2.4 Проектирование пользовательского интерфейса программно-информационной системы
- 3 Рабочий проект
  - 3.1 Спецификация компонентов программно-информационной системы
    - 3.1.1 Спецификация класса 1...
    - 3.1.2 Спецификация класса 2 ...
    - 3.1.3 Спецификация класса 3 ...
  - 3.2 Тестирование программной системы
    - 3.2.1 Модульное тестирование компонентов программно-информационной системы
      - 3.2.1.1 Тестирование методом белого ящика компонента ...
      - 3.2.1.2 Тестирование методом черного ящика компонента ...
      - 3.2.1.3 Модульные тесты для компонента 1 ...
      - 3.2.1.4 Модульные тесты для компонента 2
    - 3.2.2 Интеграционное тестирование компонентов программной системы
    - 3.2.3 Системное тестирование компонентов программно-информационной системы
  - 3.3 Сборка компонентов программно-информационной системы

## 4 Приложения

### 4.1 Приложение А. Графический материал

Все листы графического материала выполняются в формате А1 и должны оформляться как чертежи в соответствии с требованиями ЕСКД. На одном листе могут быть выполнены несколько чертежей, с другой стороны один чертеж может быть выполнен на нескольких листах А1. На всех листах должны быть нанесены рамки чертежа и основная надпись («штамп») рисунки 2-4.

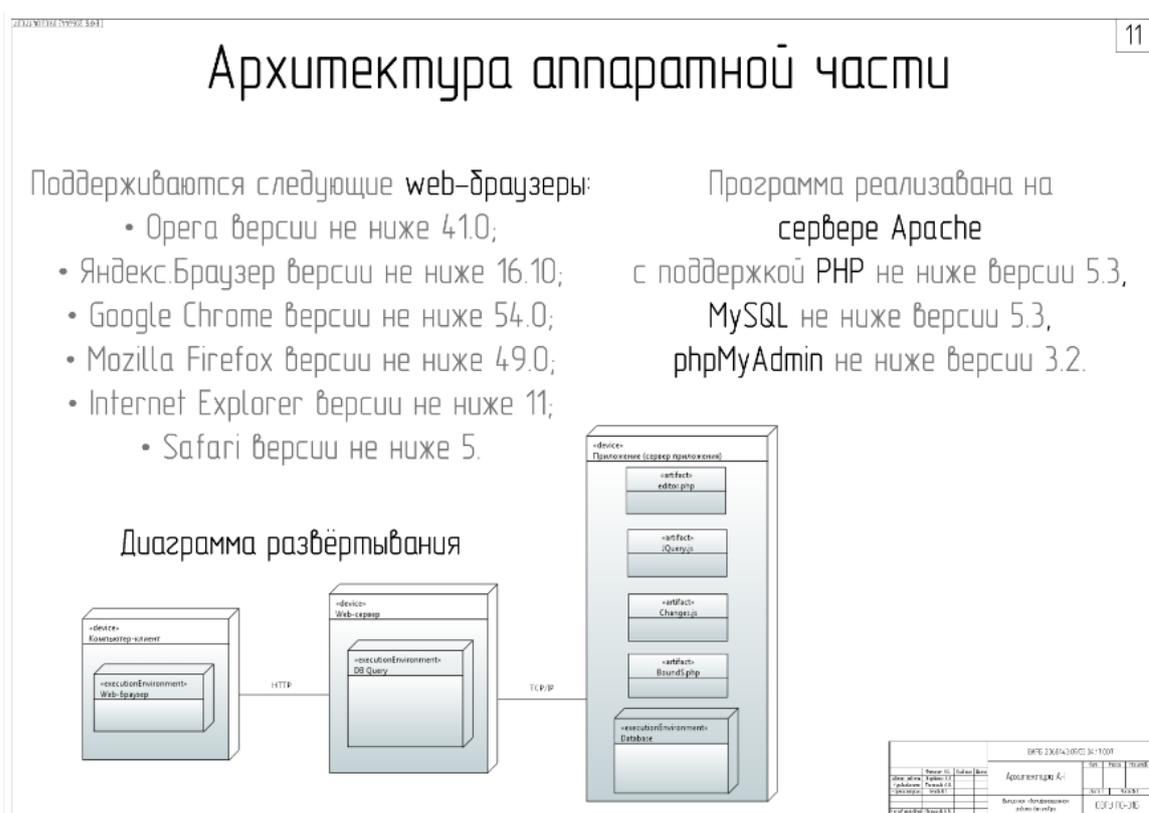


Рисунок 2

В правом верхнем углу внутри квадрата ставится номер листа в соответствии с его номером в содержании ВКРБ, используя шрифт хорошо видимый при презентации графического материала.

Структура шифра ВКРБ (рисунок 3) в основной надписи ВКРБ 2068443.09.03.04.17.007 следующая:

- ВКРБ – выпускная квалификационная работа бакалавра;

- 2068443 – код вуза;
- 09.03.04 – код специальности;
- 17 – год выпуска;
- 007 – трехзначный номер студента в приказе об утверждении тем дипломных работ (первая цифра 0 соответствует основному приказу, цифра 1 соответствует первому дополнению к основному приказу, цифра 2 соответствует второму дополнению к приказу и т.д.).

				ВКРБ 2068443.09.03.04.17.007			
	Фамилия И.О.	Подпись	Дата	Архитектура АЧ	Лист	Масса	Масштаб
Автор работы	Жерденко К.А.						
Руководитель	Малишев А.В.						
Нормоконтроль	Беляев В.Г.				Лист 1	Листов 1	
				Выпускная квалификационная работа бакалавра	ЮЗГУ ПО-31Б		

Рисунок 3

Шифр ВКРБ должен быть также отображен в рамке вдоль длинной стороны чертежа так, как представлено на рисунке 4, за исключением формата А4.



Рисунок 4

Так как графический материал выполняется в электронном виде, то три подписи: автора работы, руководителя и нормоконтролера ставятся внизу на свободном месте листа А4 внешнего вида графического материала, приведенного в приложении А.

## **4.2 Приложение Б. Исходный текст программного продукта**

Оформление текстов исходных модулей возможно так, как представлено на рисунке 5.

Приложение Б  
Исходные тексты программной среды  
¶

На рисунках Б.1 – Б.3 представлен исходный текст файла «editor.php».

```
<!DOCTYPE html>¶
<html>¶
→ <head>¶
→ → <title>Редактор баннеров</title>¶
→ → <meta http-equiv="Content-Type" content="text/html; charset=utf-8">¶
→ → <meta name="viewport" content="width=device-width, initial-scale=1.0">¶
→ <script src="js/jquery-3.1.1.min.js"></script>¶
→ → <link rel="stylesheet" href="css/DIPEditingStyle.css">¶
→ <script src="js/fileLoader.js"></script>¶
→ <php-include('downloader.php');?>¶
→ </head>¶
→ <body class="DIPEEDITING">¶
→ → <header class="DIPEEDITINGheader">Редактор баннеров</header>¶
→ → <div id="DIPEEDITINGcontent">+ → ¶
→ → → <aside id="DIPEEDITINGaside_left" class="DIPEEDITINGaside_left
blueToBlack">¶
→ → → → <input value="Изменить картинку" type="button" name="
"DIPEEDITINGchangePic" class="DIPEEDITINGcontrolButton">¶
→ → → → <input value="Изменить надпись" type="button" name="
"DIPEEDITINGchangeText" class="DIPEEDITINGcontrolButton">¶
→ → → → <input value="Изменить ссылку" type="button" name="
"DIPEEDITINGchangeRef" class="DIPEEDITINGcontrolButton">¶
→ → → → <input value="Изменить размер" type="button" name="
"DIPEEDITINGchangeSize" class="DIPEEDITINGcontrolButton">¶
```

Рисунок Б.1 – Исходный текст файла «editor.php»

## 5 Подготовка ВКРБ для проверки на объем заимствованного текста

ВКРБ проверяется на объем заимствованного текста с помощью системы "Антиплагиат".

Заимствование—использование правомерно опубликованного произведения без согласия автора и без выплаты вознаграждения, но с обязательным указанием имени автора и источника заимствования в определенных строго оговоренных законом случаях.

Неправомерное заимствование – отсутствие указания на автора и источник заимствования или превышение допустимого объема заимствований, не оправданного целью работы.

Техническое (правомерное) заимствование – использование в тексте наименований учреждений, органов государственной власти и местного самоуправления; ссылок на нормативные правовые акты; текстов законов; списков литературы; повторов, в том числе часто повторяющихся устойчивых выражений и юридических терминов.

Правомерно заимствованными могут быть следующие материалы:

– официальные документы государственных органов и органов местного самоуправления муниципальных образований, в том числе законов, других нормативных актов, судебные решения, иные материалы законодательного, административного и судебного характера, официальные документы международных организаций, а также их официальные переводы;

– государственные символы и знаки (флаги, гербы, ордена, денежные знаки тому подобное), а также символы и знаки муниципальных образований;

– произведения народного творчества (фольклор), не имеющие конкретных авторов;

– сообщения о событиях и фактах, имеющие исключительно информационный характер (сообщения о новостях дня, программы телепередач, расписания движения транспортных средств, и тому подобное);

– устойчивые выражения;

– ранее опубликованные материалы автора работы (самоцитирование).

Не считаются воспроизведением/цитированием включенные в текст ВКР: исходные формулы, шапки типовых таблиц, графиков и диаграмм, библиографические описания источников (кроме списков использованных источников, воспроизведенных большими фрагментами или целиком), фрагменты нормативных правовых актов и локальных актов организаций, предприятий, включенные в текст ВКР в качестве иллюстраций и примеров (при соблюдении правил цитирования).

Использование заимствованного текста без ссылки на автора и (или) источник заимствования в ВКР не допускается.

Для проверки на антиплагиат используется вариант пояснительной записки содержащий структурные разделы, начиная с введения и заканчивая заключением (до списка использованных источников). Вариант предоставляется по электронной почте в формате doc ответственным от кафедры.

Если ВКР содержит оригинального текста менее 50% от общего объема, то она возвращается обучающемуся на доработку и должна пройти повторную проверку не позднее 3 календарных дней до даты защиты.

## **6 Подготовка ВКРБ для передачи в библиотеку**

Для размещения ВКРБ в библиотеке используется вариант пояснительной записки. Все листы ВКРБ, на которых имеются подписи, включая графический материал, должны быть вставлены на свои места в виде сканов с подписями. Такой вариант пояснительной записки должен быть представлен в формате pdf и не должен превышать 10Mb.

В дополнение к файлам ВКРБ формируется файл в формате doc со сведениями, представленными в таблице 1.

Таблица 1- Сведения о ВКРБ

<b>Название поля</b>	<b>Содержание</b>
<b>Номер студенческого билета</b>	01986
<b>ФИО автора работы</b>	Жерденко Карина Артуровна
<b>ФИО руководителя</b>	Малышев Александр Васильевич
<b>ФИО рецензента</b>	
<b>Название работы</b>	Программная среда форматонезависимого редактирования и размещения интернет-баннеров
<b>Ключевые слова</b>	интернет-баннер, форматонезависимое редактирование, фон, картинка, надпись, ссылка, размер, размещение.
<b>Год написания работы</b>	2017
<b>Объем</b>	160
<b>Степень оригинальности</b>	100
<b>Специальность</b>	09.03.04
<b>Выпускающая кафедра</b>	Программная инженерия
<b>Уровень подготовки</b>	Бакалавриат

Данные три файла пересылаются ответственному представителю от кафедры.

## **7 Документы, вкладываемые в ВКРБ**

В карман папки с ВКРБ вкладываются следующие документы:

- Отзыв руководителя (рисунок 6).
- Отчет о проверке текста ВКР на оригинальность (рисунок 7).
- Обоснование об исключении сведений из ВКРБ (если необходимо) (рисунок 8).
- Справка о практическом использовании материалов ВКРБ (рисунок 19).

Минобрнауки России  
Юго-Западный государственный университет

Кафедра \_\_\_\_\_ Программной инженерии \_\_\_\_\_

**ОТЗЫВ**

руководителя о выпускной квалификационной работе

Дипломный проект

(указать нужно: дипломная работа, дипломный проект)

Студента Жерденко Марины Артуровны

(фамилия, имя, отчество)

Группы ПО-316 направления подготовки (специальности) «Программная инженерия»

На тему: Программная среда форматонезависимого редактирования и размещения интернет-баннеров

1. Объем работы: количество страниц 160. Графическая часть 21 лист.

2. Цель и задачи исследования:

Разработать программную среду, позволяющую редактировать интернет-баннеры, созданные на разном программном обеспечении, и размещать их в сети Интернет.

3. Актуальность, теоретическая, практическая значимость темы исследования:

Результаты данной работы можно использовать в различных организациях, пользующихся интернет-рекламой, для редактирования интернет-баннеров.

4. Соответствие содержания работы заданию (полное или неполное): полное

5. Основные достоинства и недостатки работы:

Достоинством данной работы является возможность редактировать интернет-баннеры, созданные на разном программном обеспечении. Существенных недостатков нет.

6. Степень самостоятельности и способности к исследовательской работе (умение и навыки искать, обобщать, анализировать материал и делать выводы):

Работа выполнена студентом самостоятельно с проявлением способностей правильно анализировать материал и программно реализовывать необходимые алгоритмы.

7. Оценка деятельности студента в период выполнения работы (степень добросовестности, работоспособности, ответственности, аккуратности и т.п.):

Данная работа была выполнена добросовестно и ответственно.

8. Достоинства и недостатки оформления текстовой части, графического, демонстрационного, иллюстративного, компьютерного и информационного материала.

Соответствие оформления требованиям стандартов:

Оформление данной выпускной квалификационной работы, графического и иллюстративного материалов соответствует требованиям стандарта университета.

9. Целесообразность и возможность внедрения результатов исследования:

Результаты ВКР находятся на стадии предложения внедрения.

10. По результатам проверки ВКР на объем заимствования в системе «Антиплагиат» оригинальный текст составляет 100%; заимствования составляют 0 %, из которых правомочные заимствования - 0 %.

Итоговая оценка оригинальности, с учетом правомочных заимствований, составляет 100%.

Распечатка отчета о проверке прилагается.

11. При выполнении ВКР автор показал высокий уровень сформированности компетенций, установленных ФГОС ВО.

12. Общее заключение и предлагаемая оценка отлично

Руководитель Малышев Александр Васильевич, и.о.зав.кафедрой ПИ, к.т.н., доцент

(фамилия, имя, отчество, должность, ученая степень, ученое звание)

Дата «29» мая 20 17г. Подпись \_\_\_\_\_

Рисунок 6

## Отчет о проверке на заимствования №1



**Автор:** Чаплыгин Александр Александрович [alex\\_chaplygin@mail.ru](mailto:alex_chaplygin@mail.ru) / ID: 180  
**Проверяющий:** Чаплыгин Александр Александрович ([alex\\_chaplygin@mail.ru](mailto:alex_chaplygin@mail.ru) / ID: 180)  
**Организация:** Юго-Западный государственный университет  
 Отчет предоставлен сервисом «Антиплагиат» - <http://swsu.antiplagiat.ru>

### ИНФОРМАЦИЯ О ДОКУМЕНТЕ

№ документа: 58  
 Начало загрузки: 17.12.2021 09:24:58  
 Длительность загрузки: 00:00:11  
 Имя исходного файла: Андросов А.С..docx  
 Название документа: Андросов А.С.  
 Размер текста: 1 кБ  
 Символов в тексте: 53 570  
 Слов в тексте: 6441  
 Число предложений: 321

### ИНФОРМАЦИЯ ОБ ОТЧЕТЕ

Последний готовый отчет (ред.)  
 Начало проверки: 17.12.2021 09:25:10  
 Длительность проверки: 00:01:54  
 Комментарии: не указано  
 Поиск с учетом редактирования: да  
 Модули поиска: ИПС Адилет, Библиография, Сводная коллекция ЭБС, Интернет Плюс, Сводная коллекция РГБ, Цитирование, Переводные заимствования (RuEn), Переводные заимствования по eLIBRARY.RU (EnRu), Переводные заимствования по Интернету (EnRu), Переводные заимствования издательства Wiley (RuEn), eLIBRARY.RU, СПС ГАРАНТ, Медицина, Диссертации НББ, Перефразирования по eLIBRARY.RU, Перефразирования по Интернету, Патенты СССР, РФ, СНГ, СМ И России и СНГ, Шаблонные фразы, Модуль поиска "ЮЗГУ", Кольцо вузов, Издательство Wiley, Переводные заимствования



Заимствования — доля всех найденных текстовых пересечений, за исключением тех, которые система отнесла к цитированиям, по отношению к общему объему документа.  
 Самоцитирования — доля фрагментов текста проверяемого документа, совпадающий или почти совпадающий с фрагментом текста источника, автором или со автором которого является автор проверяемого документа, по отношению к общему объему документа.  
 Цитирования — доля текстовых пересечений, которые не являются авторскими, но система посчитала их использование корректным, по отношению к общему объему документа. Сюда относятся оформленные по ГОСТу цитаты; общеупотребительные выражения; фрагменты текста, найденные в источниках из коллекций нормативно-правовой документации.  
 Текстовое пересечение — фрагмент текста проверяемого документа, совпадающий или почти совпадающий с фрагментом текста источника.  
 Источник — документ, проиндексированный в системе и содержащийся в модуле поиска, по которому проводится проверка.  
 Оригинальность — доля фрагментов текста проверяемого документа, не обнаруженных ни в одном источнике, по которому шла проверка, по отношению к общему объему документа.  
 Заимствования, самоцитирования, цитирования и оригинальность являются отдельными показателями и в сумме дают 100%, что соответствует всему тексту проверяемого документа.  
 Обращаем Ваше внимание, что система находит текстовые пересечения проверяемого документа с проиндексированными в системе текстовыми источниками. При этом система является вспомогательным инструментом, определение корректности и правомерности заимствований или цитирований, а также авторства текстовых фрагментов проверяемого документа остается в компетенции проверяющего.

№	Доля в отчете	Источник	Актуален на	Модуль поиска	Комментарии
[01]	3,47%	tashlykova_a_s_razrabotka-prilojeniya-dlya-sistemizatsii-rezultatov-testirovaniya-pr-ogrammnyh-produktiv.docx	24 Мар 2020	Кольцо вузов	
[02]	0%	NikolaevPV_ivcht_2012_text.doc	25 Фев 2013	Кольцо вузов	
[03]	0%	- «1С. Предприятие 8. Управление строительной организацией для Украины»; (1/2) <a href="http://studfiles.ru">http://studfiles.ru</a>	раньше 2011	Интернет Плюс	
[04]	0%	Условие (2/2) <a href="http://reshaem.n.et">http://reshaem.n.et</a>	12 Фев 2018	Интернет Плюс	
[05]	0%	Внедрение системы электронного документооборота на базе Docsvision 5 в «Московскую областную энергосетевую компанию»	16 Июнь 2014	Кольцо вузов	
[06]	0,15%	2.2 Диаграммы uml организационной структуры проектируемой информационной системы <a href="http://studfiles.ru">http://studfiles.ru</a>	03 Авг 2016	Интернет Плюс	
[07]	0%	Диаграммы UML организационной структуры проектируемой информационной системы <a href="http://lektii.org">http://lektii.org</a>	06 Ноя 2017	Интернет Плюс	
[08]	0,33%	Автоматизированная информационная система управления «Отдел кадров» (АИСОК) Техническое задание Действует с _____ скачать документ doc, docx <a href="http://folio.ru">http://folio.ru</a>	14 Янв 2017	Интернет Плюс	
[09]	0%	Богун Дарья Юрьевна ИУ/ИТ id_e-618945 id_vf_work-286588.docx	19 Июнь 2017	Кольцо вузов	
[10]	0%	ВКР	24 Июнь 2021	Кольцо вузов	
[11]	0%	Автоматизация информационной системы <a href="https://revolution.allbest.ru">https://revolution.allbest.ru</a>	25 Окт 2020	Интернет Плюс	
[12]	0%	Автоматизация информационной системы <a href="https://revolution.allbest.ru">https://revolution.allbest.ru</a>	04 Июнь 2020	Интернет Плюс	

Характеристика объектов автоматизации, Требования к системе,

Рисунок 7

Федеральное государственное бюджетное образовательное учреждение  
высшего образования

Юго-Западный государственный университет

Кафедра программной инженерии

Обоснование

исключения производственных, технических, экономических, организационных  
и других сведений из электронной версии ВКР

на тему: «Автоматизированная система бухгалтерского учета для МКУ

«Централизованная бухгалтерия учреждений образования»

Автор: Евстифорова Алина Викторовна

Сведения были изъяты из ВКР в связи с тем, что они содержат результаты  
интеллектуальной деятельности в научно-технической сфере, имеющие  
потенциальную коммерческую ценность в силу неизвестности их третьим лицам.

доцент  
Должность руководителя ВКР

29.05.17  
Дата, подпись руководителя ВКР

А.В.Малышев  
И.О.Фамилия

Одобрено на заседании кафедры программной инженерии № «12» 26.05.2017г.  
(наименование кафедры, дата, номер протокола)

Заведующий кафедрой

29.05.17  
Дата, подпись

А.В. Малышев  
И.О.Фамилия

Рисунок 8

## ООО «ФИНКОМ»

ИНН- 4632159113 КПП- 463201001 ОГРН- 1114632011350

Р/С – 40702810633000008341 в Курском ОСБ №8596

К/С – 30101810300000000606 БИК – 043807606

305014 г Курск, проспект Победы, дом №26 т. 39-76-76

### СПРАВКА

о результатах внедрения решений,  
разработанных в выпускной квалификационной работе  
студентки группы ПО-316 Жерденко Марины Артуровны

В процессе работы над выпускной квалификационной работой по теме «Программная среда форматонезависимого редактирования и размещения интернет-баннеров» студентка Жерденко К.А. приняла непосредственное участие в разработке программной среды форматонезависимого редактирования и размещения интернет-баннеров, которая позволяет вносить изменения в интернет-баннеры, созданные на любом программном обеспечении, сохранять их и размещать на сторонних веб-сайтах.

Результаты данной выпускной квалификационной работы предполагается использовать в организации ООО «ФинКом».

Директор ООО «ФинКом»

  
К.О. Веретенникова  
М.П.



Рисунок 9

## **8 Брошюрование ВКРБ**

Листы пояснительной записки перфорируются скоросшивателем и сшиваются толстой ниткой в аккуратную стопку. На обороте последнего листа нитка завязывается узлом. Сам узел и его кончики приклеиваются (наложением на них сверху) к последнему листу небольшим по формату прямоугольным листом бумаги, с датируемыми фио и подписями руководителя и студента с обязательным указанием количества сшитых листов. Полученная стопка вставляется в папку со скоросшивателем, усики которого представляют длинные пружины. Планка скоросшивателя железная. Папка должна иметь твердую обложку с карманом, в который вставляются вкладываемые документы.

## **9 Подготовка электронного варианта ВКРБ**

### **9.1 Содержание диска в приклеенном конверте**

Диск содержит в электронном виде все материалы, создаваемые в процессе выполнения ВКРБ, и имеет следующую структуру:

**Папка программный продукт:**

**- папка программный продукт в исходных кодах:**

**- папка компоненты:**

**- папка библиотеки (архивные, разделяемые),**

**- папка файлы данных (документов),**

**- папка программные файлы,**

**- папка база данных,**

**- папка программный продукт в исполнимых кодах:**

**- папка библиотеки (разделяемые),**

- папка файлы данных (документов),
- папка программные файлы,
- папка база данных.

#### **Папка выпускная квалификационная работа:**

##### **-папка пояснительная записка:**

- полная пояснительная записка в формате doc,

##### **- папка графический материал:**

- файл листа 1 графического материала на листе формата А1 в формате использованного графического редактора,
- файл листа 2 графического материала на листе формата А1 в формате использованного графического редактора,
- ...
- файл листа N графического материала на листе формата А1 в формате использованного графического редактора.

#### **Папка документы ВКРБ:**

- сканы документов с подписями:
  - отзыв руководителя,
  - отчет о проверке текста ВКР на оригинальность,
  - обоснование об исключении сведений из ВКРБ,
  - справка о практическом использовании материалов ВКРБ.

#### **Папка презентация ВКРБ:**

- файлы или файл презентации, соответствующие всем листам графического материала.

#### **Папка раздаточный материал ВКРБ :**

- файл с внешним видом графического материала на листах формата А4 в формате doc.

**Приложение**  
**Пример выполнения ВКРБ**  
**Минобрнауки России**  
**Юго-Западный государственный университет**

Кафедра \_\_\_\_\_ Программной инженерии \_\_\_\_\_

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА ПО**  
**ПРОГРАММЕ БАКАЛАВРИАТА**

09.03.04 «Программная инженерия»

(код, наименование ОПОП ВО: направление подготовки, направленность (профиль))

«Разработка программно-информационных систем»

Система управления IT-услугами на основе концепции ITSM

(название темы)

Дипломный проект

(вид ВКР: дипломная работа или дипломный проект)

Автор ВКР

\_\_\_\_\_ (подпись, дата)

С. Е. Копылов

\_\_\_\_\_ (инициалы, фамилия)

Группа ПО-71з

Руководитель ВКР

\_\_\_\_\_ (подпись, дата)

В. В. Апальков

\_\_\_\_\_ (инициалы, фамилия)

Нормоконтроль

\_\_\_\_\_ (подпись, дата)

А.А. Чаплыгин

\_\_\_\_\_ (инициалы, фамилия)

ВКР допущена к защите:

Заведующий кафедрой

\_\_\_\_\_ (подпись, дата)

А.В. Малышев

\_\_\_\_\_ (инициалы, фамилия)

Курск 2021 г.

# Юго-Западный государственный университет

Кафедра \_\_\_\_\_ Программной инженерии \_\_\_\_\_

УТВЕРЖДАЮ:  
Заведующий кафедрой

\_\_\_\_\_  
(подпись, инициалы, фамилия)

«\_\_» \_\_\_\_\_ 20\_\_ г.

## ЗАДАНИЕ НА ВЫПУСКНУЮ КВАЛИФИКАЦИОННУЮ РАБОТУ ПО ПРОГРАММЕ БАКАЛАВРИАТА

Студента Копылова С.Е. шифр 17-06-0250 группа ПО-71з

1. Тема «Система управления IT-услугами на основе концепции ITSM» утверждена приказом ректора ЮЗГУ от «17» ноября 2021 г. № 4592-с.

2. Срок предоставления работы к защите 06.12.2021 г.

3. Исходные данные для создания программно-информационной системы:

3.1 Перечень решаемых задач: разработать программно – информационную систему управления IT – услугами на основе концепции ITSM.

3.2 Входные данные и требуемые результаты для программы: Входными данными для программно-информационной системы являются: данные справочников комплектующих, конфигураций, ПО, критериев качества SLA, IT-услуг, департаментов компании; технические данные IT-ресурсов; данные входящих заявок на IT-ресурсы; данные запросов поставщикам на комплектующие.

Выходными данными для программной системы являются: сформированные заявки на обслуживание IT-ресурсов; сформированные запросы на закупку комплектующих; сведения о выполненных работах по заявкам; статусы заявок; выходные отчеты (инфографика) – по качеству услуг, по состоянию IT-ресурсов, по деятельности IT-отдела, по стоимости обслуживания IT-ресурсов, воронка заявок.

4. Содержание работы (по разделам):

4.1. Введение

4.2. Техническое задание: основание для разработки, назначение разработки, требования к программно-информационной системе, требования к оформлению документации.

4.3. Технический проект: общие сведения о программно-информационной системе, проект данных программно-информационной системы, проектирование архитектуры программно-информационной системы, проектирование пользовательского интерфейса программно-информационной системы.

4.4. Рабочий проект: спецификация компонентов и классов программно-информационной системы, тестирование программно-информационной системы, сборка компонентов программно-информационной системы.

4.5. Заключение

4.6. Список использованных источников

5. Перечень графического материала:

Лист 1. Сведения о ВКРБ

Лист 2. Цель и задачи ВКРБ

Лист 3. Диаграмма классов для концептуальной предметной области

Лист 4. Диаграмма прецедентов для программно-информационной системы

Лист 5. Диаграмма последовательности для прецедентов типа ведения справочников

Лист 6. Диаграмма данных

Лист 7. Диаграмма компонентов

Лист 8. Диаграмма классов объектной структуры предметной области

Лист 9. Результаты разработки. Сформированные отчеты. Часть 1

Лист 10. Результаты разработки. Сформированные отчеты. Часть 2

Лист 11. Результаты разработки. Сформированные отчеты. Часть 3

Лист 12. Заключение

Руководитель работы

\_\_\_\_\_ (подпись, дата)

**В.В. Апальков**

\_\_\_\_\_ (инициалы, фамилия)

Задание принял к исполнению

\_\_\_\_\_ (подпись, дата)

**С.Е. Копылов**

\_\_\_\_\_ (инициалы, фамилия)

## РЕФЕРАТ

Объем работы равен 121 странице. Работа содержит 36 иллюстраций, 22 таблицы, 30 библиографических источников и 12 листов графического материала. Количество приложений – 2. Графический материал представлен в приложении А. Текст программы представлен в приложении Б.

Перечень ключевых слов: информационная система, ИТ-ресурсы, ИТ-услуга, ITSM, ITIL, автоматизация, бизнес-процесс, технический проект системы.

Объектом разработки является программно-информационная система управления ИТ-услугами, построенную в соответствии с концепцией ITSM.

В процессе создания программно-информационной системы были разработаны методы и классы для работы с сущностями предметной области и ее отображением в базе данных MS SQL Server 2017.

Целью выпускной квалификационной работы является автоматизация процесса управления ИТ-инфраструктурой предприятия.

При создании программно-информационной системы применялась парадигма объектно-ориентированного программирования.

Для размещения программы необходимо наличие сервера СУБД MySQL версии не ниже 6.0. Клиентские компьютеры должны обладать следующими характеристиками: процессор с тактовой частотой 1.8 ГГц, ОЗУ 512 Мб, место на жестком диске не менее 130 Мб, операционная система Windows, не ниже Windows 7, установленная на компьютер программная платформа .NET Framework 4.

Данная программно-информационная система находится на стадии внедрения. Её можно использовать на предприятиях, в которых важно интегрировать ИТ-сегмент в общую структуру бизнес-процессов.

## ABSTRACT

The amount of work is 121 pages. The work contains 36 illustrations, 22 tables, 30 bibliographic sources and 12 sheets of graphic material. The number of applications is 2. The graphic material is presented in annex A. The text of the programme is presented in annex B.

List of keywords: information system, IT resources, IT service, ITSM, ITIL, automation, business process, technical design of the system.

The object of development is a software and information system for managing IT services, built in accordance with the ITSM concept.

In the process of creating a software information system, methods and classes were developed to work with the entities of the subject area and its mapping in the MS SQL Server 2017 database.

The purpose of the graduation qualification work is to automate the process of managing the IT infrastructure of the enterprise.

When creating a software-information system, the paradigm of object-oriented programming was used.

To host the program, you must have a DBMS server MySQL version 6.0 or higher. Client computers must have the following characteristics: a processor with a clock speed of 1.8 GHz, RAM of 512 MB, hard disk space of at least 130 MB, Windows operating system, at least Windows 7, installed on the computer .NET Framework 4 software platform.

This software and information system is under implementation. It can be used in enterprises where it is important to integrate the IT segment into the overall business process structure.

## СОДЕРЖАНИЕ

ВВЕДЕНИЕ	37
1 Техническое задание	39
1.1 Основание для разработки	39
1.2 Назначение разработки	39
1.3 Требования к программно-информационной системе	39
1.3.1 Требования к данным программно-информационной системы	39
1.3.2 Функциональные требования к программно-информационной системе	40
1.3.3 Нефункциональные требования к программно-информационной системе	51
2 Технический проект	55
2.1 Общие сведения о программно-информационной системе	55
2.2 Проект данных программно-информационной системы	55
2.2.1 Структура базы данных	55
2.2.2 Схема данных	62
2.3 Проектирование архитектуры программной системы	64
2.3.1 Компоненты программно-информационной системы	64
2.3.2 Архитектура программно-информационной системы	72
2.3.3 Комплекс технических средств программно-информационной системы	76
2.4 Проектирование пользовательского интерфейса программно-информационной системы	78
3 Рабочий проект	86
3.1 Спецификация компонентов и классов системы	86
3.2 Тестирование программной системы	89
3.2.1 Модульное тестирование классов и методов программы	89
3.2.2 Системное тестирование программно-информационной системы	95
3.3 Сборка компонентов программно-информационной системы	109
ЗАКЛЮЧЕНИЕ	111

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	113
ПРИЛОЖЕНИЕ А Представление графического материала	118
ПРИЛОЖЕНИЕ Б Текст программы	1
На отдельных листах (CD-RW в прикрепленном конверте)	17
Сведения о ВКРБ (Графический материал / Сведения о ВКРМ.png)	Лист 1
Цель и задачи ВКР (Графический материал / Цель и задачи.png)	Лист 2
Диаграмма классов для концептуальной предметной области (Графический материал / Диаграмма классов.png)	Лист 3
Диаграмма прецедентов для программно-информационной системы (Графический материал / Диаграмма прецедентов.png)	Лист 4
Диаграмма последовательности для прецедентов типа ведения справочников (Графический материал / Диаграмма последовательности.png)	Лист 5
Диаграмма данных (Графический материал / Диаграмма данных.png)	Лист 6
Диаграмма компонентов (Графический материал / Диаграмма компонентов.png)	Лист 7
Диаграмма классов объектной структуры предметной области (Графический материал / Диаграмма классов.png)	Лист 8
Результаты разработки. Сформированные отчеты. Часть 1 (Графический материал / Отчет 1.png)	Лист 9
Результаты разработки. Сформированные отчеты. Часть 2 (Графический материал / Отчет 2.png)	Лист 10
Результаты разработки. Сформированные отчеты. Часть 3 (Графический материал / Отчет 3.png)	Лист 11
Заключение (Графический материал / Заключение.png)	Лист 12

## ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ

БД – база данных

ИС – информационная система

ИТ – информационные технологии

ИТИ – ИТ-инфраструктура

ИТР – ИТ-ресурсы

КТС – комплекс технических средств

ОМТС – отдел материально-технического снабжения

ПО – программное обеспечение

РП – рабочий

СУБД – система управления базами данных

ТЗ – техническое задание

ТП – технический проект

ИТ (Information Technologies) – приёмы, способы и методы применения средств вычислительной техники при выполнении функций сбора, хранения, обработки, передачи и использования данных

ITIL (Information Technologies Infrastructure Library) – это методология управления, отладки и непрерывного улучшения бизнес-процессов.

ITSM (Information Technologies Service Management) – подход к управлению и организации ИТ-услуг, направленный на удовлетворение потребностей бизнеса.

SD (Service Desk) – инструмент технической поддержки пользователей.

SLA (Service License Agreement) – инструмент непрерывной оценки и управления качеством предоставления ИТ – услуг.

UML (Unified Modelling Language) – язык графического описания для объектного моделирования в области разработки программного обеспечения.

## ВВЕДЕНИЕ

Любое современное предприятие не может функционировать без развитой ИТ-инфраструктуры, включающей в себя, как минимум: компьютеры, оргтехнику, внутреннюю сеть. Подразделение, осуществляющее обслуживание ИТ-инфраструктуру предприятия, должно рассматриваться не как «механический» департамент, выполняющий действия по запросу пользователя и существующий отдельно от всей остальной структуры предприятия, но как подразделение, предоставляющее всем остальным бизнес-подразделениям ИТ-услуги. Цель функционирования ИТ-подразделения должна обеспечивать поддержание ИТ-услуг на том качественном уровне, который будет согласован с подразделениями основных бизнес-процессов (клиентами предоставляемых сервисов) [1]. Так, ИТ-подразделение рассматривается неразрывно от всей остальной бизнес-структуры предприятия. Такой подход интегрированности ИТ в бизнес принят как основной в рамках распространенной методологии IT Service Management (ITSM). Данная методология была разработана в конце 1980-х и является результатом многолетнего накопленного опыта и информации в области менеджмента ИТ-сервисов. ITSM – методология, ориентированная на потребителя ИТ-услуг [2]. ITSM была разработана на основе библиотеки ITIL (Information Technologies Infrastructure Library) [3]. В 2005 году был создан международный стандарт для управления и обслуживания ИТ-сервисов ISO 20000. В 2010 году был утверждён российский ГОСТ Р ИСО/МЭК 20000 «Информационная технология. Менеджмент услуг» также в двух частях [4].

Таким образом, очевидно, что ИТ-подразделения в соответствии с современными требованиями бизнеса должны интегрироваться в бизнес-структуру предприятия, а понятие ИТ-услуга должно стать частью общих внутренних бизнес-процессов. Для поддержания методологии ITSM для эффективного управления ИТ-услугами в рамках предприятия должна быть организована информационная система, объединяющая в себе клиентскую

подсистему, понятия ИТ-услуги, заявки, соглашения по качеству, а также систему визуализации структуры ИТ-сегмента в разрезе общих бизнес-процессов [5].

Цель настоящей работы – автоматизация процесса управления ИТ-инфраструктурой предприятия. Для достижения поставленной цели в ходе работы будет решена следующая задача: разработка программно-информационной системы управления ИТ-услугами на основе концепции ITSM. (обозначение системы: ITMS.System).

Данные задачи будут решены с помощью использования специальных инструментов проектирования систем:

системы разработки программного обеспечения (ПО) IDE Visual Studio 2019 C# .NET;

системы управления базами данных (СУБД) MS SQL Server 2017;

системы разработки функций базы данных MS SQL Management Studio;

CASE-системы Visual Paradigm for UML версии 16.2 с поддержкой code-engineering [7-13];

CASE-системы Sparx Enterprise Architect версии 15.0 с поддержкой технологии forward database-engineering [14].

## **1 Техническое задание**

### **1.1 Основание для разработки**

Полное наименование системы: система управления ИТ-услугами «IT Services Management System».

Краткое обозначение системы: «ITSM.System».

Основанием для разработки программы является приказ ректора ЮЗГУ от «17» ноября 2021 г. №4592-с «Об утверждении тем выпускных квалификационных работ».

### **1.2 Назначение разработки**

Программный продукт предназначен для повышения эффективности менеджмента управления ИТ-сервисами за счет применения новых средств автоматизации и учета [15], приведения процесса управления ИТ-сервисами на предприятии в соответствии с практиками ITIL, введения справочников ИТ-услуг, качественных оценок по SLA, получения новых средств анализа результатов деятельности ИТ-департамента и оценки состояния ИТ-ресурсов предприятия.

### **1.3 Требования к программно-информационной системе**

#### **1.3.1 Требования к данным программно-информационной системы**

Входными данными для программно-информационной системы являются:

данные справочников комплектующих, конфигураций, ПО, критериев качества SLA, ИТ-услуг, департаментов компании;

технические данные ИТ-ресурсов;

данные входящих заявок на ИТ-ресурсы;

данные запросов поставщикам на комплектующие

Выходными данными для программной системы являются:

сформированные заявки на обслуживание ИТ-ресурсов;

сформированные запросы на закупку комплектующих;

сведения о выполненных работах по заявкам;

статусы заявок;



спецификации программного обеспечения [17-20]. На рисунке 1.2 представлена диаграмма прецедентов ITSM.System.

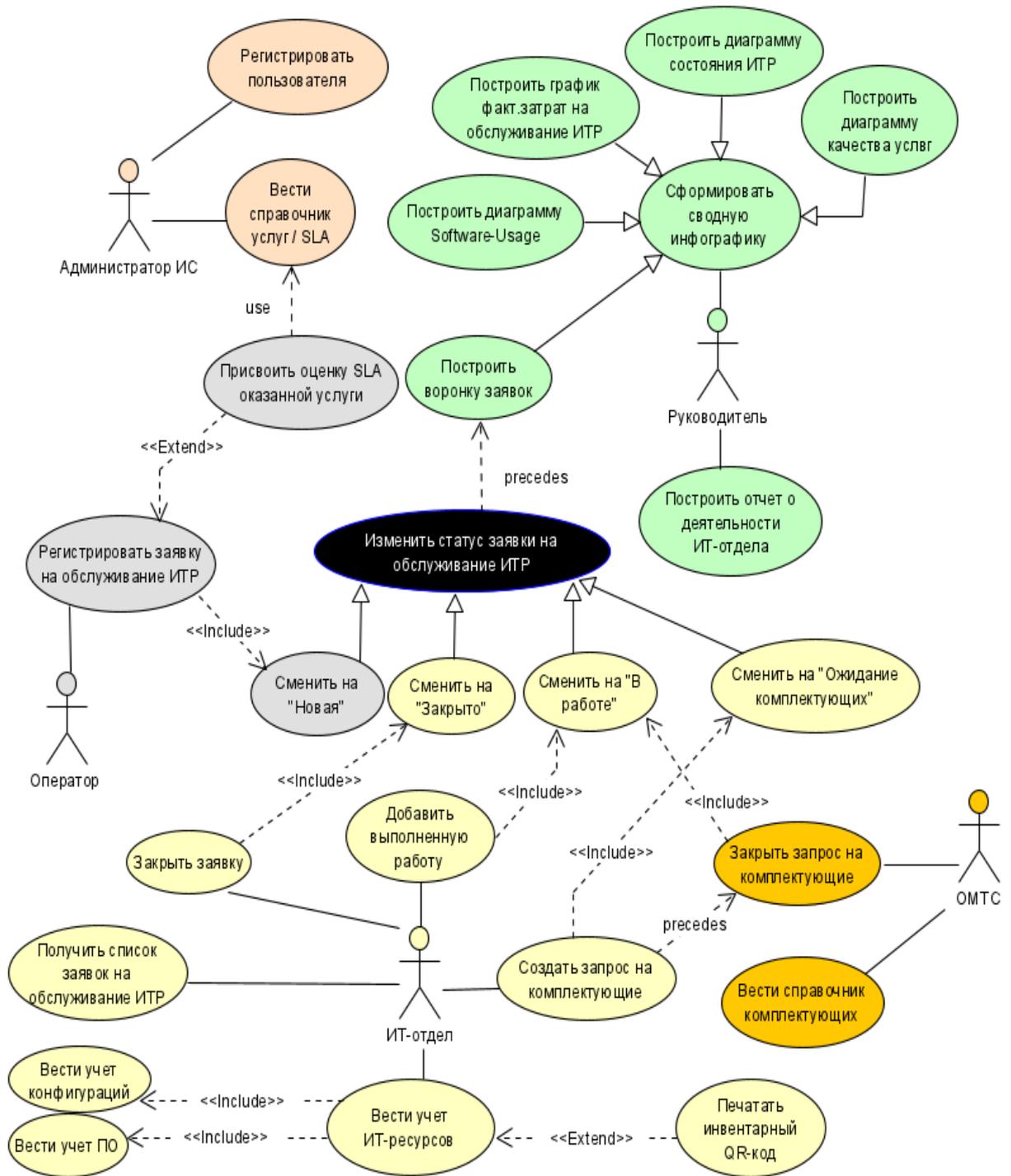


Рисунок 1.2 – Функциональные требования к ITSM.System в виде диаграммы прецедентов

В таблице 1.1 приведено описание сценария прецедента «Регистрация пользователя».

Таблица 1.1 – Сценарий прецедента «Регистрация пользователя»

Раздел	Описание
Основной исполнитель	Администратор ИС
Заинтересованные лица и их требования	Администратор должен иметь возможность создавать аккаунты пользователей, пользователи должны иметь возможность авторизовываться в ITSM.System
Предусловие	Необходимость создания нового аккаунта пользователя в ITSM.System
Постусловие	В ITSM.System создан аккаунт нового пользователя
Основной поток событий	
Действие пользователя	Реакция системы
<ol style="list-style-type: none"> <li>1. Входит в подсистему создания аккаунтов</li> <li>2. Выполняет команду создания аккаунта пользователя</li> <li>4. Вводит данные аккаунта пользователя и подтверждает ввод</li> </ol>	<ol style="list-style-type: none"> <li>3. Выводит диалог ввода данных аккаунта пользователя</li> <li>5. В БД создается запись о новом аккаунте пользователя (A1.1)</li> </ol>
Альтернативный поток событий	
Действие пользователя	Реакция системы
A1.1: подключение к БД не выполнено	
	<ol style="list-style-type: none"> <li>1. Выводит ошибку с информированием об отсутствии соединения с БД</li> <li>2. Выводит панель для настройки подключения к БД (адрес сервера, учетная запись для подключения, порт, имя БД – конкретные</li> </ol>

	параметры подключения к БД установить на этапе разработки)
--	---

В таблице 1.2 приведено описание сценария прецедента «Вести учет справочника услуг / SLA».

Таблица 1.2 – Сценарий прецедента «Вести учет справочника услуг / SLA»

Раздел	Описание
Основной исполнитель	Администратор ИС
Заинтересованные лица и их требования	Администратор ИС должен иметь возможность вести справочник услуг / SLA
Предусловие	В систему загружен справочник услуг / SLA
Постусловие	В справочник услуг / SLA добавлены (отредактированы) записи
Основной поток событий	
Действие пользователя	Реакция системы
2. Выбирает действие	<ol style="list-style-type: none"> <li>1. Запрашивает требуемое действие (добавить запись, редактировать запись, удалить запись)</li> <li>3. Выполняет один из подчиненных сценариев (П2.1, П2.2, П2.3) в соответствии с выбранным действием</li> <li>4. Обновляет данные в таблице подсистемы</li> </ol>
Подчиненный сценарий П2.1: Добавление записи	
Действие пользователя	Реакция системы
2. Вводит данные в поля, предложенные диалогом ввода данных	<ol style="list-style-type: none"> <li>1. Выводит форму диалога ввода данных услуги / SLA</li> <li>3. Заносит новую запись в БД в соответствующую таблицу</li> <li>4. Переходит к п.4 основного сценария</li> </ol>
Подчиненный сценарий П2.2: Редактирование записи	
Действие пользователя	Реакция системы
	1. Выводит форму диалога редактирования данных услуги / SLA

2. Редактирует одно или несколько полей	3. Редактирует в БД соответствующую запись в соответствующей таблице 4. Переходит к п.4 основного сценария
Подчиненный сценарий П2.3: Удаление записи	
Действие пользователя	Реакция системы
2. Подтверждает удаление данных (A2.1)	1. Запрашивает подтверждение удаления записи 3. Удаляет из БД соответствующую запись из таблицы 4. Переходит к п.4 основного сценария

Продолжение таблицы 1.2

Альтернативный поток событий	
Действие пользователя	Реакция системы
A2.1: отменяет удаление данных	
	1. Переходит к п.1 основного сценария

В таблице 1.3 приведено описание сценария прецедента «Вести учет конфигураций».

Таблица 1.3 – Сценарий прецедента «Вести учет конфигураций»

Раздел	Описание
Основной исполнитель	ИТ-отдел (ИТ)
Заинтересованные лица и их требования	ИТ должен иметь возможность вести справочник конфигураций
Предусловие	В систему загружен справочник конфигураций
Постусловие	В справочник конфигураций добавлены (отредактированы) записи
Основной поток событий	
Действие пользователя	Реакция системы
· Выбирает действие	· Запрашивает требуемое действие (добавить запись, редактировать запись, удалить запись) · Выполняет один из подчиненных сценариев (ПЗ.1, ПЗ.2, ПЗ.3) в соответствии с выбранным действием

	. Обновляет данные в таблице подсистемы
Подчиненный сценарий ПЗ.1: Добавление записи	
Действие пользователя	Реакция системы
2. Вводит данные в поля, предложенные диалогом ввода данных	1. Выводит форму диалога ввода данных конфигурации 3. Заносит новую запись в БД в соответствующую таблицу 4. Переходит к п.4 основного сценария

Продолжение таблицы 1.3

Подчиненный сценарий ПЗ.2: Редактирование записи	
Действие пользователя	Реакция системы
2. Редактирует одно или несколько полей	<ol style="list-style-type: none"> <li>1. Выводит форму диалога редактирования данных конфигурации</li> <li>3. Редактирует в БД соответствующую запись в соответствующей таблице</li> <li>4. Переходит к п.4 основного сценария</li> </ol>
Подчиненный сценарий ПЗ.3: Удаление записи	
Действие пользователя	Реакция системы
2. Подтверждает удаление данных (А3.1)	<ol style="list-style-type: none"> <li>1. Запрашивает подтверждение удаления записи</li> <li>3. Удаляет из БД соответствующую запись из таблицы</li> <li>4. Переходит к п.4 основного сценария</li> </ol>
Альтернативный поток событий	
Действие пользователя	Реакция системы
А3.1: отменяет удаление данных	
	1. Переходит к п.1 основного сценария

В таблице 1.4 приведено описание сценария прецедента «Вести учет ПО».

Таблица 1.4 – Сценарий прецедента «Вести учет ПО»

Раздел	Описание
Основной исполнитель	ИТ-отдел (ИТ)
Заинтересованные лица и их требования	ИТ должен иметь возможность вести справочник программного обеспечения
Предусловие	В систему загружен справочник ПО
Постусловие	В справочник ПО добавлены (отредактированы) записи
Основной поток событий	
Действие пользователя	Реакция системы

2. Выбирает действие	<ol style="list-style-type: none"> <li>1. Запрашивает требуемое действие (добавить запись, редактировать запись, удалить запись)</li> <li>3. Выполняет один из подчиненных сценариев (П4.1, П4.2, П4.3) в соответствии с выбранным</li> </ol>
----------------------	---

Продолжение таблицы 1.4

	действием <ol style="list-style-type: none"> <li>4. Обновляет данные в таблице подсистемы</li> </ol>
--	---

Подчиненный сценарий П4.1: Добавление записи

Действие пользователя	Реакция системы
2. Вводит данные в поля, предложенные диалогом ввода данных	<ol style="list-style-type: none"> <li>1. Выводит форму диалога ввода данных ПО</li> <li>3. Заносит новую запись в БД в соответствующую таблицу</li> <li>4. Переходит к п.4 основного сценария</li> </ol>

Подчиненный сценарий П4.2: Редактирование записи

Действие пользователя	Реакция системы
2. Редактирует одно или несколько полей	<ol style="list-style-type: none"> <li>1. Выводит форму диалога редактирования данных ПО</li> <li>3. Редактирует в БД соответствующую запись в соответствующей таблице</li> <li>4. Переходит к п.4 основного сценария</li> </ol>

Подчиненный сценарий П4.3: Удаление записи

Действие пользователя	Реакция системы
2. Подтверждает удаление данных (А4.1)	<ol style="list-style-type: none"> <li>1. Запрашивает подтверждение удаления записи</li> <li>3. Удаляет из БД соответствующую запись из таблицы</li> <li>4. Переходит к п.4 основного сценария</li> </ol>

Альтернативный поток событий

Действие пользователя	Реакция системы
A4.1: отменяет удаление данных	
	2. Переходит к п.1 основного сценария

В таблице 1.5 приведено описание сценария прецедента «Вести учет ИТ-ресурсов».

Таблица 1.5 – Сценарий прецедента «Вести учет ИТ-ресурсов»

Раздел	Описание
Основной исполнитель	ИТ-отдел (ИТ)
Заинтересованные лица и их требования	ИТ должен иметь возможность вести справочник ИТ-ресурсов
Предусловие	В систему загружен справочник ИТ-ресурсов
Постусловие	В справочник ИТ-ресурсов добавлены (отредактированы) записи
Основной поток событий	
Действие пользователя	Реакция системы
2. Выбирает действие	<p>1. Запрашивает требуемое действие (добавить, редактировать, удалить запись)</p> <p>3. Выполняет один из подчиненных сценариев (П5.1, П5.2, П5.3)</p> <p>4. Обновляет данные в таблице подсистемы</p>
Подчиненный сценарий П5.1: Добавление записи	
Действие пользователя	Реакция системы
<p>2. Вводит данные в поля ввода</p> <p>3. Указывает конфигурацию ИТ-ресурса</p> <p>4. Указывает примененное ПО</p>	<p>1. Выводит форму диалога ввода данных ИТ-ресурса</p> <p>5. Заносит новую запись в БД в соответствующую таблицу</p> <p>6. Переходит к п.4 основного сценария</p>
Подчиненный сценарий П5.2: Редактирование записи	
Действие пользователя	Реакция системы
2. Редактирует одно или несколько полей	1. Выводит форму диалога редактирования данных ИТ-ресурса

	3. Редактирует в БД соответствующую запись в соответствующей таблице 4. Переходит к п.4 основного сценария
Подчиненный сценарий П5.3: Удаление записи	
Действие пользователя	Реакция системы
2. Подтверждает удаление данных (A5.1)	1. Запрашивает подтверждение удаления записи 3. Удаляет из БД соответствующую запись из таблицы

Продолжение таблицы 1.5

	4. Переходит к п.4 основного сценария
Альтернативный поток событий	
Действие пользователя	Реакция системы
A5.1: отменяет удаление данных	
	1. Переходит к п.1 основного сценария

На рисунке 1.3 показана диаграмма деятельности, соответствующая действиям по ведению любого из справочников.

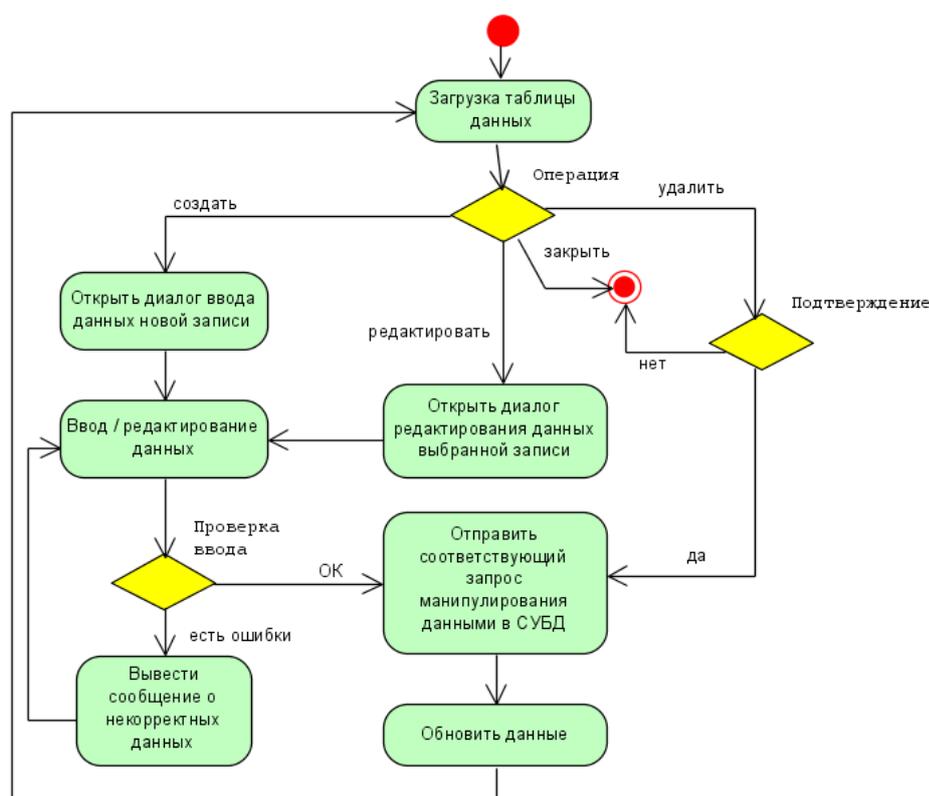


Рисунок 1.3 – Диаграмма деятельности прецедентов ведения справочников

### 1.3.3 Нефункциональные требования к программно-информационной системе

#### 1.3.3.1 Требования к надежности

Время восстановления после отказа, вызванного сбоем электропитания технических средств (иными внешними факторами), не фатальным сбоем (не крахом) операционной системы, не должно превышать времени, необходимого на перезагрузку операционной системы и запуск программы, при условии соблюдения условий эксплуатации технических и программных средств.

Время восстановления после отказа, вызванного неисправностью технических средств, фатальным сбоем (крахом) операционной системы, не должно превышать времени, требуемого на устранение неисправностей технических средств и переустановку программных средств.

Время восстановления после отказа, вызванного неисправностью на сервере, не должно превышать времени, требуемого на устранение неисправностей.

Под аварийной ситуацией понимается аварийное завершение рабочего процесса, выполняемого ITSM.System, а также «зависание» этого процесса.

При работе ITSM.System возможны следующие аварийные ситуации, которые влияют на надежность работы системы:

сбой в электроснабжении сервера базы данных;

сбой в электроснабжении персонального компьютера пользователя;

сбой в электроснабжении обеспечения сети;

сбой программного обеспечения сервера.

### **1.3.3.2 Требования к защите информации от несанкционированного доступа**

При аутентификации пользователей ITSM.System должно использоваться хеширование паролей.

Должно быть соблюдено разграничение прав доступа к функциональности ITSM.System.

СУБД должна быть защищена паролем администратора.

Доступ к структурам таблиц базы данных должен иметь только уполномоченный системный администратор предприятия.

### **1.3.3.3 Требования к лингвистическому обеспечению**

При реализации системы должны применяться следующие языки высокого уровня: объектно-ориентированный язык программирования C#, декларированный язык запросов SQL, стандартизированный язык разметки HTML, язык описания физической структуры базы данных – DDL.

Для реализации алгоритмов манипулирования данными в БД необходимо использовать СУБД SQL Server версии 2017 и выше.

Для реализации диалога ITSM.System с пользователями должен применяться графический оконный пользовательский интерфейс.

### **1.3.3.4 Требования к программному обеспечению**

Системные программные средства, используемые ITSM.System, должны быть представлены локализованной версией операционной системы Windows. Заранее должны быть установлены библиотеки .NET Framework версии 4.6 и

выше (поставляются комплектно с операционными системами Microsoft Windows версии 8.1 и выше), а также Microsoft Office для работы с документами.

#### **1.3.3.5 Требования к техническому обеспечению**

КТС (комплекс технических средств) ITSM.System должен включать:  
набор автоматизированных рабочих мест (на каждого пользователя из пользовательской роли) с установленной ITSM.System;  
сервер СУБД.

Требования к персональному рабочему месту пользователя должны определяться системными требованиями, достаточными для запуска и нормальной стабильной работы операционной системы Microsoft Windows версии 7 и выше.

Требования к серверу СУБД должны определяться системными требованиями, достаточными для запуска и нормальной стабильной работы операционной системы Microsoft Windows Server версии 2016 или выше.

#### **1.3.3.6 Требования к комплектности поставки системы**

Комплект, сдаваемый разработчиком заказчику, должен включать:  
скомпилированные исполняемые файлы ITSM.System со всеми необходимыми библиотеками (кроме стандартных системных);

исходные коды ITSM.System с комментариями;

описание требований к установке дополнительных компонентов среды разработки, необходимых для корректного открытия проекта ITSM.System (если имеются);

файл с описанием физической структуры базы данных на языке описания данных DDL;

файл дампа базы данных MS SQL Server 2017 с подготовленными тестовыми наборами данных, который можно импортировать в физическую среду СУБД;

технический проект ITSM.System, выполненный в виде файлов моделей в CASE-системе;

пояснительная записка, включающая техническое задание на разработку ITSM.System, технический проект ITSM.System, рабочий проект ITSM.System.

## **2 Технический проект**

### **2.1 Общие сведения о программно-информационной системе**

Полное наименование системы: система управления ИТ-услугами «IT Services Management System». Краткое обозначение системы: «ITSM.System».

ITSM.System предназначена для повышения эффективности менеджмента ИТ-сервисами за счет применения новых средств автоматизации и учета, приведения процесса управления ИТ-сервисами на предприятии в соответствии с практиками ITIL, введения справочников ИТ-услуг, качественных оценок по SLA, получения новых средств анализа результатов деятельности ИТ-департамента и оценки состояния ИТ-ресурсов предприятия [21].

ITSM.System предназначена для эксплуатации в нормальных условиях работы с обеспечением охраны помещений и информационной безопасности.

ITSM.System предполагает клиент-серверную архитектуру с использованием централизованного источника данных (на основе СУБД SQL SERVER 2017), доступ к которой осуществляется каждым пользователем со своего рабочего места. Предварительно каждый пользователь должен пройти процедуру авторизации в программно-информационной системе.

### **2.2 Проект данных программно-информационной системы**

#### **2.2.1 Структура базы данных**

База данных ITSM.System должна включать информацию обо всех сущностях предметной области [22]. Так, каталог базы данных ITSM.System будет содержать следующие справочные (с) и оперативные (о) таблицы:

пользователи (с);

сервисы (с);

оценки качества (с);

программное обеспечение (с);

комплектующие (с);

оборудование (с);

подразделения (с);

- конфигурации (с);
- заявки пользователей (о);
- запросы на комплектующие (о);
  - выполненные работы (о).

Подробный реквизитный состав перечисленных таблиц приведен в таблицах 2.1 – 2.11.

В таблице 2.1 приведена спецификация таблицы TConfig, содержащей типовые конфигурации оборудования.

Таблица 2.1 – Спецификация таблицы TConfig

Поле	Тип	Обязательное	Характеристика
c_id	int	True	Идентификатор конфигурации оборудования
c_name	varchar (100)	True	Наименование типовой конфигурации оборудования
c_cpu	varchar (255)	False	Описание характеристик процессорного устройства
c_graphic	varchar (255)	False	Описание характеристик графического устройства (графический адаптер, GPU, дисплейный модуль и т.д.)
c_mem	varchar (255)	False	Описание характеристик памяти
c_net	varchar (255)	False	Описание характеристик сетевого устройства
c_interface	varchar (255)	False	Описание интерфейсов
c_volume	varchar (255)	False	Описание характеристик хранилищ данных (дисковые устройства, прочие накопители)
c_basic	varchar (255)	False	Описание характеристик базового компонента (материнской платы, например)

Продолжение таблицы 2.1

Поле	Тип	Обязательное	Характеристика
c_additional	varchar (255)	False	Описание дополнительных характеристик, не включенных в предыдущие разделы конфигурации

В таблице 2.2 приведена спецификация таблицы TDepartment, содержащей данные подразделений.

Таблица 2.2 – Спецификация таблицы TDepartment

Поле	Тип	Обязательное	Характеристика
d_id	int	True	Идентификатор подразделения
d_name	varchar (50)	True	Наименование подразделения
d_person	varchar (50)	True	Ф.И.О. руководителя подразделения
d_post	varchar (50)	False	Должность руководителя подразделения
d_phone	varchar (12)	True	Номер контактного телефона руководителя

В таблице 2.3 приведена спецификация таблицы TEquipment, содержащей инвентарный учет ИТ-ресурсов.

Таблица 2.3 – Спецификация таблицы TEquipment

Поле	Тип	Обязательное	Характеристика
e_id	int	True	Инвентарный номер оборудования
e_type	varchar (50)	True	Тип оборудования из фиксированного списка
e_config_id	int	False	Идентификатор конфигурации
e_department_id	int	True	Идентификатор департамента, ответственного за оборудование
e_iss_date	date	True	Дата ввода оборудования в эксплуатацию
e_status	varchar (20)	True	Автоматически изменяемый статус оборудования: используется, в ремонте, списан

В таблице 2.4 приведена спецификация таблицы TFeedback, содержащей типовые оценки качества оказанных услуг.

Таблица 2.4 – Спецификация таблицы TFeedback

Поле	Тип	Обязательное	Характеристика
fb_id	int	True	Идентификатор оценки

fb_score	int	True	Числовая характеристика оценки (балл)
fb_mark	varchar(50)	True	Текстовая характеристика оценки
fb_color	varchar(50)	True	Цветовой код оценки в формате R G B
fb_description	varchar(50)	False	Описание оценки качества оказанной услуги в соответствии со SLA

В таблице 2.5 приведена спецификация таблицы TPart, содержащей данные комплектующих, применяемых при выполнении ремонтов компьютерной техники.

Таблица 2.5 – Спецификация таблицы TPart

Поле	Тип	Обязательное	Характеристика
p_id	int	True	Идентификатор комплектующего
p_name	varchar (50)	True	Наименование комплектующего
p_features	varchar (255)	False	Характеристики комплектующего
p_vendor	varchar (50)	False	Производитель (если важен)
p_price	decimal (18)	True	Стоимость комплектующего (заполняется ОМТС)

В таблице 2.6 приведена спецификация таблицы TService, содержащей данные ИТ-сервисов (видов услуг), которые предоставляет пользователям ИТ-подразделение.

Таблица 2.6 – Спецификация таблицы TService

Поле	Тип	Обязательное	Характеристика
s_id	int	True	Идентификатор услуги в справочнике
s_category	varchar(30)	True	Категория услуги (фиксированный справочник с возможностью ввода своего варианта)
s_type	varchar(30)	True	Тип услуги (фиксированный справочник с возможностью ввода своего варианта)
s_name	varchar(50)	True	Наименование услуги
s_description	varchar(255)	False	Описание услуги
s_bp_role	varchar(255)	False	Описание роли услуги в бизнес-процессе
s_document	varchar(255)	False	Наименование документа - соглашения о качестве (SLA) услуги
s_norm	int	True	Нормативное время оказания услуги

В таблице 2.7 приведена спецификация таблицы TUser, содержащей учетные данные (аккаунты) пользователей, зарегистрированных в АИС с различными ролями.

Таблица 2.7 – Спецификация таблицы TUser

Поле	Тип	Обязательное	Характеристика
u_id	int	True	Идентификатор пользователя
u_name	varchar(50)	True	Имя пользователя
u_post	varchar(50)	False	Должность
u_role	varchar(50)	False	Описание роли пользователя
u_role_id	int	True	Идентификатор роли пользователя: 0 - оператор, 1 - отдел ОМТС, 2- отдел ИТ, 3 - руководитель, 4 - администратор
u_login	varchar(10)	True	Логин аккаунта пользователя
u_password	varchar(10)	True	Пароль аккаунта пользователя

Продолжение таблицы 2.7

Поле	Тип	Обязательное	Характеристика
u_last_enter	date	False	Дата последнего входа (последней авторизации) пользователя в систему

В таблице 2.8 приведена спецификация таблицы TSoftware, содержащей данные программного обеспечения, используемого в подразделениях компании.

Таблица 2.8 – Спецификация таблицы TSoftware

Поле	Тип	Обязательное	Характеристика
sw_id	int	True	Идентификатор программного обеспечения
sw_name	varchar (20)	True	Наименование программного обеспечения
sw_developer	varchar (20)	True	Разработчик программного обеспечения
sw_version	varchar (20)	False	Актуальная версия ПО
sw_description	varchar (255)	False	Описание программного обеспечения
sw_type	varchar (20)	True	Тип версии ПО ("коробка", SaaS, Web)
sw_license_price	decimal (18)	True	Стоимость лицензии ПО
sw_license_amount	varchar (50)	True	Срок лицензии

В таблице 2.9 приведена спецификация таблицы TOrder, содержащей данные заявок пользователей, поступающих в ИТ-подразделение.

Таблица 2.9 – Спецификация таблицы TOrder

Поле	Тип	Обязательное	Характеристика
o_id	int	True	Номер заявки пользователя
o_date_create	date	True	Дата создания заявки
o_date_complete	date	False	Дата завершения заявки
o_status	varchar(15)	True	Статус заявки

Продолжение таблицы 2.9

Поле	Тип	Обязательное	Характеристика
o_register_id	int	True	ID зарегистрировавшего заявку
o_service_id	int	True	ID сервиса заявке
o_feedback_id	int	False	ID оценки качества, оставленной пользователем по заявке
o_feedback_comment	varchar(255)	False	Комментарий к оценке качества заявки, оставленный пользователем
o_equipment_id	int	False	Инвентарный код оборудования, по которому создана заявка
o_description	text	True	Комментарий пользователя к заявке, содержащий первичное словесное описание проблемы (сути заявки)

В таблице 2.10 приведена спецификация таблицы TJob, содержащей данные работ, выполненных в ходе решения пользователей, поступающих в ИТ-подразделение.

Таблица 2.10 – Спецификация таблицы TJob

Поле	Тип	Обязательное	Характеристика
j_id	int	True	Идентификатор выполненной работы
j_order_id	int	True	Номер заявки, по которой выполнена работа
j_name	varchar (150)	True	Наименование выполненной работы
j_description	varchar (255)	False	Краткое описание выполненной работы
j_date	date	True	Дата выполнения работы

В таблице 2.11 приведена спецификация таблицы TRequest, содержащей данные запросов, созданных в ОМТС ИТ-подразделением для закупки комплектующих, необходимых для выполнения ремонтов по заявкам пользователей.

Таблица 2.11 – Спецификация таблицы TRequest

Поле	Тип	Обязательное	Характеристика
r_id	int	True	Номер запроса на комплектующие
r_part_id	int	True	Идентификатор комплектующего
r_count	int	True	Количество комплектующего
r_date_create	date	True	Дата создания запроса
r_date_complete	date	False	Дата исполнения запроса
r_comment	varchar (255)	False	Необязательный комментарий, указывается при создании запроса
r_order_id	int	True	Номер заявки пользователя, по которой запрашиваются комплектующие

### 2.2.2 Схема данных

Приведенная в таблицах 2.1 – 2.11 спецификация физических таблиц базы данных ITSM.System соответствует выбранной СУБД – SQL Server 2017.

На рисунке 2.1 приведена ER-диаграмма базы данных ITSM.System.

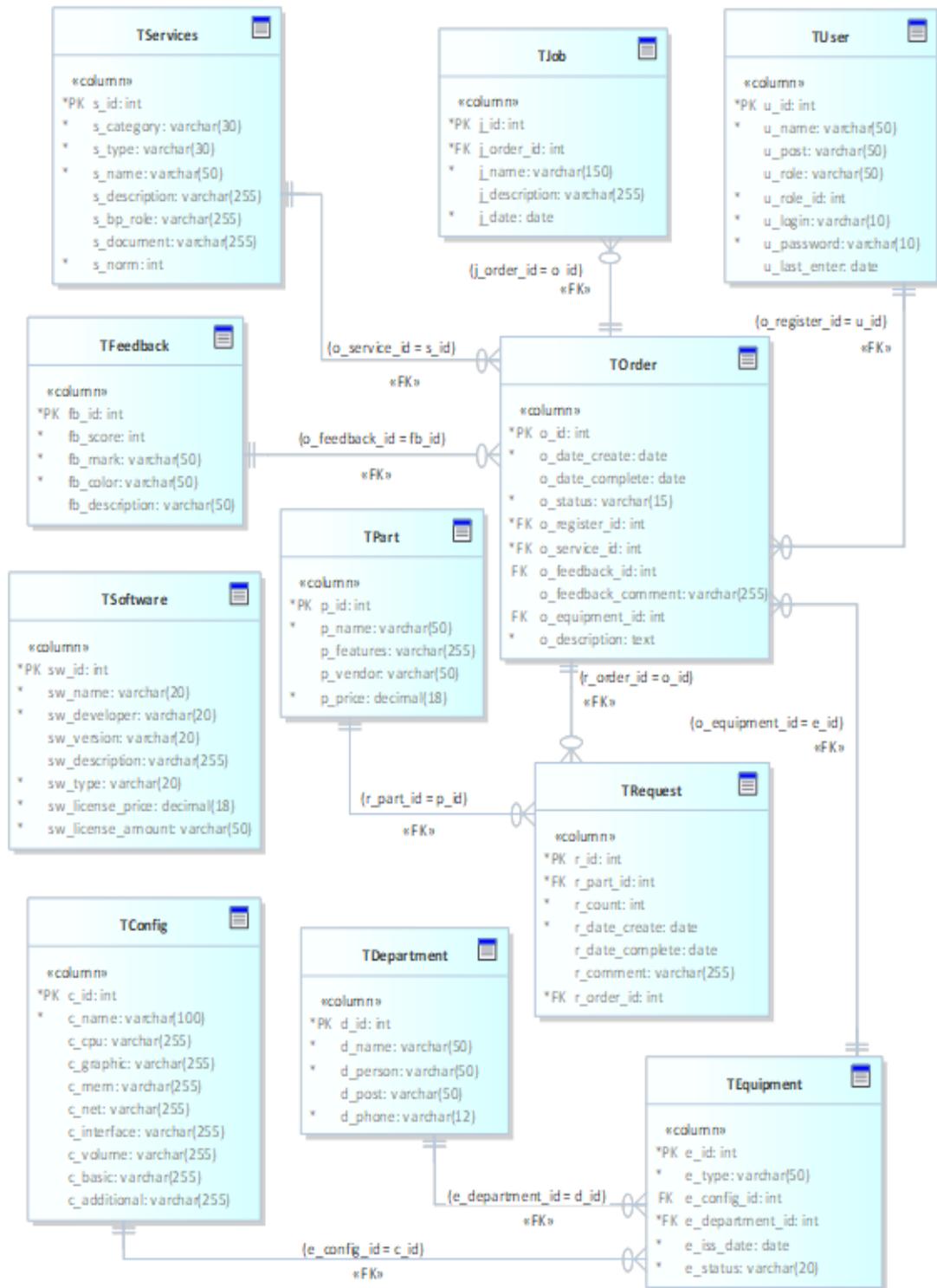


Рисунок 2.1 – Схема базы данных ITSM.System

## 2.3 Проектирование архитектуры программной системы

### 2.3.1 Компоненты программно-информационной системы

Архитектура системы определяет ее конечную сложность. По [23] архитектура системы представляет собой организационную структуру системы, в которой определены ее компоненты, их интерфейсы и концептуальные решения взаимодействия между ними. На рисунке 2.2 приведена архитектурная диаграмма пакетов программно-информационной системы ITSM.System.

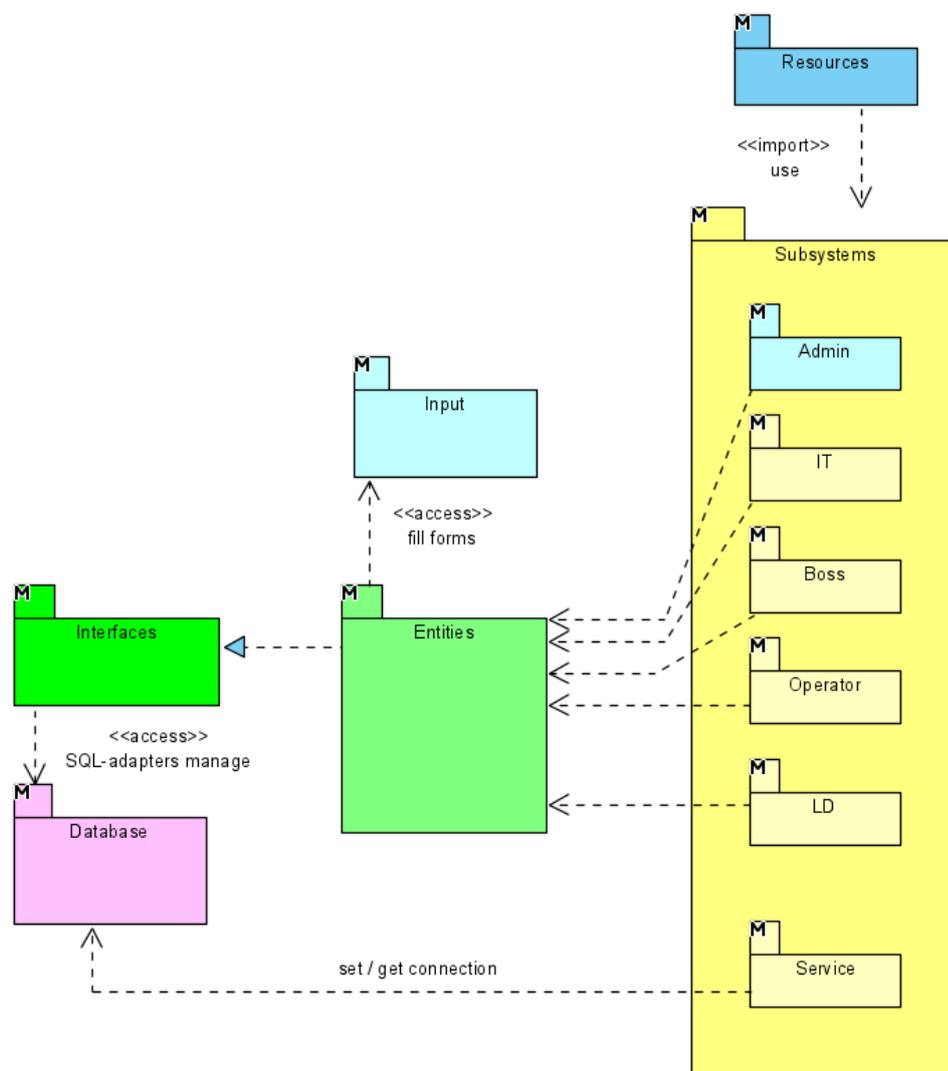


Рисунок 2.2 – UML-диаграмма пакетов ITSM.System

Спецификация диаграммы:

Resources – ресурсы программы;

Input – диалоги ввода данных;

Interfaces – интерфейсы системы;

Database – объекты отражения базы данных;

Entities – классы-сущности предметной области;

Subsystem – пакет уровня пользовательского интерфейса;

Admin – компоненты пользовательского интерфейса роли администратора системы;

IT – компоненты пользовательского интерфейса роли IT-департамента;

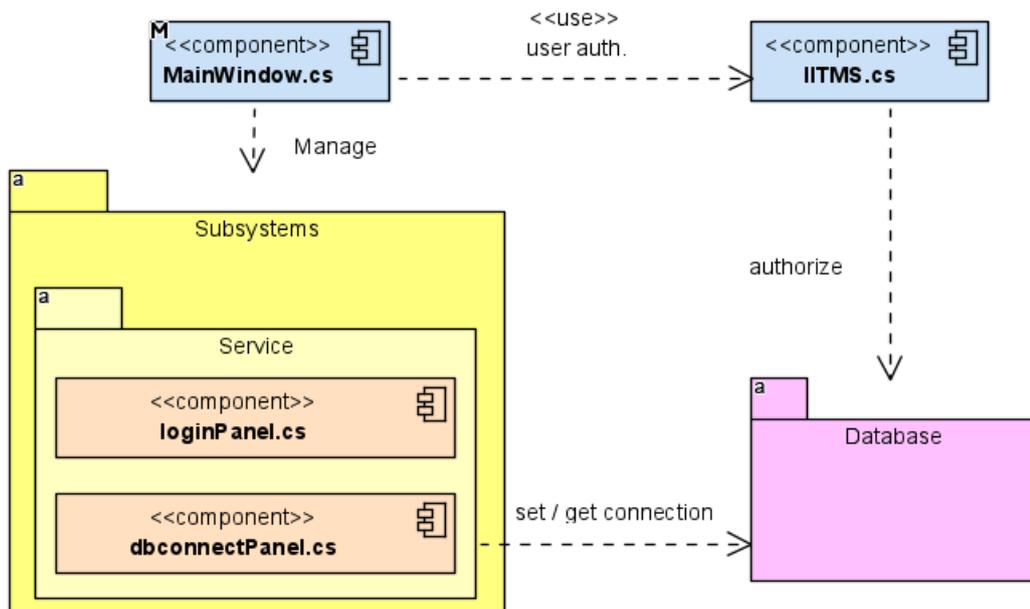
Boss – компоненты пользовательского интерфейса роли руководителя;

Operator – компоненты пользовательского интерфейса роли оператора;

LD – компоненты пользовательского интерфейса роли сотрудника ОМТС;

Services – компоненты пользовательского интерфейса сервисных функций (подключение к БД, авторизация).

На рисунке 2.3 приведена архитектурная диаграмма компонентов общих сервисных функций программно-информационной системы ITSM.System.



## Рисунок 2.3 – UML-диаграмма компонентов ITSM.System: общие сервисные функции

Спецификация диаграммы:

MainWindow – компонент главного окна системы;

loginPanel – компонент панели авторизации пользователя;

dbconnectPanel – компонент панели настройки подключения к БД;

ITMS – компонент глобальных определений системы.

На рисунке 2.3 приведена архитектурная диаграмма компонентов модели предметной области программно-информационной системы ITSM.System.

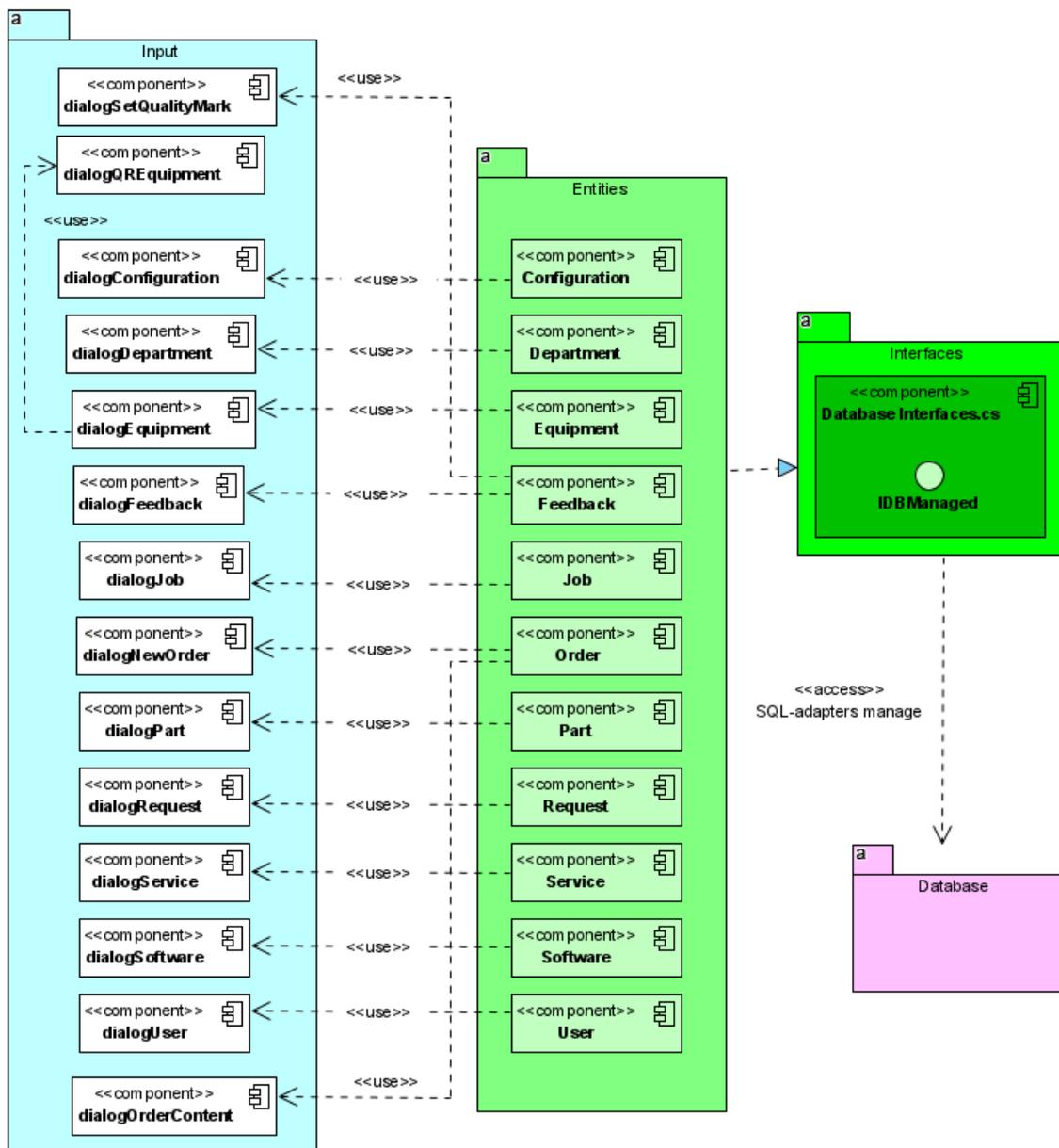


Рисунок 2.4 – UML-диаграмма компонентов ITSM.System: модель и функции предметной области

Спецификация диаграммы:

dialogSetQualityRank – диалог ввода данных оценки качества оказанной ИТ-услуги;

dialogQREquipment – форма вывода инвентарного QR-кода оборудования;

dialogConfiguration – диалог ввода данных конфигурации компьютера;

dialogDepartment – диалог ввода данных подразделения предприятия;

dialogEquipment – диалог ввода данных технических ИТ-ресурсов (например, компьютеров);

dialogFeedback – диалог ввода данных в справочник оценки качества услуг по SLA;

dialogJob – диалог ввода данных в справочник работ ИТ-отдела;

dialogNewOrder – диалог ввода данных новой заявки на обслуживания ИТ-ресурса;

dialogPart – диалог ввода данных в справочник комплектующих;

dialogRequest – диалог ввода данных новой заявки в ОМТС на приобретение комплектующих;

dialogService – диалог ввода данных в справочник ИТ-сервисов;

dialogSoftware – диалог ввода данных в справочник программного обеспечения;

dialogUser – диалог ввода данных пользователя системы;

dialogOrderContent – диалог просмотра данных по заявке на обслуживания ИТ-ресурсов;

Cofiguration – класс-сущность, реализующий объекты предметной области типа «конфигурация»;

Department – класс-сущность, реализующий объекты предметной области типа «департамент»;

Equipment – класс-сущность, реализующий объекты предметной области типа «ИТ-ресурс»;

Feedback – класс-сущность, реализующий объекты предметной области типа «Оценка качества по SLA»;

Job – класс-сущность, реализующий объекты предметной области типа «выполненная работа по заявке на обслуживание ИТ-ресурсов»;

Order – класс-сущность, реализующий объекты предметной области типа «заявка на обслуживание ИТ-ресурсов»;

Part – класс-сущность, реализующий объекты предметной области типа «комплектующие»;

Request – класс-сущность, реализующий объекты предметной области типа «заявка на закупку комплектующих по заявке на обслуживание ИТ-ресурсов»;

Service – класс-сущность, реализующий объекты предметной области типа «ИТ-сервис»;

Software – класс-сущность, реализующий объекты предметной области типа «ПО»;

User – класс-сущность, реализующий объекты предметной области типа «пользователь информационной системы»;

Database Interfaces – компонент, соержащий интерфейсы связи данных классов-сущностей с БД;

IDBManagement – интерфейс-посредник между пользователем и базой данных – двусторонний: с одной стороны – обеспечивает выполнение информационных и управляющих запросов в базу данных через классы-сущности, с другой – обеспечивает процессы манипуляции данными посредством пользовательского интерфейса, также через классы-сущности.

На рисунке 2.4 приведена архитектурная диаграмма компонентов пользовательского интерфейса подсистем программно-информационной системы ITSM.System.

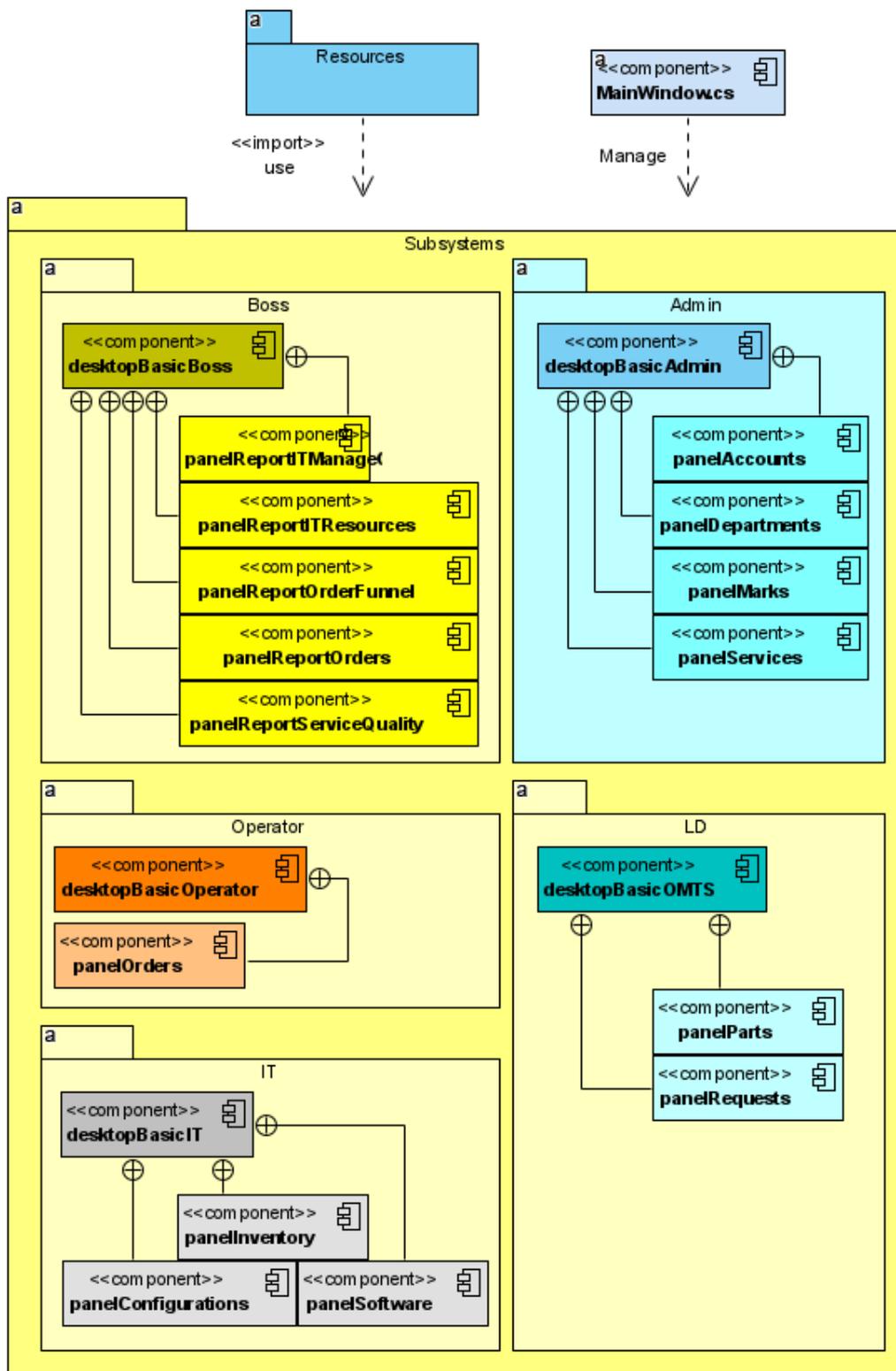


Рисунок 2.5 – UML-диаграмма компонентов ITSM.System: компоненты пользовательского интерфейса

#### Спецификация диаграммы:

desktopBasicAdmin – компонент пользовательского интерфейса роли администратора системы – рабочий стол;

panelAccounts – компонент пользовательского интерфейса роли администратора системы – подсистема ведения справочника аккаунтов пользователей;

panelDepartments – компонент пользовательского интерфейса роли администратора системы – подсистема ведения справочника подразделений предприятия;

panelMarks – компонент пользовательского интерфейса роли администратора системы – подсистема ведения справочника оценок качества услуг по SLA;

panelServicess – компонент пользовательского интерфейса роли администратора системы – подсистема ведения справочника типов ИТ-сервисов, предоставляемых ИТ-департаментом;

desktopBasicIT – компонент пользовательского интерфейса роли ИТ-департамента – рабочий стол;

panelInventory – компонент пользовательского интерфейса роли ИТ-департамента – подсистема ведения справочника ИТ-ресурсов предприятия;

panelConfiguration – компонент пользовательского интерфейса роли ИТ-департамента – подсистема ведения справочника конфигураций ИТ-ресурсов;

panelSoftware – компонент пользовательского интерфейса роли ИТ-департамента – подсистема ведения справочника программного обеспечения;

desktopBasicBoss – компонент пользовательского интерфейса роли руководителя – рабочий стол;

panelReportITManage – компонент пользовательского интерфейса роли руководителя – подсистема формирования отчета о деятельности ИТ-департамента;

panelReportTResources – компонент пользовательского интерфейса роли руководителя – подсистема формирования отчета о состоянии ИТ-ресурсов предприятия;

panelReportOrderFunnel – компонент пользовательского интерфейса роли руководителя – подсистема формирования визуального отчета (инфографика) – воронки заявок;

panelReportOrders – компонент пользовательского интерфейса роли руководителя – подсистема формирования отчета о выполнении заявок на обслуживание ИТ-ресурсов;

panelReportServiceQuality – компонент пользовательского интерфейса роли руководителя – подсистема формирования отчета о качестве услуг, оказанных ИТ-департаментом;

desktopBasicOperator – компонент пользовательского интерфейса роли оператора – рабочий стол;

panelOrders – компонент пользовательского интерфейса роли оператора – подсистема менеджмента заявок пользователей на обслуживание ИТ-ресурсов;

desktopBasicOMTS – компонент пользовательского интерфейса роли сотрудника ОМТС – рабочий стол;

panelParts – компонент пользовательского интерфейса роли сотрудника ОМТС – подсистема ведения справочника комплектующих ИТ-ресурсов;

panelRequests – компонент пользовательского интерфейса роли сотрудника ОМТС – подсистема менеджмента заявок на закупку комплектующих по заявкам на обслуживание ИТ-ресурсов.

### **2.3.2 Архитектура программно-информационной системы**

На рисунке 2.6 приведена архитектурная диаграмма классов программно-информационной системы ITSM.System – показаны классы, отражающие предметную (сущностную) область системы. Назначение элементов соответствует одноименным компонентам (см. спецификацию диаграммы на рис. 2.4) [24-26].

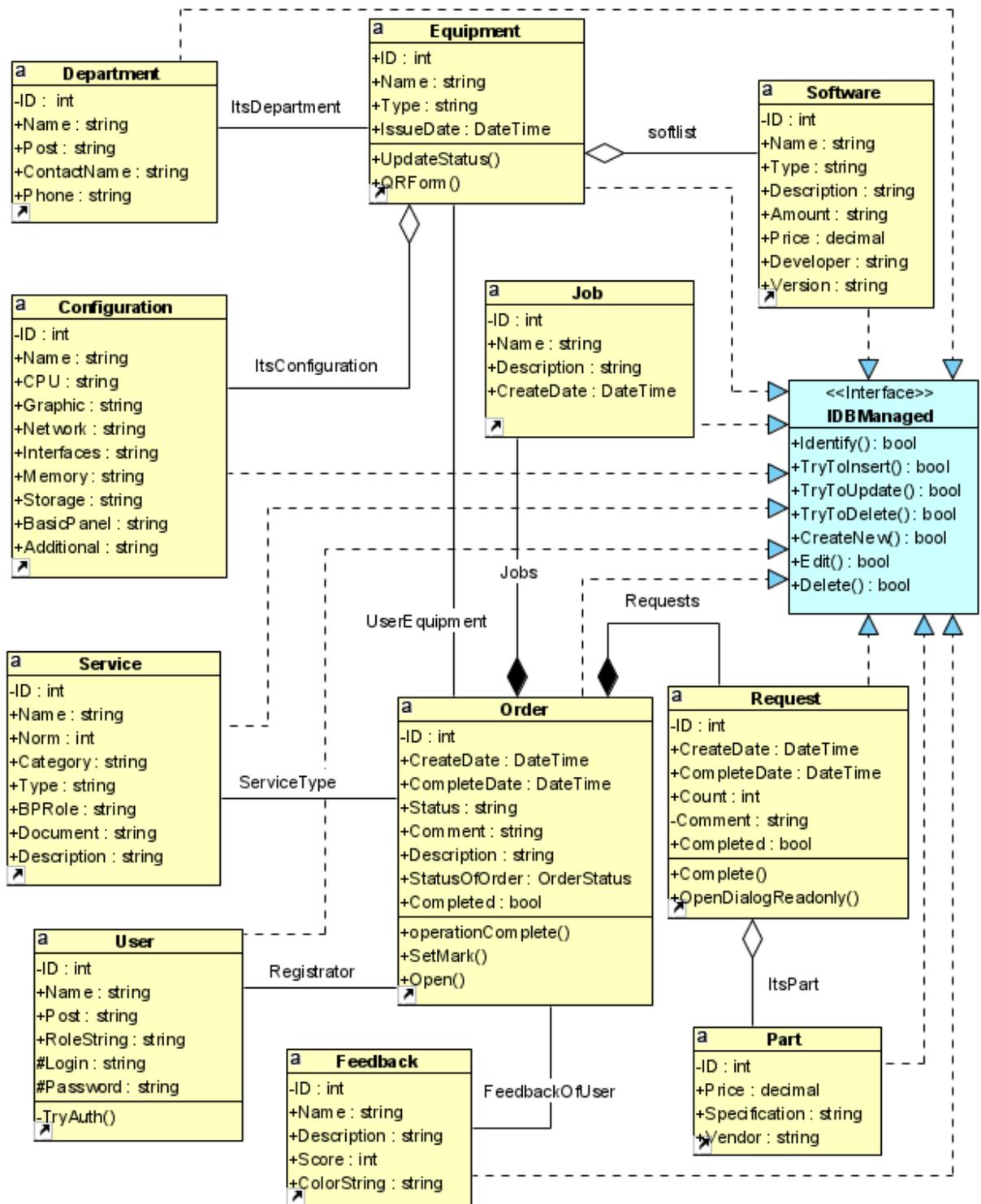


Рисунок 2.6 – UML-диаграмма классов ITSM.System: объектная модель предметной области

Классы-сущности реализуют интерфейс управления данными и взаимодействия с базой данных. При этом в рамках работы системы для всех классов-сущностей можно определить типовое поведение, которое можно

показать на диаграмме состояний обобщенного класса-сущности (см. рисунок 2.7).

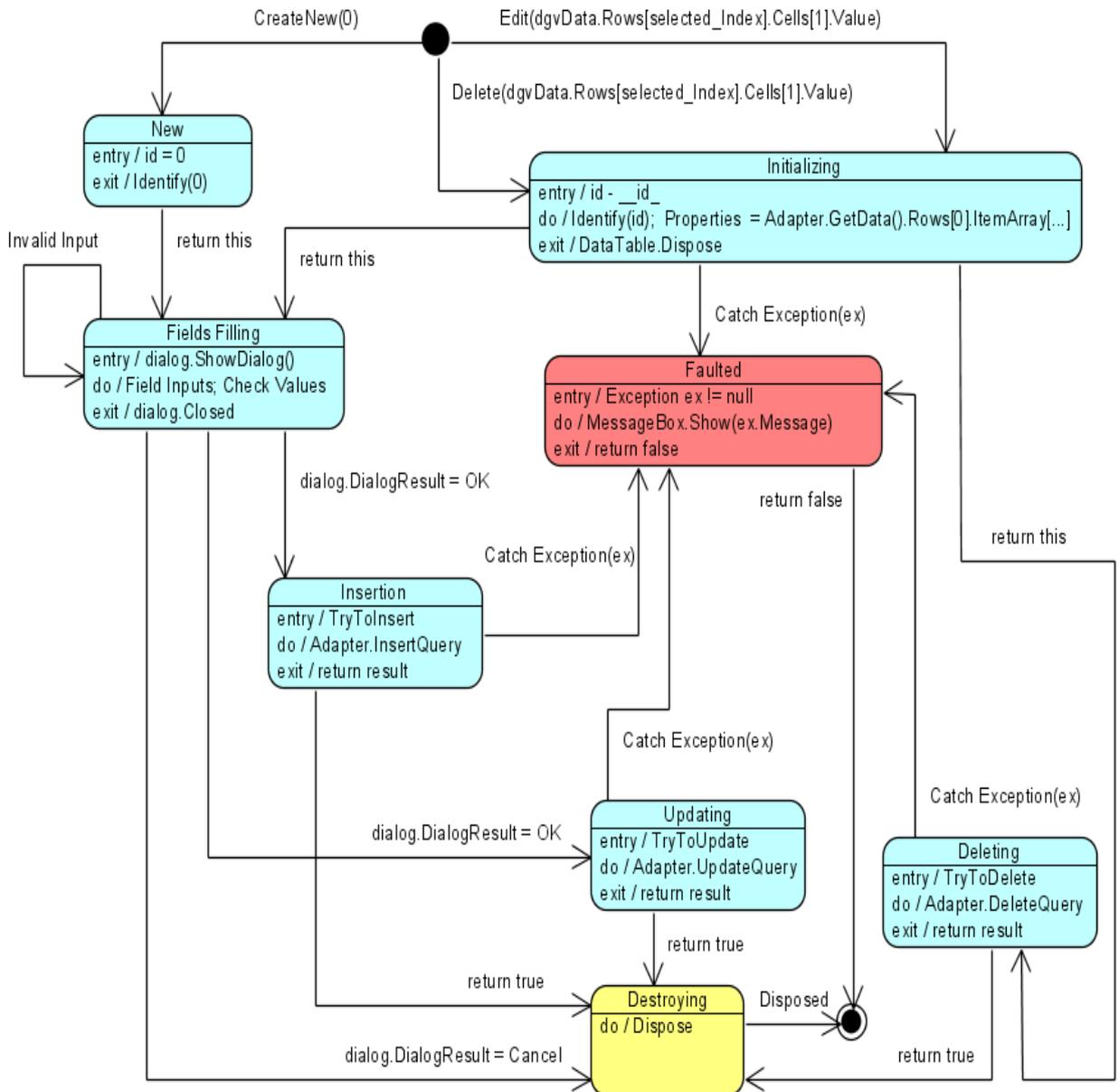


Рисунок 2.7 – UML-диаграмма состояний класса-сущности в ITSM.System

На диаграмме показаны все состояния, в которых пребывает типичный класс-сущность на протяжении всего жизненного цикла в процессе операций управления данными, выполняемых совокупностью методов, реализуемых указанными ранее интерфейсами.

Краткая характеристика состояний, приведенных на диаграмме, описана в таблице 2.12.

Таблица 2.12 – Характеристика состояний класса-сущности

Состояние	Краткое описание	Действия при входе в состояние (entry)	Действия при выходе из состояния (exit)	Действия при пребывании в состоянии (do)
New	Создание экземпляра	Присвоение: ID = null	Нет действия	Обращение к методу инициализации данных из БД или по умолчанию
Initializing	Получение данных	Получение ID экземпляра	Освобождение ресурса подключения к БД (DataSet)	Чтение данных из полей БД в соответствующие поля класса
Fields Filling	Заполнение полей данных	Открыть диалог ввода данных	Закрыть диалог ввода данных	Ввод данных в поля класса посредством GUI
Faulted	Ошибка управления	Обработано исключение	Вернуть в управляющую функцию ошибку ввода	Вывод сообщения с информацией об исключении
Inserting	Добавление в БД	Вызов метода интерфейса вставки записи	Вернуть в управляющую функцию результат операции управления данными	Выполнение команды SQL: Insert-запроса
Updating	Правка в БД	Вызов метода интерфейса обновления записи		Выполнение команды SQL: Update-запроса
Deleting	Удаление из БД	Вызов метода удаления записи		Выполнение команды SQL: Delete-запроса

Продолжение таблицы 2.12

Destroying	Диспозиция экземпляра	Нет действия	Системная память освобождается	Вызов системной операции
------------	-----------------------	--------------	--------------------------------	--------------------------

			(ссылка на объект помечается как неиспользуемая)	Dispose, «Сборка мусора» Garbage Collector.
--	--	--	--	---

### 2.3.3 КТС программно-информационной системы

На рисунке 2.8 представлена UML-диаграмма развертывания ITSM.System, одновременно определяющая ее техническую структуру.

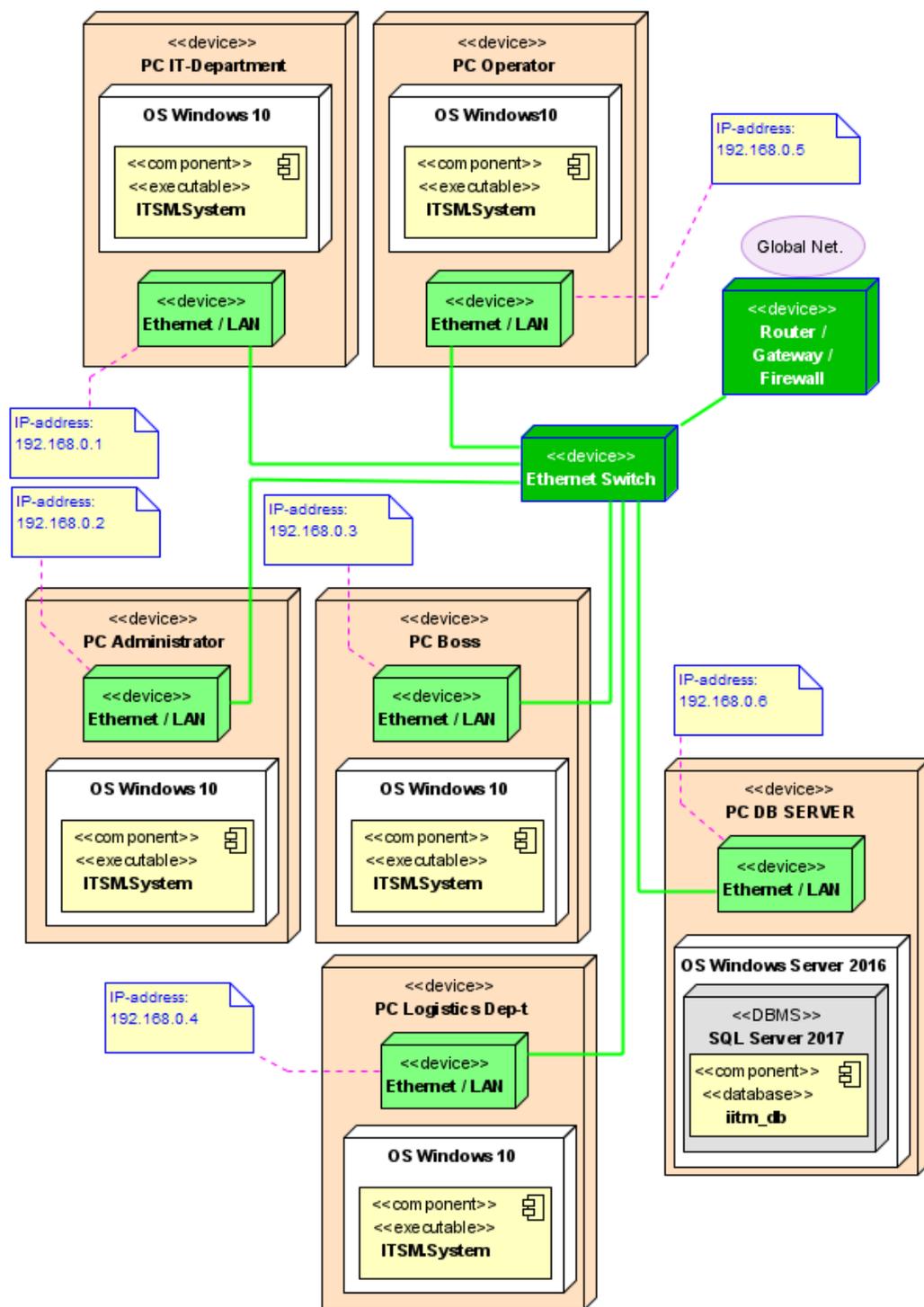


Рисунок 2.8 – UML-диаграмма развертывания ITSM.System

В соответствии с приведенной диаграммой KTC ITSM.System составляют:

автоматизированные рабочие места (АРМ) для всех пользовательских ролей (администратор, ОМТС, ИТ, оператор, руководство) с установленным на них исполняемым файлом приложения ITSM.System;

сервер СУБД MS SQL Server 2017 с развернутой на нем базой данных ITSM.System – iitm\_db;

устройства сетевой инфраструктуры локальной сети, обеспечивающие работу перечисленных ранее компонентов: хабы, свитчи, линии связи и т.д.

На рисунке 2.8 не приведены конкретные модели и спецификации устройств, указаны примерные сетевые адреса, характеристики. Все эти свойства должны быть определены по месту ИТ-подразделением, интегрирующим систему в ИТ-инфраструктуру в соответствии с утверждённой ИТ-политикой. Также может быть изменена топология сети.

## **2.4 Проектирование пользовательского интерфейса программно-информационной системы**

Основная концепция по интерфейсу системы ITSM.System предполагает следующие решения:

- использование единого главного окна системы – MainWindow, в рамках которого в пользовательском контейнере (Panel Container – элемент типа System.Windows.Forms.Panel [27]) будет организовано переключение подсистем: справочников, учета, отчетности, сервисных функций;
- использование отдельных пользовательских панелей (на базе System.Windows.Forms.UserControl) для выполнения сервисных функций, таких, как: настройка подключения к базе данных, авторизация пользователя в системе;
- использование отдельных пользовательских панелей (на базе System.Windows.Forms.UserControl) для ведения справочников – каждый справочник снабжается учетной таблицей, которая обращается в БД через адаптеры посредством связки «Data Source – Data Table – Data Table Adapter», строкой состояния, меню команд управления данными, системой поиска и фильтрации данных (при необходимости);

- использование отдельных пользовательских панелей (на базе `System.Windows.Forms.UserControl`) для предоставления отчетов: отчеты могут быть выполнены в виде таблиц / текста в формате HTML на базе элемента `System.Windows.Forms.WebBrowser` или в виде инфографики на базе `System.Windows.Forms.Charting`;
- для организации диалога с пользователем применены специальные диалоговые окна (формы) модального исполнения, предусматривающие набор полей, определяемый соответствующей таблицей базы данных;
- формы диалогов поддерживают первичный контроль целостности данных, вводимых пользователем, за счет следующих средств: контроль длины вводимых строк в соответствии с ограничениями полей в БД, использование списков для внешних ключевых полей, применение ограничивающих компонентов (типа `System.Windows.Forms.NumericUpDown`) для соблюдения ограничений полей базы данных по числовым значениям.

Таким образом, пользовательский интерфейс системы `ITSM.System` спроектирован в соответствии с модульным принципом, позволяет разграничить функции компонентов, предоставляя пользователю все необходимые элементы в контексте решаемых задач (выполнения функций), информативные отчеты, а также выполняя функцию первичного контроля и верификации вводимых данных. Кроме того, пользовательский интерфейс системы `ITSM.System` спроектирован в соответствии с основными принципами (правилами) Нильсена-Молиха которые обеспечивают эффективность интерфейса:

- организация диалога с пользователем, поддержка наиболее естественного языка общения с пользователем;
- актуализация данных;

- минимизация загрузки пользователя;
- унификация элементов;
- предоставление оперативной обратной связи;
- информативная и удобная навигация по функциям и подсистемам;
- кастомизация компонентов;

минимизация вероятности допущения ошибок пользователем (например, при вводе данных).

На рисунке 2.9 приведена общая схема, показывающая концепцию организации пользовательского интерфейса системы ITSM.System.

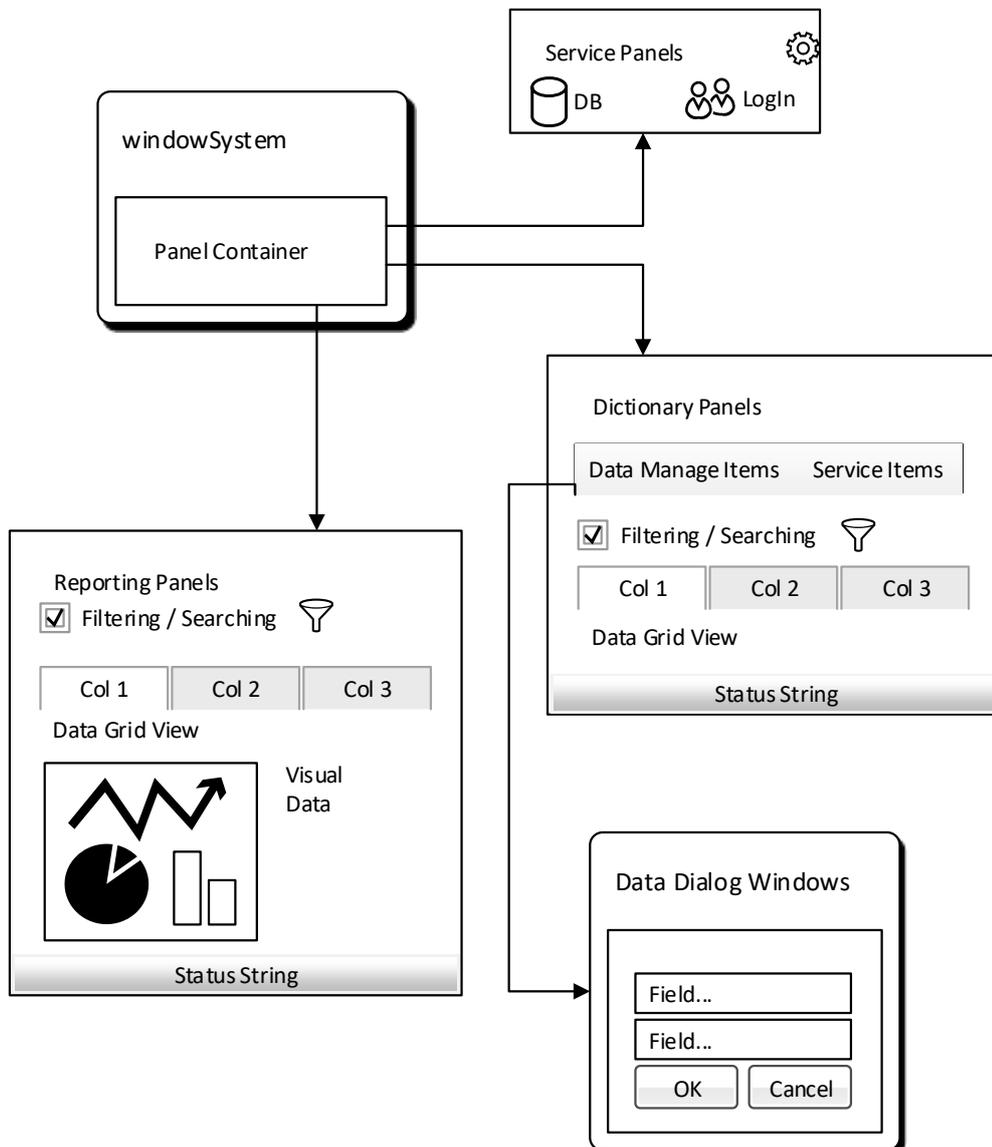


Рисунок 2.9 – Концептуальная схема построения пользовательского интерфейса ITSM.System

На рисунке 2.10 приведен макет формы предоставления отчетных данных по затратам, связанным с закупкой комплектующих для ИТ-компонентов. Этот и другие приведенные макеты разработаны с помощью средств визуального проектирования форм среды разработки MS Visual Studio. Основными элементами данной формы являются:

график затрат, на котором точками на временной линии фиксируются все затраты на приобретение ИТ-комплектующих, а также их размер;

таблица учета затрат на приобретение ИТ-комплектующих с датами, затратами и количеством, а также указанием номера связанной заявки пользователя;

фильтр данных, позволяющий осуществить выборку.

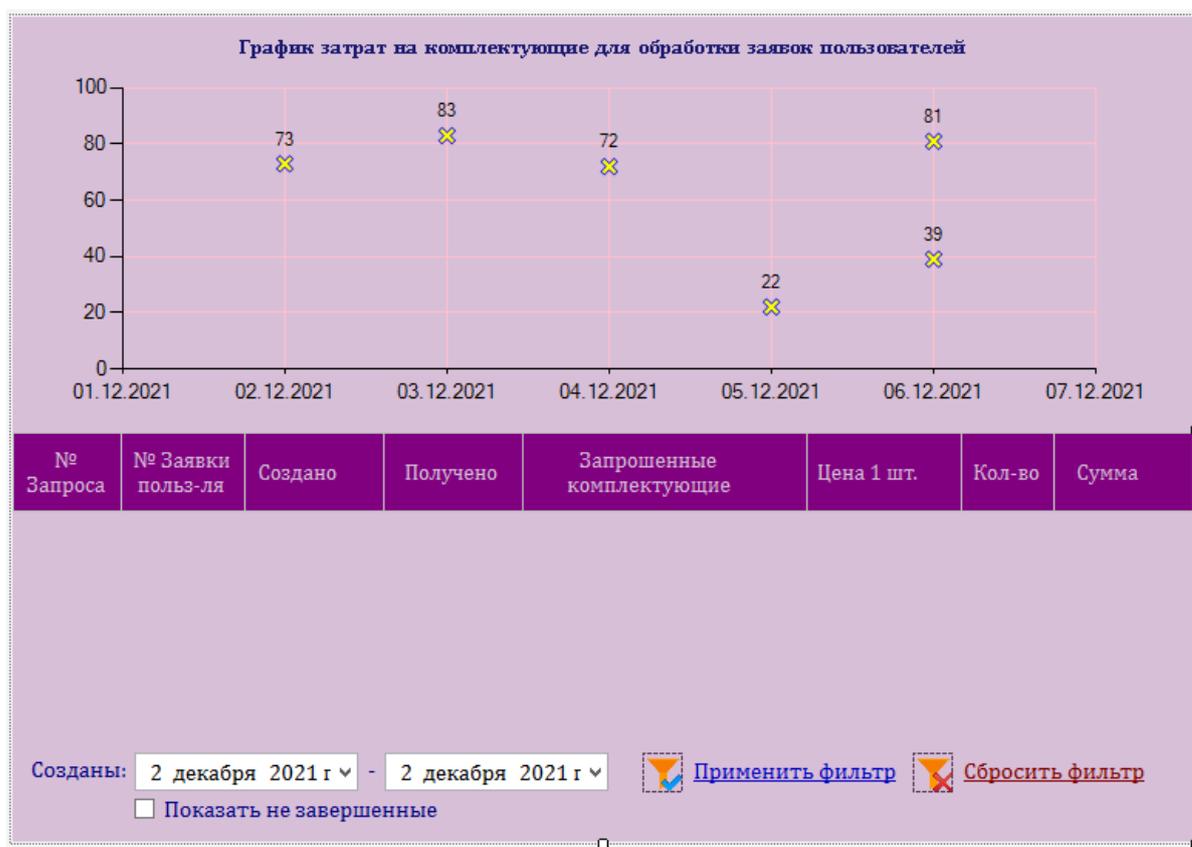


Рисунок 2.10 – Макет формы визуализации графика затрат на комплектующие

На рисунке 2.11 приведен макет формы предоставления отчетных данных по состоянию ИТ-ресурсов компании.

Основными элементами данной формы являются:

диаграмма учета ИТ-ресурсов по подразделениям предприятия с указанием процентного соотношения ИТ в отделе;

таблица учета конфигураций ИТ-ресурсов;

диаграмма отображения общего состояния ИТ-ресурсов предприятия по статусам (ремонт, списание, в работе и т.д.);

полная таблица учета ИТ-ресурсов предприятия.

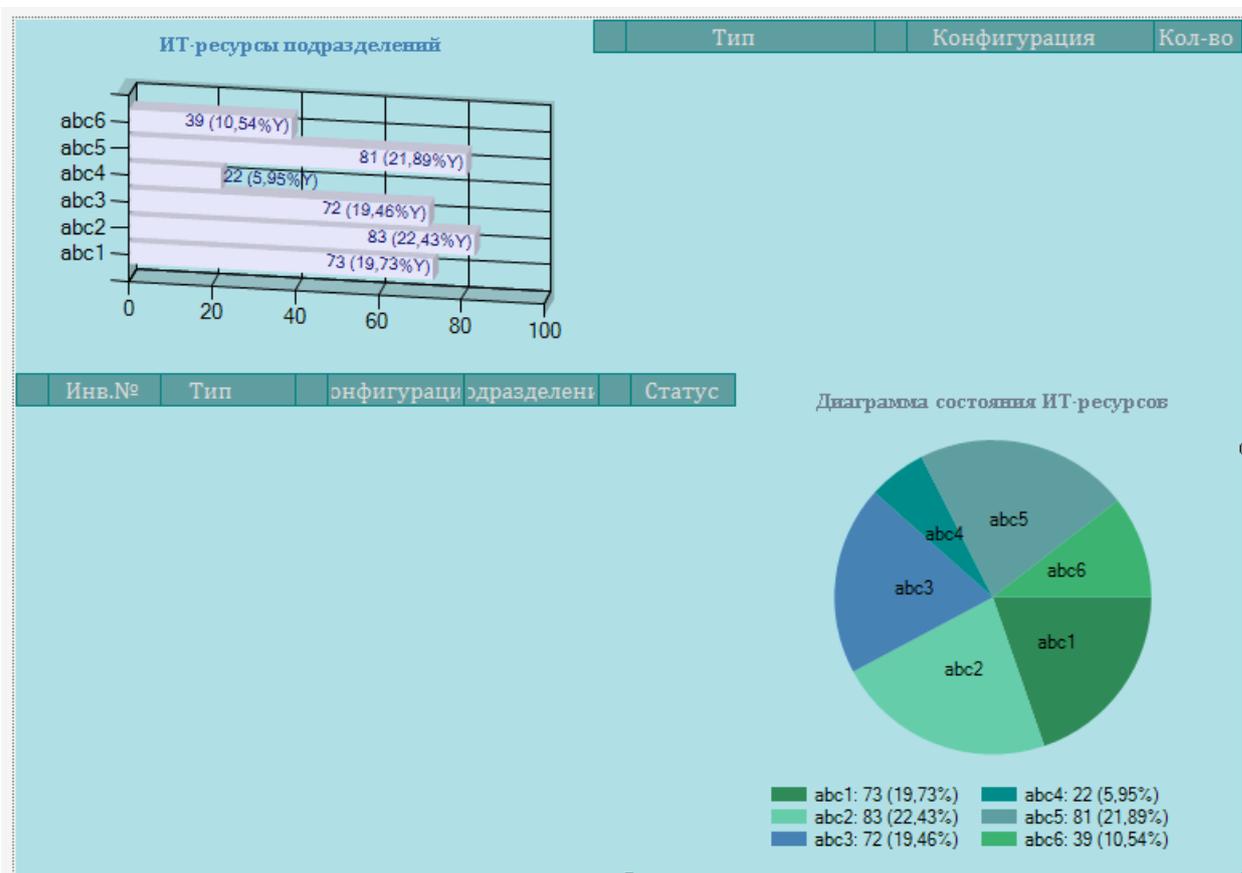


Рисунок 2.11 – Макет формы визуализации состояния ИТ-ресурсов

На рисунке 2.12 приведен макет формы предоставления отчетных данных по сводке заявок пользователей в ИТ-отдел. Основными элементами данной формы являются:

диаграмма «воронка» заявок пользователей с распределением и отображением общего количества заявок пользователей по статусу их выполнения;

таблица учета заявок пользователей со статусами и датами.

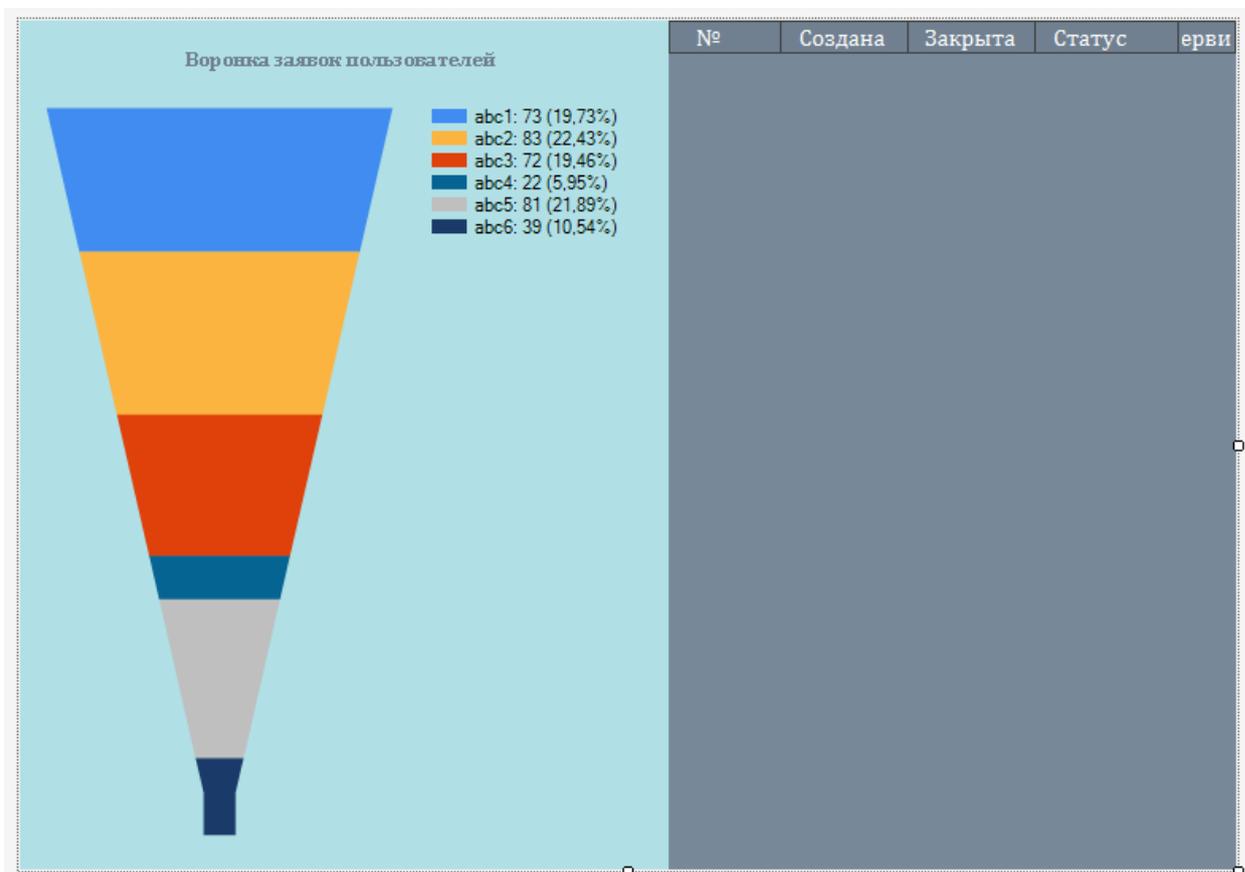


Рисунок 2.12 – Макет формы визуализации воронки заявок в ИТ-отдел

На рисунке 2.13 приведен макет формы предоставления отчетных данных по работе ИТ-подразделения. Основными элементами данной формы являются:

общая сводная таблица, отражающая деятельность ИТ-подразделения – таблица выполнена в виде компонента на базе WebBrowser, наполняемого динамически текстовой строкой в виде форматированного HTML-кода: таблица по каждой заявке содержит ее идентификатор, информацию о дате создания и завершения, статусе (на момент формирования отчета), описанию, краткой характеристике ИТ-ресурса (инвентарный номер, конфигураци., ти,

наименование), оценке качества, количестве выполненных работ и примененных комплектующих, общей сумме затрат на комплектующие;

фильтр данных, позволяющий осуществить выборку данных в таблице за выбранный период;

элементы для формирования документа на печать или сохранения его на дисковое хранилище.

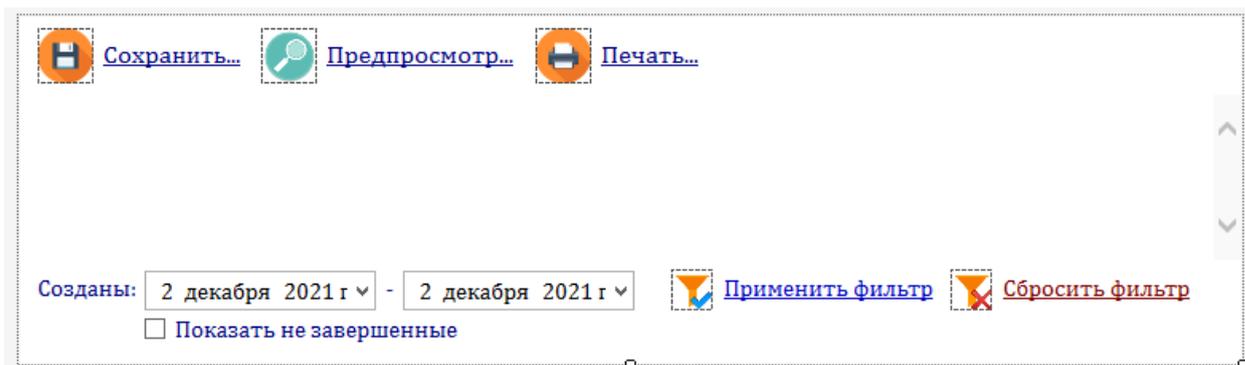


Рисунок 2.13 – Макет формы отчета о деятельности ИТ-отдела

На рисунке 2.14 приведен макет формы предоставления отчетных данных по работе ИТ-подразделения. Основными элементами данной формы являются:

диаграмма распределения оценок качества оказанных ИТ-услуг;

таблица учета заявок с указанием качества предоставленных услуг.

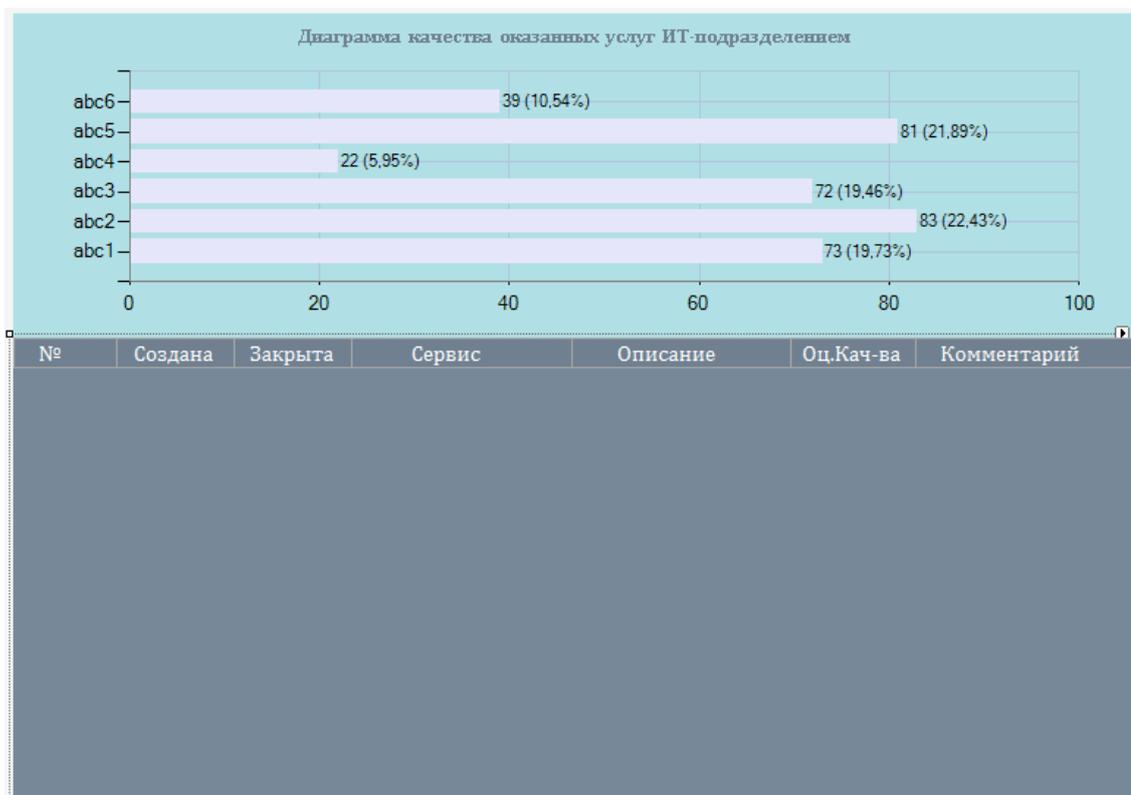


Рисунок 2.14 – Макет формы визуализации данных по качеству предоставления услуг ИТ-отделом

### 3 Рабочий проект

#### 3.1 Спецификация компонентов и классов системы

Основными являются методы, реализующие интерфейс `IDBManaged`. Эти методы отвечают за все операции управления данными как на уровне интерфейса пользователя, так и на уровне манипуляций данными в БД. В таблице 3.1 приведена краткая характеристика этих методов.

Таблица 3.1 – Спецификация методов интерфейса `IDBManaged`

Имя метода (функции)	Область видимости	Назначение
<code>Identify</code>	<code>public</code>	Интерфейс в БД: идентификация объекта по соответствующей записи из таблицы БД.
<code>TryToInsert</code>	<code>public</code>	Интерфейс в БД: вставка в БД данных в соответствии со значениями полей объекта.
<code>TryToUpdate</code>	<code>public</code>	Интерфейс в БД: обновление в БД данных в соответствии со значениями полей объекта.
<code>TryToDelete</code>	<code>public</code>	Интерфейс в БД: удаление из БД записи, соответствующей идентификатору объекта.
<code>CreateNew</code>	<code>public</code>	Интерфейс для пользователя: управление данными, обеспечивающее возможность добавлять новые записи через форму ввода данных (GUI).
<code>Edit</code>	<code>public</code>	Интерфейс для пользователя: управление данными, обеспечивающее возможность редактировать существующие записи через форму ввода данных (GUI).

Продолжение таблицы 3.1

Destroying	Диспозиция экземпляра	Нет действия	Системная память освобождается (ссылка на объект помечается как неиспользуемая)	Вызов системной операции Dispose, «Сборка мусора» Garbage Collector.
------------	-----------------------	--------------	---	--

В таблице 3.2 приведена спецификация дополнительных (кроме описанных в таблице 3.1) методов класса Equipment.

Таблица 3.2 – Спецификация методов класса Equipment

Имя метода (функции)	Область видимости	Назначение
UpdateStatus	public	Вызывает запрос к БД на обновление статуса ИТ-ресурса, представленного экземпляром
QRForm	public	Формирует строку с метаданными ИТ-ресурса (включающими наименование, владельца, дату ввода в эксплуатацию, конфигурацию, инвентарный номер), преобразует эту строку в Bitmap-объект в виде QR-кода (используется метод сторонней библиотеки QRCodeBitmapImage), формирует форму (помещает на нее сформированный текст и изображение QR-кода) и отображает ее в модальном режиме.

В таблице 3.3 приведена спецификация дополнительных (кроме описанных в таблице 3.1) методов класса Order.

Таблица 3.3 – Спецификация методов класса Order

Имя метода (функции)	Область видимости	Назначение
OperationComplete	public	Вызывает операции проверки возможности завершения заявки пользователя в ИТ-отдел (в соответствии с бизнес-правилами управления заявками) и запроса к БД на обновление статуса заявки «Завершено»
SetMark	public	Вызывает в модальном режиме форму внесения отметки качества оказанных услуг текущей заявки (в соответствии со SLA), присваивает назначенную пользователем в диалоге оценку и вызывает в БД соответствующий запрос на обновление данных полей o_feedback_id и o_feedback_comment таблицы TOrder
Open	public	Открывает форму данных текущей заявки, в которой ведется учет выполненных работ и примененных комплектующих

В таблице 3.4 приведена спецификация дополнительных (кроме описанных в таблице 3.1) методов класса Request.

Таблица 3.4 – Спецификация методов класса Request

Имя метода (функции)	Область видимости	Назначение
Complete	public	Вызывает запрос к БД на обновление статуса запроса на комплектующие как выполненного: TRequest. r_date_complete = {DateTime.Now.Date [9]}
OpenDialogReadOnly	public	Открывает форму данных текущего запроса на комплектующие в режиме доступа только для просмотра

В таблице 3.5 приведена спецификация дополнительных (кроме описанных в таблице 3.1) методов класса User.

Таблица 3.5 – Спецификация методов класса User

Имя метода (функции)	Область видимости	Назначение
TryAuth	public	Вызывает запрос к БД с попыткой найти аккаунт, соответствующий указанным логину и паролю. Возвращает null или ID найденного (авторизованного) аккаунта

## 3.2 Тестирование программной системы

### 3.2.1 Модульное тестирование классов и методов программы

В данном разделе приводятся некоторые наиболее важные модульные тесты, разработанные для проверки и тестирования разработанного приложения ITSM.System [28-29].

Поскольку все тесты связаны так или иначе с функциями, затрагивающими методы управления данными в БД, то в проекте ITSM.System необходимо выполнить некоторые подготовительные действия перед запуском модульных тестов. Глобальный класс ссылается на главную форму системы (чтобы отслеживать статусы подсистем и контролировать уровень доступа авторизованных пользователей). Главная же форма в свою очередь в конструкторе пытается установить связь с базой данных и подготовить строку подключения. Модульные тесты тоже в свою очередь пытаются установить связь с БД. Поэтому, чтобы исключить дублирования, необходимо на время запуска модульных тестов закомментировать строки в конструкторе класса MainWindow, инициализирующие соединение с БД (как показано на рисунке 3.1).

```
10 public partial class MainWindow : Form
11 {
12     /// <summary>
13     /// Конструктор класса
14     /// </summary>
15     public MainWindow()
16     {
17         // (Автоматическая функция) - Создание и инициализация UI-компонентов
18         InitializeComponent();
19         // Установить глобальную ссылку на главную форму
20         IITMS.MainForm = this;
21         // Прочитать настройки подключения к БД
22         IITMS.DB_CS = IITMS.LoadConnectionSettings();
23         // Попытка подключиться к БД
24         /*
25          * if (!IITMS.SetConnectionWithDB())
26          * {
27          *     // Показать панель настройки подключения к БД
28          *     IITMS.ShowConnectionSettings();
29          * }
30          * else // Подключение установлено
31          * {
32          *     // Показать панель авторизации пользователя
33          * }
34          */
35     }
```

Рисунок 3.1 – Подготовка класса MainWindow к запускам модульных тестов

Модульный тест проверки кодирования / декодирования данных ИТ-ресурса посредством QR-кода. Код разработанного модульного теста приведен в листинге 3.1.

### Листинг 3.1 – Код модульного теста QRGenerationTM

```
[TestClass]
public class UnitTests
{
    [TestMethod]
    public void QRGenerationTM()
    {
        // Установить соединение с БД
        IITM.IITMS.MainForm = new IITM.MainWindow();
        IITM.IITMS.DB_CS.Set("DBSRVSQL17", "sa", "ZvnV-2020", "iitm-db", "", false);
        IITM.IITMS.SetConnectionWithDB();

        // ID заведомо известный компьютера (инв.№), который есть в БД
        int test_id = 5;
        // Получение данных
        IITM.Database.Entities.Equipment t_equipm
            = new IITM.Database.Entities.Equipment(test_id);

        // Актуальные данные для QR-кода:
        string _EXPECTED_ = t_equipm.GetQRInfoText();

        // Создание QR-кода
        Image qr_image = t_equipm.GetQRImage();
        // Декодирование
        QRCodeDecoder decoder = new QRCodeDecoder();

        string _ACTUAL_ = decoder.decode(new QRCodeBitmapImage((Bitmap)qr_image),
            System.Text.Encoding.UTF8);

        // Проверка теста
        Assert.AreEqual(_EXPECTED_, _ACTUAL_);
    }
}
```

Модульный тест проверки авторизации пользователя. Для теста необходимо указать заведомо известные данные аккаунта пользователя из таблицы аккаунтов (рисунок 3.2).

	u_role_id	u_login	u_password
	3	boss	boss
	2	itm	itm
	1	omts	omts
⌘	0	operator	operator
	4	admin	admin
ресурсов	2	shpak	shpak

Рисунок 3.2 – Фрагмент таблицы аккаунтов БД системы

Код разработанного модульного теста приведен в листинге 3.2.

### Листинг 3.2 – Код модульного теста AuthTestMethod

```
[TestMethod]
public void AuthTestMethod()
{
    // Установить соединение с БД
    IITM.IITMS.MainForm = new IITM.MainWindow();
    IITM.IITMS.DB_CS.Set("DBSRVSQL17", "sa", "ZvnV-2020", "iitm-db", "", false);
    IITM.IITMS.SetConnectionWithDB();

    // Заведомо известный логин/пароль пользователя, который есть в БД
    string login = "admin";
    string password = "admin";

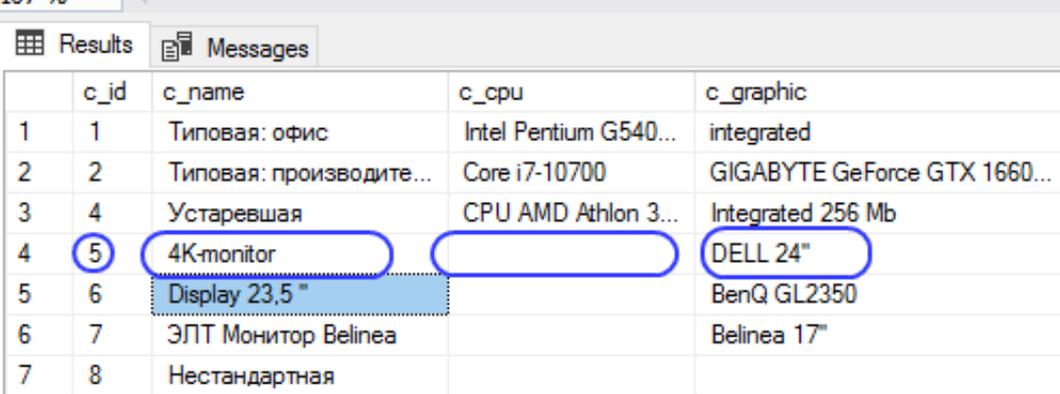
    // Ожидаемое
    bool _EXPECTED_ = true;

    // Авторизация
    IITM.Database.Entities.User t_user = new IITM.Database.Entities.User();
    bool _ACTUAL_ = t_user.TryAuth(login, password);

    // Проверка теста
    Assert.AreEqual(_EXPECTED_, _ACTUAL_);
}
```

Модульный тест проверки методов интерфейса управления данными – на примере класса конфигурации ИТ-ресурса – Configuration.

Для теста необходимо указать заведомо известные данные конфигурации из таблицы конфигураций TConfiguration, в примере взяты данные, соответствующие записи с c\_id = 5 (рисунок 3.3).



	c_id	c_name	c_cpu	c_graphic
1	1	Типовая: офис	Intel Pentium G540...	integrated
2	2	Типовая: производител...	Core i7-10700	GIGABYTE GeForce GTX 1660...
3	4	Устаревшая	CPU AMD Athlon 3...	Integrated 256 Mb
4	5	4K-monitor		DELL 24"
5	6	Display 23,5"		BenQ GL2350
6	7	ЭЛТ Монитор Belinea		Belinea 17"
7	8	Нестандартная		

Рисунок 3.3 – Фрагмент таблицы конфигураций БД системы

Кроме того, для класса сущности, определенного в тесте, необходимо также раскрыть поля, которые будут меняться в рамках теста, для публичного доступа. В данном примере это поля наименования (Name) и характеристики графического адаптера (Graphics). Поля сущности открываются для доступа в

коде соответствующего класса в начальной секции «Доступ к атрибутам пользователя» путем конфигурирования сеттеров соответствующих полей: удаления приватного модификатора доступа, как показано на рисунке 3.4 для данного примера.

```
public class Configuration
{
    // -----
    // Доступ к атрибутам пользователя
    // -----
    /// <summary>
    /// ID
    /// </summary>
    ссылка: 8
    public int ID { get; private set; }
    /// <summary>
    /// Наименование
    /// </summary>
    ссылка: 12
    public string Name { get; private set; }
    /// <summary>
    /// ЦПУ
    /// </summary>
    ссылка: 10
    public string CPU { get; private set; }
    /// <summary>
    /// Графика
    /// </summary>
    ссылка: 12
    public string Graphic { get; private set; }
    /// <summary>
    /// </summary>

```

Рисунок 3.4 – Открытие публичного доступа для изменения полей наименования (Name) и характеристики графического адаптера (Graphics) класса-сущности конфигурации (Configuration)

### Листинг 3.3 – Код модульного теста DBEditRowTestMethod

```
[TestMethod]
public void DBEditRowTestMethod()
{
    // Установить соединение с БД
    IITM.IITMS.MainForm = new IITM.MainWindow();
    IITM.IITMS.DB_CS.Set("DBSRVSQL17", "sa", "ZvnV-2020", "iitm-db", "", false);
    IITM.IITMS.SetConnectionWithDB();

    // Ожидаемое старое
    string _EXPECTED_OLD_VAR_CNAME_FIELD_ = "4K-monitor";
    string _EXPECTED_OLD_VAR_CGRAPHIC_FIELD_ = "DELL 24''";

    // Заведомо известная запись о конфигурации, который есть в БД
    int test_id_cfg = 5;
    IITM.Database.Entities.Configuration t_config
        = new IITM.Database.Entities.Configuration(test_id_cfg);

    // Проверка Идентификации
    Assert.AreEqual(_EXPECTED_OLD_VAR_CNAME_FIELD_, t_config.Name.Trim());
    Assert.AreEqual(_EXPECTED_OLD_VAR_CGRAPHIC_FIELD_, t_config.Graphic.Trim());

    // Ожидаемое новое данных
    string _EXPECTED_NEW_VAR_CNAME_FIELD_ = "4K-monitor scheme";
    string _EXPECTED_NEW_VAR_CGRAPHIC_FIELD_ = "Any 4K-supportable";
    bool _EXPECTED_RESULT_ = true;

    // Обновление данных
    t_config.Name = _EXPECTED_NEW_VAR_CNAME_FIELD_;
    t_config.Graphic = _EXPECTED_NEW_VAR_CGRAPHIC_FIELD_;
    bool _ACTUAL_RESULT_ = t_config.TryToUpdate();

    // Чтение обновленных данных
    t_config = new IITM.Database.Entities.Configuration(test_id_cfg);

    // Проверка Обновления данных
    Assert.AreEqual(_EXPECTED_RESULT_, _ACTUAL_RESULT_);
    Assert.AreEqual(_EXPECTED_NEW_VAR_CNAME_FIELD_, t_config.Name.Trim());
    Assert.AreEqual(_EXPECTED_NEW_VAR_CGRAPHIC_FIELD_, t_config.Graphic.Trim());
}
```

На рисунке 3.5 показаны результаты автоматизированного запуска модульных тестов средствами MS Visual Studio.

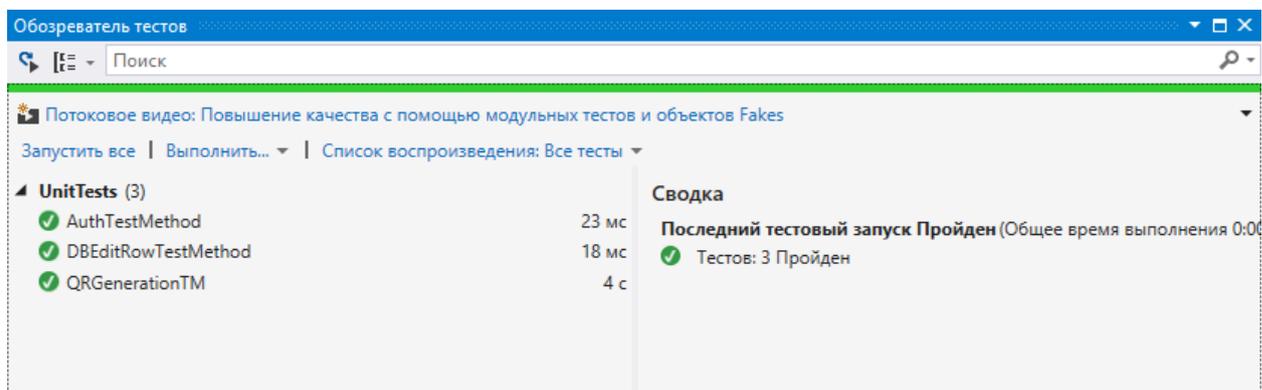


Рисунок 3.5 – Результаты запуска модульных тестов

Как видно на рисунке 3.5 – все модульные тесты выполнены.

Приведенный набор тестов охватывает все важнейшие функции разработанной ITSM.System: управления данными классов-сущностей, авторизации пользователя, формирования инвентарного QR-кода ИТ-ресурса.

### **3.2.2 Системное тестирование программно-информационной системы**

В рамках системного тестирования программно-информационной системы ITSM.System производились запуски разработанной системы: от разных пользовательских ролей вызывались различные функции. Результаты этих тестов запротоколированы в данном разделе в виде снимков экрана (фрагментов) системы с результатами выполнения функций.

Функция: вход в систему.

Условия: соединение с БД установлено.

Результат: панель авторизации пользователя (рисунок 3.6).

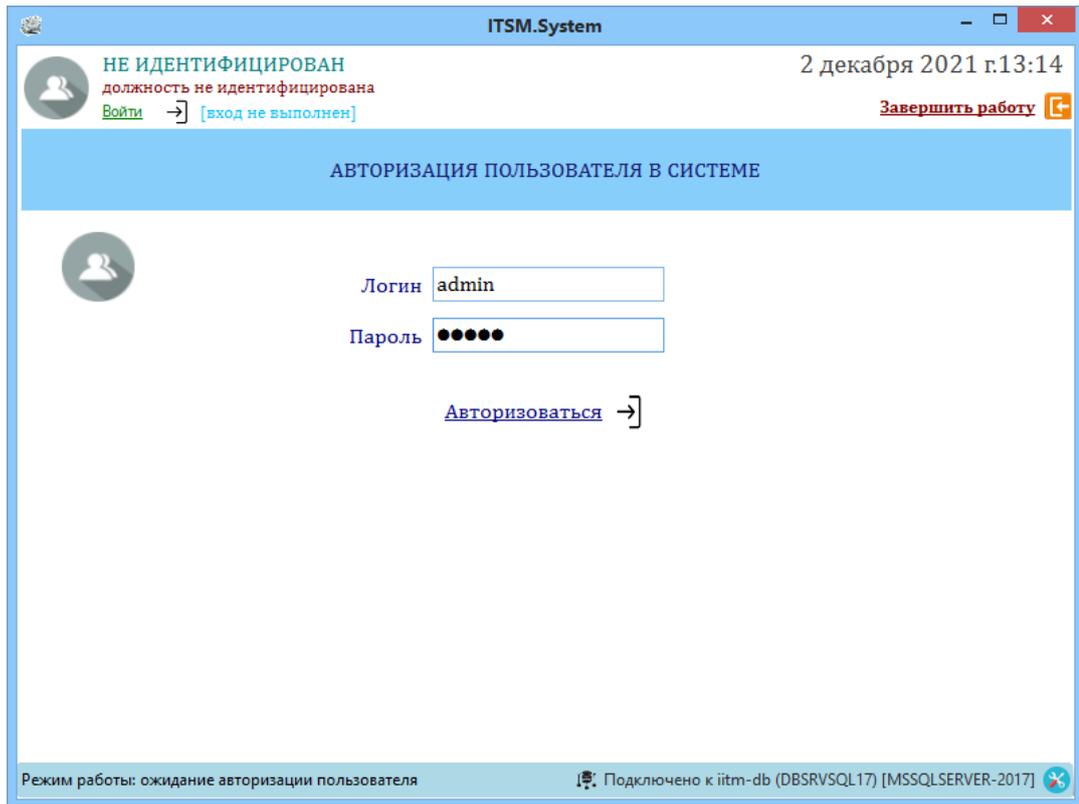


Рисунок 3.6 – Вид панели авторизации пользователя в системе

Функция: регистрация пользователя.

Роль: системный администратор.

Условия: команда регистрации нового пользователя.

Входные данные: данные нового аккаунта пользователя, его роль, учетные авторотационные данные.

Результат: форма ввода данных нового аккаунта (рисунок 3.7).

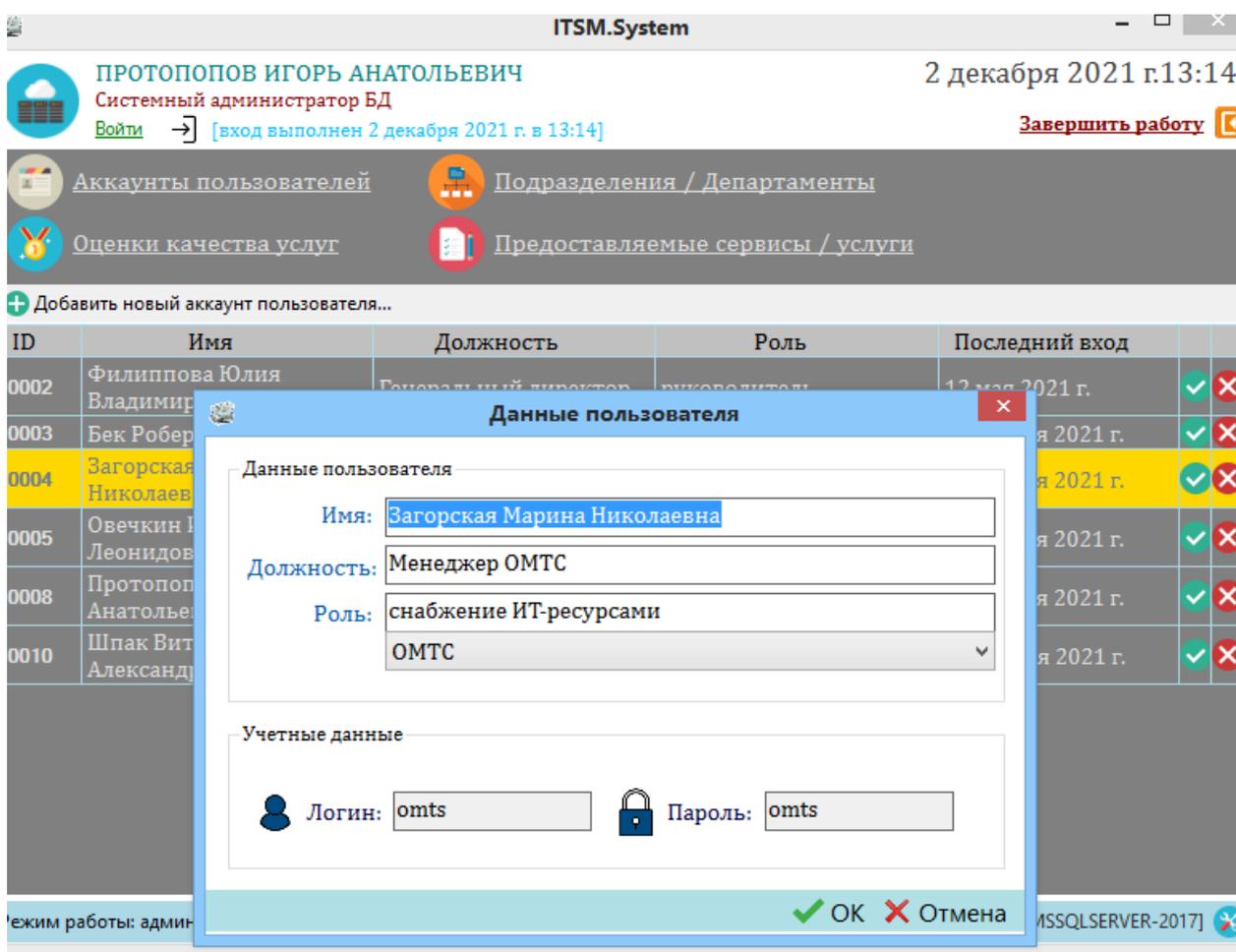


Рисунок 3.7 – Вид подсистемы и диалога регистрации нового аккаунта пользователя в системе

Функция: регистрация нового вида ИТ-услуги.

Роль: системный администратор.

Условия: команда регистрации нового вида ИТ-услуги.

Входные данные: наименование ИТ-сервиса, тип, категория, описание, дополнительная информация о связи с бизнес-процессом, документация SLA, норматив исполнения.

Результат: форма ввода / редактирования данных нового вида ИТ-услуги (рисунок 3.8).

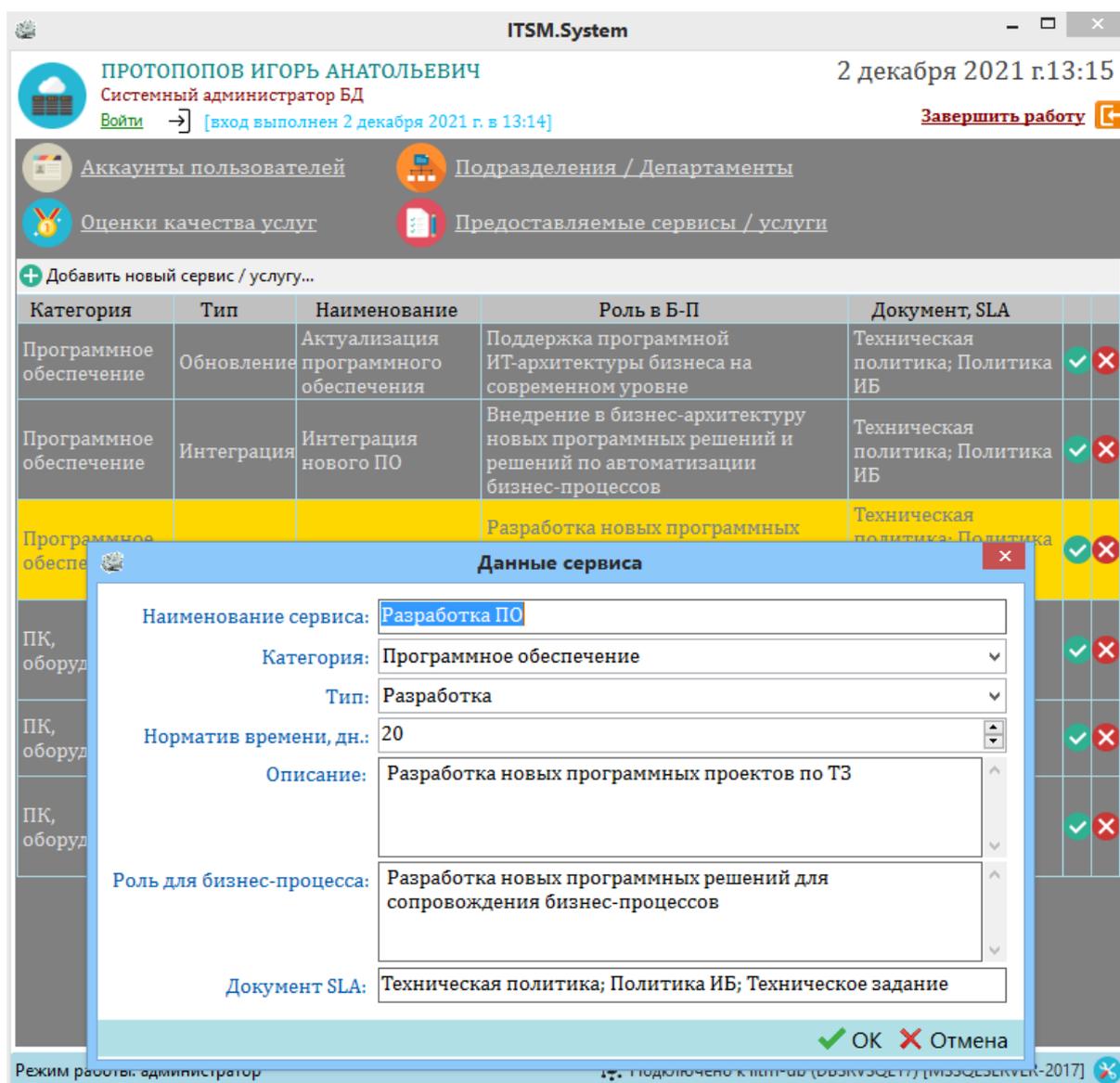


Рисунок 3.8 – Вид подсистемы и диалога регистрации нового вида ИТ-услуги в системе

Функция: просмотр заявок пользователей.

Роль: оператор ИТ-поддержки.

Условия: вход в систему авторизованного пользователя с соответствующей ролью.

Входные данные: нет.

Результат: форма сводной таблицы заявок пользователей, по умолчанию выводится за текущий год (рисунок 3.9).

ITSM.System

ОВЕЧКИН ИГОРЬ ЛЕОНИДОВИЧ  
 Оператор ИТ-поддержки пользователей  
 Войти [вход выполнен 2 декабря 2021 г. в 13:16]

2 декабря 2021 г.13:16  
 Завершить работу

+ Создать новую заявку...

№	Создана	Закрыта	Статус	Описание	Оц.Кач-ва	Сервис
1018	14.04.2021	21.04.2021	Решена	Перестала загружаться ОС	Превосходно	[ПК, оборудование]: Ремонт. Ремонт компьютерной техники
1017	12.04.2021	25.04.2021	Завершена	Сильно шумит системный блок		[ПК, оборудование]: Ремонт. Ремонт компьютерной техники
1016	17.04.2021	23.04.2021	В работе	Нужен дополнительный монитор	Хорошо	[ПК, оборудование]: Расширение. Введение новых единиц оборудования
1015	05.04.2021	09.04.2021	Решена	Разработка просмотрщика смет в форматах ".smt"	Превосходно	[Программное обеспечение]: Разработка. Разработка ПО
1014	25.04.2021	25.04.2021	Решена	Установить программу "Консультант"	Превосходно	[Программное обеспечение]: Интеграция. Интеграция нового ПО
1013	03.04.2021	15.04.2021	Решена	Необходим калькулятор стоимости смет по ТЗ № 43 от 02.11.2020 г.	Минимум	[Программное обеспечение]: Разработка. Разработка ПО
1012	10.04.2021	13.04.2021	Решена	Установка нового программного обеспечения Enterprise Architect версии 15.0 для расширения спектра проектируемых моделей	Отлично	[Программное обеспечение]: Интеграция. Интеграция нового ПО

Созданы: 1 января 2021 г - 31 декабря 2021 г  
 Показать завершенные

Применить фильтр Сбросить фильтр

Режим работы: поддержка пользователей Подключено к itm-db (DBSRVSQL17) [MSSQLSERVER-2017]

Рисунок 3.9 – Форма сводной таблицы заявок пользователей

Функция: ввод данных по заявке пользователя.

Роль: ИТ-отдел.

Условия: открытие активной заявки пользователя для ввода данных по этой заявке.

Входные данные: данные о произведенных работах, список необходимых комплектующих для выполнения заявки, для составления соответствующего запроса в ОМТС.

Результат: форма подробных сведений о выполненных работах по заявке (рисунок 3.10).

Данные заявки пользователя					
Создана: 14 апреля 2021 г.					
Зарегистрировал: Овечкин Игорь Леонидович / Оператор ИТ-поддержки пользователей					
Категория сервисов: [ПК, оборудование]: Ремонт. Ремонт компьютерной техники					
Ссылка на оборудование: Инв.№0010: Персональный компьютер (Типовая: производительный) // Технологический отдел, введен с 11 апреля					
Выполненные работы по заявке пользователя: + Добавить...					
Дата	Наименование	Описание			
✓ 15.04.2021	Диагностика неисправности				✗
✓ 20.04.2021	Замена HDD				✗
✓ 21.04.2021	Установка ОС	Windows 10			✗
Запросы в ОМТС на дополнительные комплектующие для решения заявки: + Добавить...					
№ Запроса	Создано	Получено	Запрошенные комплектующие	Кол-во	Цена 1 шт.
0020	⚙ 15.04.2021	19.04.2021	[WD] Жесткий диск: HDD 3.5" 2TB SATA-III	1	5 200,00 Р

Рисунок 3.10 – Форма вывода подробных данных по заявке

Функция: учет конфигураций.

Роль: ИТ-отдел.

Условия: вход в подсистему учета конфигураций.

Входные данные: нет.

Результат: форма учета конфигураций (рисунок 3.11).

Наименование	ЦП	Графика	Память	Сеть	Интерфейс	Данные	База		
Типовая: офис	Intel Pentium G5400 LGA1151v2 BOX	integrated	2x AMD DDR3 2Gb 2666MHz	integrated LAN 100 Mbps	2x HDMI, DVI, VGA, 2x DP, COM, USB concentrator 6 ports	HDD 500 Gb Seagate 3.5'' SSD	Gigabyte B450 I Aorus Pro WIFI	✓	✗
Типовая: производитель	Core i7-10700	GIGABYTE GeForce GTX 1660 SUPER MINI ITX OC	4x Kingston HyperX FURY Black [HX436C17FB 4096 MB	Realtek LAN 1000 Mbps	2x HDMI, DVI, VGA, 2x DP, COM, USB concentrator 6 ports	Toshiba P300 Burst 2TB SSD	ASRock A320M-DVS R4.0	✓	✗
Устаревшая	CPU AMD Athlon 3800+ 2000 MHz	Integrated 256 Mb	2 x 1024 (A-Data DDR-II) 12-14-14-12	Integrated LAN 100 Mbps	4x USB, PS/2, VGA, DVI	HDD 140 GB Transcend	MSI	✓	✗
4K-monitor scheme		Any 4K-supportabl			DP, HDMI, 2x USB			✓	✗
Display 23,5 ''		BenQ GL2350			VGA, DVI, HDMI			✓	✗
ЭЛТ Монитор Belinea		Belinea 17''			VGA			✓	✗
Нестандартна:								✓	✗

Режим работы: ИТ-управление Подключено к iitm-db (DBSRVSQL17) [MSSQLSERVER-2017]

Рисунок 3.11 – Форма учета конфигураций

Функция: формирование инвентарного QR-кода для выбранного ИТ-ресурса.

Роль: ИТ-отдел.

Условия: выбор требуемого ИТ-ресурса.

Входные данные: ID требуемого ИТ-ресурса.

Результат: форма данных инвентарного QR-кода с изображением (доступным для сохранения) и его текстовой расшифровкой, включающей: инвентарный номер, наименование, подразделение, состояние, техническую спецификацию (рисунок 3.12).



Рисунок 3.12 – Форма данных инвентарного QR-кода ИТ-ресурса

Функция: регистрация нового вида комплектующего, применяемого для ИТ-ресурсов.

Роль: менеджер ОМТС.

Условия: команда регистрации нового вида комплектующего, применяемого для ИТ-ресурсов.

Входные данные: наименование, техническая спецификация, производитель, номинальная стоимость.

Результат: форма ввода / редактирования данных нового вида комплектующего, применяемого для ИТ-ресурсов (рисунок 3.13).

ITSM.System 2 декабря 2021 г.13:17

ЗАГОРСКАЯ МАРИНА НИКОЛАЕВНА  
 Менеджер ОМТС  
 Войти → [вход выполнен 2 декабря 2021 г. в 13:17] [Завершить работу](#)

+ Добавить новые комплектующие...

Наименование	Характеристики	Производитель	Стоимость		
Модуль RAM	11-11-11-11, 2048 MB	Kingston	1 080,00 Р	✓	✗
Модуль RAM	10-11-10-9, 4096 MB DDR III	Transcend	2 320,00 Р	✓	✗
Модуль RAM	11-11-10-10, 4096 MB DDR III	Kingston	3 000,00 Р	✓	✗
Жесткий диск	HDD 3.5" 1TB SATA-III	WD	4 200,00 Р	✓	✗
Жесткий диск	HDD 3.5" 2TB SATA-III	WD	5 200,00 Р	✓	✗
Жесткий диск	HDD 3.5" 1TB SATA-III	Seagate	4 350,00 Р	✓	✗
Жесткий диск	HDD 3.5" 2TB SATA-III	Seagate	5 380,00 Р	✓	✗
Кабель/Контроллер SATA-III	l=20mm.		280,00 Р	✓	✗
Аетевой адаптер TP-Link 004D-6Y	1000 Mbps	TP-Link	368,00 Р	✓	✗
Патч-корд	STP, 2.5 m., Green RJ45 - RI45	A-Data	160,00 Р	✓	✗
			112,00 Р	✓	✗
			348,00 Р	✓	✗
			95,00 Р	✓	✗
			13 450,00 Р	✓	✗
			9 500,00 Р	✓	✗
			470,00 Р	✓	✗

**Данные комплектующего**

Наименование: Штекер DVI

Спецификация: female

Производитель:

Стоимость по справочнику: 95

OK Отмена

Режим работы: снабжение ИТ-ресурсами Подключено к ittm-db (DBSRVSQL17) [MSSQLSERVER-2017]

Рисунок 3.13 – Вид подсистемы и диалога регистрации нового вида комплектующего, применяемого для ИТ-ресурсов

Функция: просмотр запросов на закупку комплектующих, полученных от ИТ-отдела.

Роль: ОМТС.

Условия: вход в систему авторизованного пользователя с соответствующей ролью.

Входные данные: нет.

Результат: форма сводной таблицы запросов на закупку комплектующих, полученных от ИТ-отдела, по умолчанию выводится за текущий год (рисунок 3.14).

ITSM.System

ЗАГОРСКАЯ МАРИНА НИКОЛАЕВНА  
Менеджер ОМТС  
Войти →] [вход выполнен 2 декабря 2021 г. в 13:17] 2 декабря 2021 г.13:19  
Завершить работу

	№ Запроса	№ Заявки поль-ля	Создано	Получено	Запрошенные комплектующие	Кол-во	Цена 1 шт.
Запросы комплектующих Справочник Parts	0001	0003	17.04.2021		[WD] Жесткий диск: HDD 3.5" 1TB SATA-III	1	4 200,00 Р
	0002	0003	17.04.2021	17.04.2021	[Кабель/Контроллер SATA-III: l=20mm.	2	280,00 Р
	0004	1003	18.04.2021	18.04.2021	[Kingston] Модуль RAM: 11-11-10-10, 4096 MB DDR III	2	3 000,00 Р
	0005	1003	18.04.2021	18.04.2021	[WD] Жесткий диск: HDD 3.5" 1TB SATA-III	1	4 200,00 Р
	0009	1007	25.04.2021	25.04.2021	[Transcend] Модуль RAM: 10-11-10-9, 4096 MB DDR III	1	2 320,00 Р
	0010	1010	25.04.2021		[Seagate] Жесткий диск: HDD 3.5" 2TB SATA-III	1	5 380,00 Р
	0011	1009	22.04.2021	23.04.2021	[TP-Link] Аетевой адаптер TP-Link 004D-6Y: 1000 Mbps	1	368,00 Р
	0012	1009	22.04.2021	22.04.2021	[A-Data] Патч-корд: STP, 2.5 m., Green RJ45 - RJ45	2	160,00 Р
	0013	1008	04.04.2021	04.04.2021	[TP-Link] Аетевой адаптер TP-Link 004D-6Y: 1000 Mbps	1	368,00 Р
	0014	1011	06.04.2021	06.04.2021	[Шнур питания 200 В + PE: 1,5 м.	1	112,00 Р
	0015	1011	06.04.2021	07.04.2021	[Блок питания, mini: 160 W	1	348,00 Р
	0016	1011	06.04.2021	08.04.2021	[Штекер DVI: female	1	95,00 Р
	0017	1012	10.04.2021	12.04.2021	[Sparx Systems] Лицензия на ПО: Enterprise Architect 15.0, single, permanent	3	13 450,00 Р
	0018	1016	17.04.2021	23.04.2021	[Philips] Монитор: 23" 16:9, LED, HD	1	9 500,00 Р

Созданы: 1 января 2021 г - 31 декабря 2021 г  Показать завершенные

Применить фильтр Сбросить фильтр

Режим работы: снабжение ИТ-ресурсами Подключено к iitm-db (DBSRVSQL17) [MSSQLSERVER-2017]

Рисунок 3.14 – Форма сводной таблицы запросов на закупку комплектующих, полученных от ИТ-отдела

Функция: формирование отчетных данных по сводке заявок пользователей в ИТ-отдел.

Роль: руководитель.

Условия: вход в систему авторизованного пользователя с соответствующей ролью, выбор данного вида отчетов.

Входные данные: нет.

Результат: форма, содержащая диаграмму «воронка» заявок пользователей с распределением и отображением общего количества заявок пользователей по статусу их выполнения, а также таблицу учета заявок пользователей со статусами и датами (рисунок 3.15).



Рисунок 3.15 – Вид формы визуализации воронки заявок в ИТ-отдел

Функция: формирование данных по качеству работы ИТ-отдела.

Роль: руководитель.

Условия: вход в систему авторизованного пользователя с соответствующей ролью, выбор данного вида отчетов.

Входные данные: нет.

Результат: форма, содержащая диаграмму распределения оценок качества оказанных ИТ-услуг и таблицу учета заявок с указанием качества предоставленных услуг (рисунок 3.16).

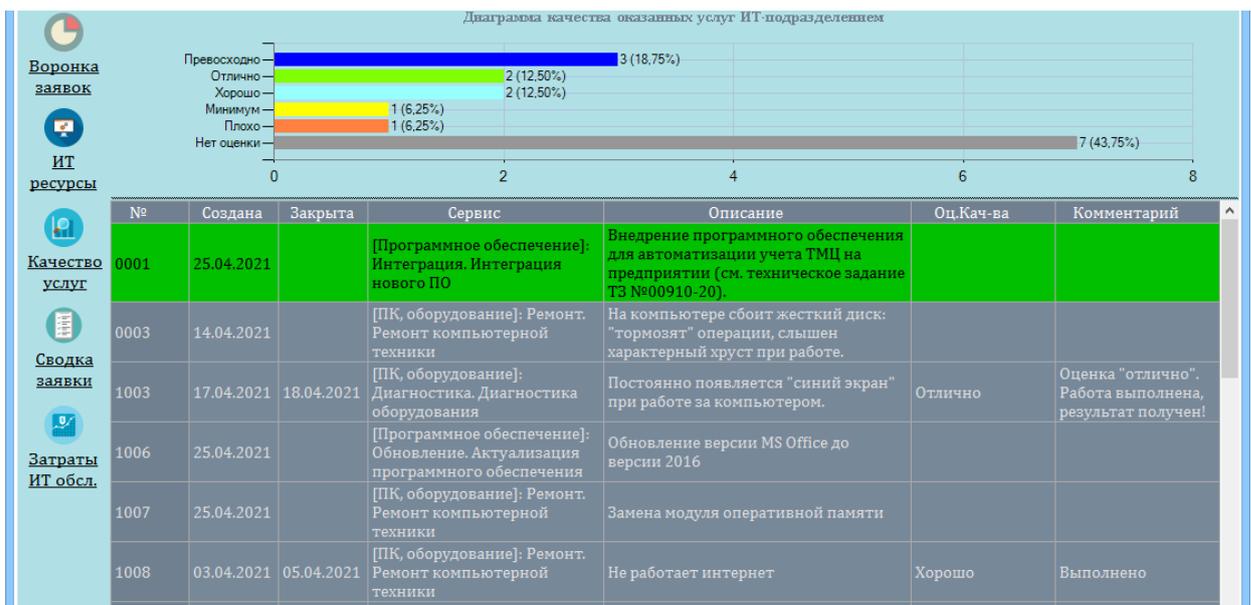


Рисунок 3.16 – Вид формы визуализации данных по качеству предоставления услуг ИТ-отделом

Функция: формирование отчетных данных по состоянию ИТ-ресурсов компании.

Роль: руководитель.

Условия: вход в систему авторизованного пользователя с соответствующей ролью, выбор данного вида отчетов.

Входные данные: нет.

Результат: форма, содержащая диаграмму учета ИТ-ресурсов по подразделениям предприятия с указанием процентного соотношения ИТ в отделе, таблицу учета конфигураций ИТ-ресурсов, диаграмму отображения общего состояния ИТ-ресурсов предприятия по статусам (ремонт, списание, в работе и т.д.), таблицу учета ИТ-ресурсов предприятия. (рисунок 3.17).

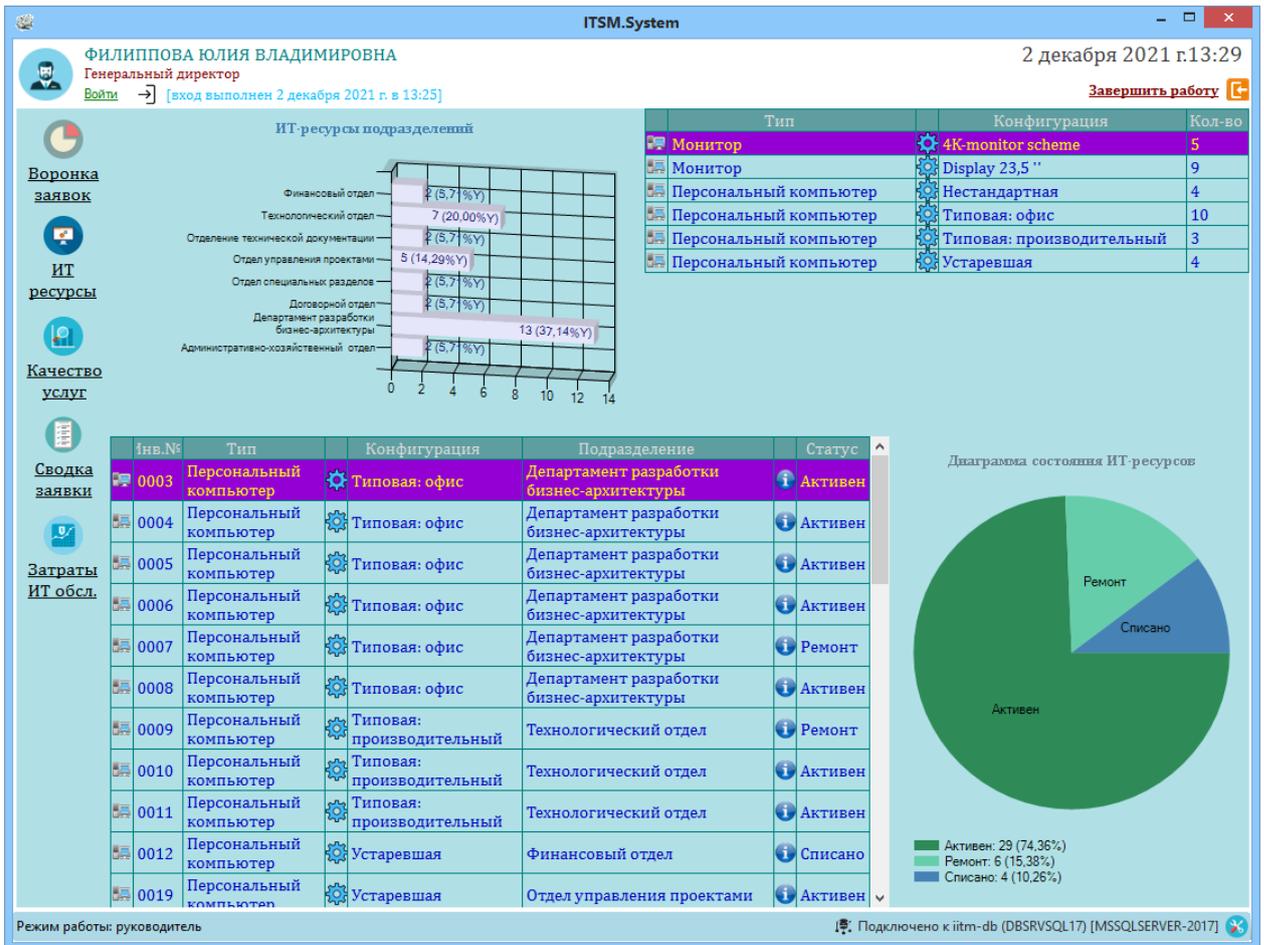


Рисунок 3.17 – Вид формы визуализации состояния ИТ-ресурсов

Функция: формирование отчетных данных по выполнению заявок ИТ-подразделением.

Роль: руководитель.

Условия: вход в систему авторизованного пользователя с соответствующей ролью, выбор данного вида отчетов.

Входные данные: нет.

Результат: форма, содержащая общую сводную таблицу, отражающую деятельность ИТ-подразделения. (рисунок 3.18).

ИТSM.System 2 декабря 2021 г.13:30

ФИЛИПОВА ЮЛИЯ ВЛАДИМИРОВНА  
Генеральный директор  
Войти → [вход выполнен 2 декабря 2021 г. в 13:25] [Завершить работу](#)

[Сохранить...](#) [Предпросмотр...](#) [Печать...](#)

### ОТЧЕТ О ВЫПОЛНЕНИИ ЗАЯВОК ПОЛЬЗОВАТЕЛЕЙ

за период с 1 января 2021 г по 31 декабря 2021 г

№ п./п.	№ Заявки	Дата создания/завершения	Сервис	Описание	Оценка	Кол-во работ	Кол-во комплектов	Сумма, р.
1	0001	25.04.2021/-	[Программное обеспечение]: Интеграция. Интеграция нового ПО	Внедрение программного обеспечения для автоматизации учета ТМЦ на предприятии (см. техническое задание ТЗ №00910-20).	...	0	0	0
2	0003	14.04.2021/-	[ПК, оборудование]: Ремонт. Ремонт компьютерной техники	На компьютере сбоят жесткий диск: "тормозят" операции, слышен характерный хруст при работе.	...	1	2	4760
3	1003	17.04.2021/ 18.04.2021	[ПК, оборудование]: Диагностика. Диагностика оборудования	Постоянно появляется "синий экран" при работе за компьютером.	Отлично	3	2	10200
4	1006	25.04.2021/-	[Программное обеспечение]: Обновление. Актуализация программного обеспечения	Обновление версии MS Office до версии 2016	...	2	0	0
5	1007	25.04.2021/-	[ПК, оборудование]: Ремонт. Ремонт компьютерной техники	Замена модуля оперативной памяти	...	0	1	2320
6	1008	03.04.2021/ 05.04.2021	[ПК, оборудование]: Ремонт. Ремонт компьютерной техники	Не работает интернет	Хорошо	2	1	368
7	1009	21.04.2021/-	[ПК, оборудование]: Расширение. Введение новых единиц оборудования	Дооснащение второй сетевой картой	...	0	2	688
8	1010	15.04.2021/-	[ПК, оборудование]: Расширение. Введение новых единиц оборудования	Установка дополнительного жесткого диска	...	0	1	5380
9	1011	06.04.2021/ 19.04.2021	[ПК, оборудование]: Ремонт. Ремонт компьютерной техники	Постоянно отключается питание	Плохо	4	3	555
10	1012	10.04.2021/ 13.04.2021	[Программное обеспечение]: Интеграция. Интеграция нового ПО	Установка нового программного обеспечения Enterprise Architect версии 15.0 для расширения спектра проектируемых моделей	Отлично	1	1	40350
...	...	03.04.2021/...	[Программное обеспечение]:	Необходим калькулятор стоимости	...	...	...	...

Созданы: 1 января 2021 г - 31 декабря 2021 г [Применить фильтр](#) [Сбросить фильтр](#)  
 Показать не завершенные

Режим работы: руководитель Подключено к иitm-db (DBSRVSQL17) [MSSQLSERVER-2017]

Рисунок 3.18 – Вид формы отчета о деятельности ИТ-отдела

Функция: формирование отчетных данных по затратам, связанным с закупкой комплектующих для ИТ-компонентов.

Роль: руководитель.

Условия: вход в систему авторизованного пользователя с соответствующей ролью, выбор данного вида отчетов.

Входные данные: нет.

Результат: форма, содержащая таблицу учета затрат на приобретение ИТ-комплектующих с датами, затратами и количеством, а также указанием номера связанной заявки пользователя. (рисунок 3.19).

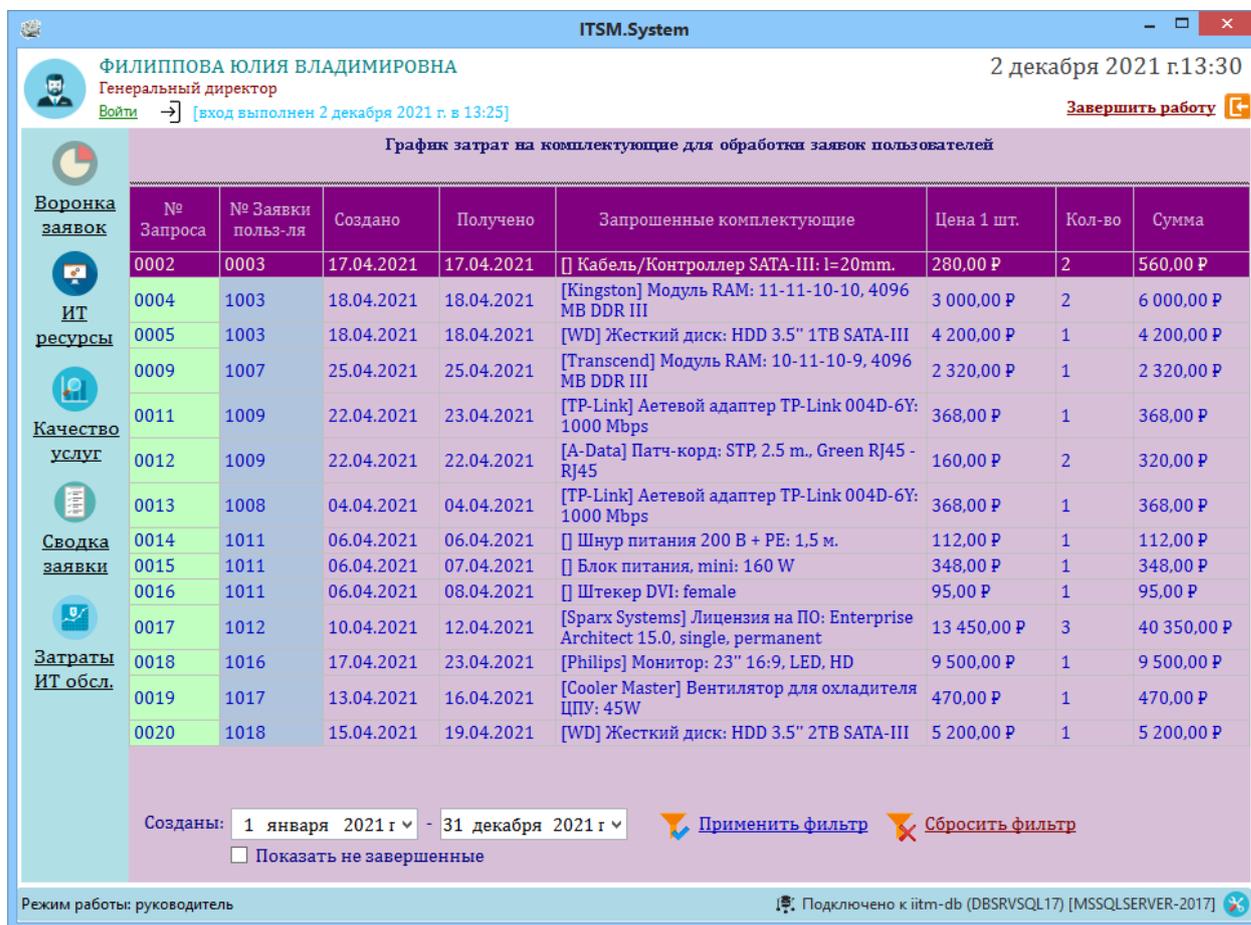


Рисунок 3.19 – Вид формы визуализации графика затрат на комплектующие

Таким образом, в рамках системного тестирования были выполнены и проверены все функции информационной системы ITSM.System, заявленные в пользовательских (функциональных) требованиях в техническом задании на разработку системы.

### 3.3 Сборка компонентов программно-информационной системы

Программные компоненты представляют собой файлы исходных кодов программно-информационной системы [30].

Компиляция и сборка приложения осуществляется в Visual Studio. Исполняемый exe-файл может быть запущен без предварительной установки.

Интерпретация исходных кодов на языке C# будет осуществляться вложенным в корневой каталог интерпретатором.

Все программные компоненты должны быть размещены непосредственно в корневом каталоге системы [30].

## ЗАКЛЮЧЕНИЕ

Согласно методологии ITSM ИТ-подразделение любого предприятия рассматривается неразрывно от всей остальной бизнес-структуры предприятия. Так, ИТ-подразделения в соответствии с современными требованиями бизнеса должны интегрироваться в бизнес-структуру предприятия, а понятие ИТ-услуга должно стать частью общих внутренних бизнес-процессов. Для этого в рамках предприятия должна быть организована информационная система, объединяющая в себе клиентскую подсистему, понятия ИТ-услуги, заявки, соглашения по качеству, а также систему визуализации структуры ИТ-сегмента в разрезе общих бизнес-процессов. В ходе настоящей работы была выполнена разработка такой системы – информационной системы поддержки управления ИТ-услугами на предприятии (обозначение системы: ITMS.System).

В ходе работы была решена следующая задача: разработана программно-информационная система управления ИТ-услугами на основе концепции ITSM.

Готовый рабочий проект представлен:

проектом исходного кода программного обеспечения ITSM.System для IDE Visual Studio 2019;

файлом дампа базы данных с подготовленными тестовыми данными для SQL Server 2018 (\*.bacpac-файл).

Рабочий проект был подвержен тестированию:

набор разработанных модульных тестов охватывает все важнейшие функции разработанной ITSM.System: управления данными классов-сущностей, авторизации пользователя, формирования инвентарного QR-кода ИТ-ресурса;

в рамках системного тестирования были выполнены и проверены все функции информационной системы ITSM.System, заявленные в

пользовательских (функциональных) требованиях в техническом задании на разработку системы.

Все проведенные тесты были успешно пройдены, что указывает на работоспособность и корректность проверенных функций. Таким образом, все задачи настоящей работы решены в полном объеме, а исходная цель – достигнута.

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Кудина, М.В. Управление компанией : теория и практика / Кудина М.В.. – Москва : Московский государственный университет имени М.В. Ломоносова, 2013. – 325 с. – ISBN 978-5-211-06414-0. – Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. – URL: <https://www.iprbookshop.ru/54668.html> (дата обращения: 09.10.2021).
2. Джепикс, Ф. Язык программирования C# 7 и платформы .NET и .NET Core / Ф. Джепикс, Э. Троелсен. – М.: «Вильямс», 2018. – 1328 с. – ISBN 978-5-6040723-1-8. – Текст : непосредственный.
3. Ван Бон, Ян. Введение в ИТ Сервис-менеджмент /Ян Ван Бон, Г. Кеммерлинг, Д. Пондман ; пер. с англ. под ред. М.Ю. Потоцкого. – М. : Открытые Системы, 2013. – 68 с. – ISBN: 90-77212-15-9. – Текст : непосредственный.
4. ГОСТ Р ИСО/МЭК 20000-1-2013. Информационная технология (ИТ). Управление услугами. Часть 1. Требования к системе управления услугами : Национальный стандарт Российской Федерации : дата введения 2015-01-01 / подготовлен Закрытым акционерным обществом "ИТ Эксперт" и обществом с ограниченной ответственностью "Информационный аналитический вычислительный центр" (ООО "ИАВЦ") на основе собственного перевода на русский язык англоязычной версии стандарта, указанного в пункте 4 . – Москва : Стандартинформ, 2014. 23 с. – Текст : непосредственный.
5. Проектирование информационных систем. Проектный практикум : учебное пособие для студентов дневного и заочного отделений, изучающих курсы «Проектирование информационных систем», «Проектный практикум», обучающихся по направлению 230700.62 (09.03.03) / А.В. Платёнкин [и др.]. – Тамбов : Тамбовский государственный технический университет, ЭБС АСВ, 2015. – 80 с. – ISBN 978-5-8265-1409-2. – Текст : непосредственный.

6. ГОСТ 34.602-89. Единая система программной документации ЕСПД. Информационная технология. Комплекс стандартов на автоматизированные системы. Техническое задание на создание автоматизированной системы : межгосударственный стандарт : дата введения 1990-01-01 / разработан Государственным комитетом СССР по стандартам, Министерством приборостроения, средств автоматизации и систем управления СССР. – Москва : Стандартинформ, 2009. 11 с. – Текст : непосредственный.

7. Скрипник, Д. А. ITIL. IT Service Management по стандартам V.3.1 : учебное пособие / Д. А. Скрипник. – 3-е изд. – Москва, Саратов : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2020. – 372 с. – ISBN 978-5-4497-0290-6. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. – URL: <https://www.iprbookshop.ru/89410.html> (дата обращения: 11.10.2021).

8. Ларман, К. Применение UML 2.0 и шаблонов проектирования. Практическое руководство / К. Ларман ; пер. с англ. А. Ю. Шелестовой под ред. С. Н. Тригуб. – Москва : Издательский дом «Вильямс», 2013. – 736 с. – ISBN 978-5-8459-1185-8. – Текст : непосредственный. 26. Фаулер, М. UML. Основы. Краткое руководство по стандартному языку объектного моделирования: учебное пособие / М. Фаулер. – СПб.: Символ-Плюс, 2016. – 192 с. – ISBN 0-321-19368-7. – Текст: непосредственный.

9. Буч, Гради. Введение в UML от создателей языка / Гради Буч, Джеймс Рамбо, Ивар Якобсон. – М.: ДМК Пресс, 2015. – 496 с. – ISBN 978-5-94074-644-7. – Текст : непосредственный.

10. Рамбо, Дж. Применение UML 2.0. Объектно-ориентированное моделирование и разработка / Дж. Рамбо, М. Блаха ; под ред. А. Крузенштерн. – Санкт-Петербург : Питер, 2007. – 544 с. – ISBN 5-469-00814-2. – Текст : непосредственный.

11. Арлоу, Д. UML 2 и Унифицированный процесс. Практический объектно-ориентированный анализ и проектирование / Д. Арлоу, А.

Нейштадт ; пер. с англ. Н. Шатохиной под ред. А. Петухова. – Санкт-Петербург : Символ-Плюс, 2007. – 642 с. – ISBN 978-5-93286-094-6. – Текст : непосредственный.

12. Ларман, К. Применение UML 2.0 и шаблонов проектирования. Практическое руководство / К. Ларман ; пер. с англ. А. Ю. Шелестовой под ред. С. Н. Тригуб. – Москва : Издательский дом «Вильямс», 2013. – 736 с. – ISBN 978-5-8459-1185-8. – Текст : непосредственный. 26. Фаулер, М. UML. Основы. Краткое руководство по стандартному языку объектного моделирования: учебное пособие / М. Фаулер. – СПб.: Символ-Плюс, 2016. – 192 с. – ISBN 0-321-19368-7. – Текст: непосредственный.

13. Гома, Хассан. UML. Проектирование систем реального времени, параллельных и распределенных приложений / Хассан Гома. – М.: ДМК Пресс, 2016. – 700 с. – ISBN 5-94074-101-0. – Текст : непосредственный.

14. Ушаков В.Я. Energy saving in the enterprises of fuel and energy complex : учебное пособие / Ушаков В.Я., Харлов Н.Н., Чубик П.С.. – Томск : Томский политехнический университет, 2017. – 176 с. – Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. – URL: <https://www.iprbookshop.ru/84005.html> (дата обращения: 09.12.2021). – Текст: непосредственный.

15. ГОСТ Р 50.1.028-2001. Информационные технологии поддержки жизненного цикла продукции. Методология функционального моделирования : рекомендации по стандартизации : дата введения 2002-07-01 / разработаны Научно-исследовательским Центром CALS-технологий "Прикладная Логистика" при участии Всероссийского научно-исследовательского института стандартизации (ВНИИСтандарт). – Москва : ИПК Издательство стандартов, 2001. – 54 с. – Текст: непосредственный.

16. IEEE Standard 830-1998, "IEEE Recommended Practice for Software Requirements Specifications", IEEE Std., 1998

17. Прайс, М. С# 7 и .NET Core. Кросс-платформенная разработка для профессионалов / М. Прайс. – СПб.: Питер, 2018 – 640 с. – ISBN 978–5–4461–0516–8. – Текст : непосредственный.
18. Албахари, Джозеф. С# 7.0. Справочник. Полное описание языка. / Албахари Бен, Албахари Джозеф ; пер. с англ. – М. : ООО «И.Д. Вильямс», 2018. – 1040 с. – ISBN 9785977502450. – Текст : непосредственный.
19. Фримен, Э. Паттерны проектирования. Обновленное юбилейное издание. / Э. Фримен, Э. Робсон, К. Сьерра, Б. Бейтс – СПб.: Питер, 2018. – 656с. – ISBN 978–5–496–03210–0. – Текст : непосредственный.
20. Visual Studio IDE, Редактор кода [электронная документация] / MSDN. Комплекс технической документации по продуктам Microsoft. URL: <https://docs.microsoft.com/ru-ru/windows/uwp/get-started/> (дата обращения: 20.10.2021 г.)
21. Балдин К.В. Информационные системы в экономике: учебник / Балдин К.В., Уткин В.Б.. – Москва : Дашков и К, 2019. – 395 с. – ISBN 978-5-394-03244-8. – Текст : непосредственный.
22. Агальцов, В.П. Базы данных. Локальные базы данных: учебник / В.П. Агальцов. – М.: ИД ФОРУМ, НИЦ ИНФРА–М, 2016. – 352 с. – ISBN 978–5–16–011625–9. – Текст: непосредственный
23. ГОСТ Р 51904-2002. Программное обеспечение встроенных систем. Общие требования к разработке и документированию : государственный стандарт Российской Федерации : дата введения 2003-07-01 / разработан Государственным научно-исследовательским институтом авиационных систем с участием Научно-исследовательского института стандартизации и унификации. – Москва : Стандартиформ, 2005. 56 с. – Текст : непосредственный.
24. Ральф, Джонсон. Приемы объектно-ориентированного проектирования. Паттерны проектирования / Джонсон Ральф, Влссидес Джон. – СПб.: Питер, 2016. – 366 с. – ISBN 978-5-459-01720-5. – Текст : непосредственный.

25. Вайсфельд, М. Объектно-ориентированное мышление / М. Вайсфельд. – СПб.: Питер, 2014. – 304 с. – ISBN 978-5-496-00793-1. – Текст : непосредственный.
26. Вершинин, М.Т. C# Enterprise Edition. Технологии проектирования и разработки / М.Т. Вершинин, Е.О. Иванова. – М. : BHV, 2013. – 1088 с. – ISBN. 5-94157-192-5. – Текст : непосредственный.
27. Мерзлякова Е.Ю. Человеко-машинное взаимодействие : учебно-методическое пособие / Мерзлякова Е.Ю. – Новосибирск : Сибирский государственный университет телекоммуникаций и информатики, 2015. – 34 с. – Текст : непосредственный.
28. Блек, Р. Ключевые процессы тестирования. Планирование, подготовка, проведение, совершенствование / Р. Блек. – М.: Лори, 2014. – 544 с. – ISBN 978-5-85582-365-3 – Текст : непосредственный.
29. Виртуальные потоки в гибридных решающих модулях классификации сложноструктурируемых данных / А. В. Киселев, Д. Ю. Савинов, С. А. Филист [и др.]. – Текст : непосредственный // Прикаспийский журнал: управление и высокие технологии. – 2018. – № 2(42). – С. 137–149.
30. Программная инженерия : теоретический и прикладной научно-технический журнал / учредитель и издатель Издательство "Новые технологии" ; главный редактор Васенин В. А. – Москва, 2010 – . – ISSN 2220-3397. – Текст : непосредственный.

## ПРИЛОЖЕНИЕ А

### Представление графического материала

Внешний вид графического материала изображен на отдельных листах, изображения на рисунках А.1 - А.12.

# Сведения о ВКРБ

Минобрнауки России  
Юго-Западный государственный университет

Кафедра программной инженерии

## ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА ПО ПРОГРАММЕ БАКАЛАВРИАТА

«Система управления IT-услугами на основе концепции ITSM»

Руководитель ВКР  
к.т.н, доцент  
Апальков Владимир Васильевич

Автор ВКР  
студент группы ПО-71з  
Копылов Сергей Евгеньевич

				ВКРБ 2068443.09.03.04.21008			
				Сведения о ВКРБ			
				Выпускная квалификационная работа бакалавра			
				ОЗГУ ПО-71з			
Исполнитель	Степанов С. С.	Таблица	Дата	Лист	Всего	Листов	Всего
Руководитель	Апальков В. В.						
Проверитель	Копылов С. Е.						

Рисунок А.1 – Сведения о ВКРБ

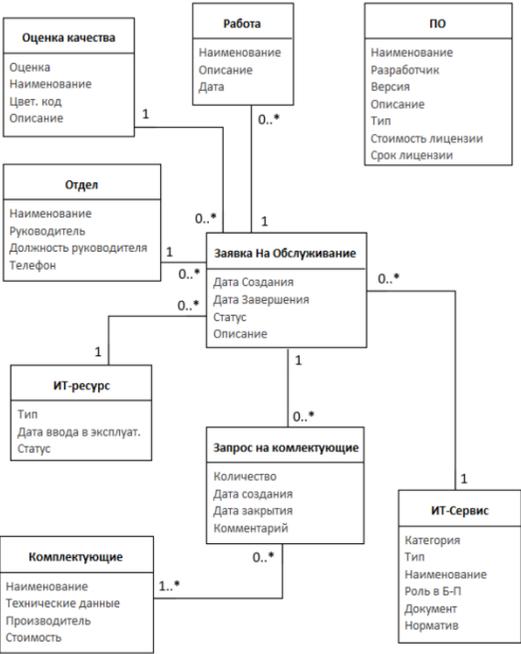
## Цель и задачи разработки

Цель настоящей работы – автоматизация процесса управления ИТ-инфраструктурой предприятия. (обозначение системы: ITMS.System). Для достижения поставленной цели в ходе работы будет решена следующая задача:  
 Разработана система управления ИТ-услугами на основе концепции ITSM.

		ВКРБ 2068443.09.03.04.21.008	
Автор работы	Земцова И. В.	Цель и задачи разработки	Лист 2
Руководитель	Кочетков С. И.		Листов 2
Модератор	Мельник А. А.	Выпускная квалификационная работа бакалавра	ЮЗГУ ПО-713

Рисунок А.2 – Цель и задачи разработки

# Диаграмма классов для концептуальной предметной области



ВКРБ 2068443.09.03.04.21.008			
Создан: И. В.	Таблица	Дата	Диаграмма классов для концептуальной предметной области
Автор работы: Руководитель: Ильяшенко А. А.	Ильяшенко А. А.	Ильяшенко А. А.	Ильяшенко А. А.
Выполнен: «автоматизация работы базиса»			ИЗГЧ ПО-71а

Рисунок А.3 – Диаграмма классов для концептуальной предметной области

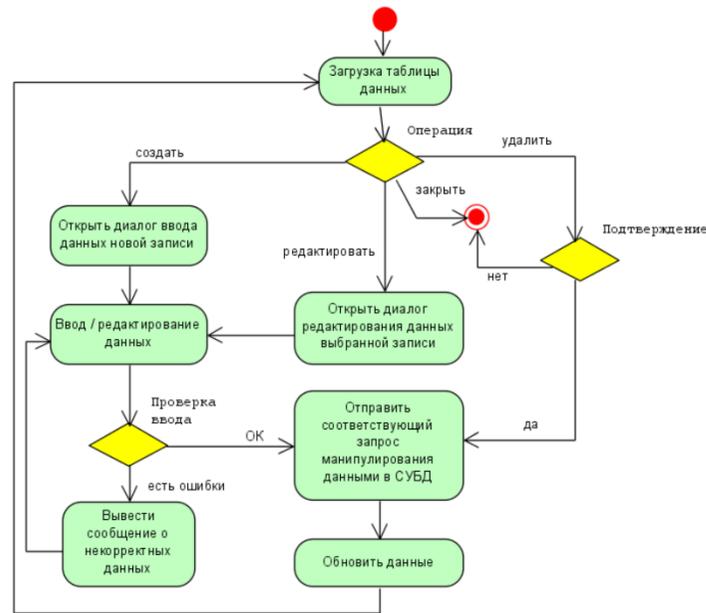
# Диаграмма прецедентов для программно-информационной системы



		ВКРБ 2068443.09.03.04.21.008			
		Диаграмма прецедентов для программно-информационной системы			
Автор работы	Исполнитель	Исполнитель	Исполнитель	Исполнитель	Исполнитель
Иванов С. С.	Иванов С. С.	Иванов С. С.	Иванов С. С.	Иванов С. С.	Иванов С. С.
Иванов С. С.	Иванов С. С.	Иванов С. С.	Иванов С. С.	Иванов С. С.	Иванов С. С.
Выполнен «автоматизация работы базиса»			03/04/2018		

Рисунок А.4 – Диаграмма прецедентов для программно-информационной системы

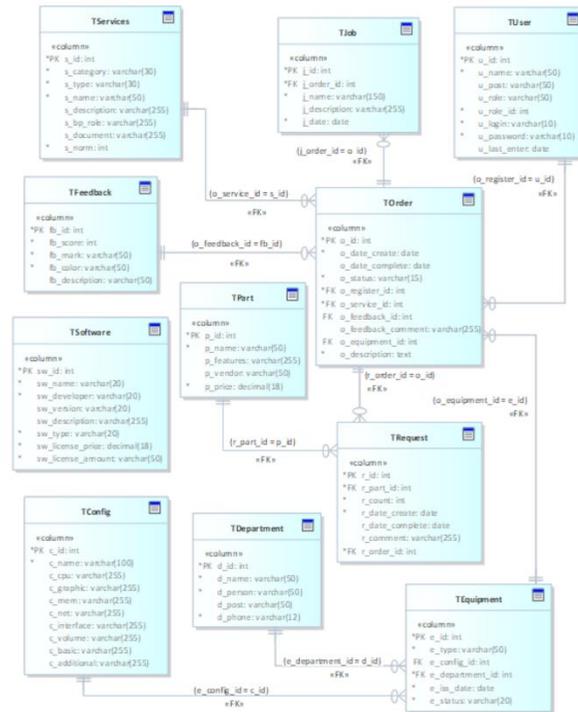
# Диаграмма последовательности для прецедентов типа ведения справочников



ВКРБ 2068443.09.03.04.21.008			
Создан и в.	Таблица	Дата	Диаграмма последовательности для прецедентов типа ведения справочников
Автор работы руководитель исполнитель	Колычев С. С. Колычев В. В. Чиркова А. А.		Лист 5 из 5
Выполнен классификационная работа: базадого			ОЗГУ ПО-71з

Рисунок А.5 – Диаграмма последовательности для прецедентов типа ведения справочников

# Диаграмма данных

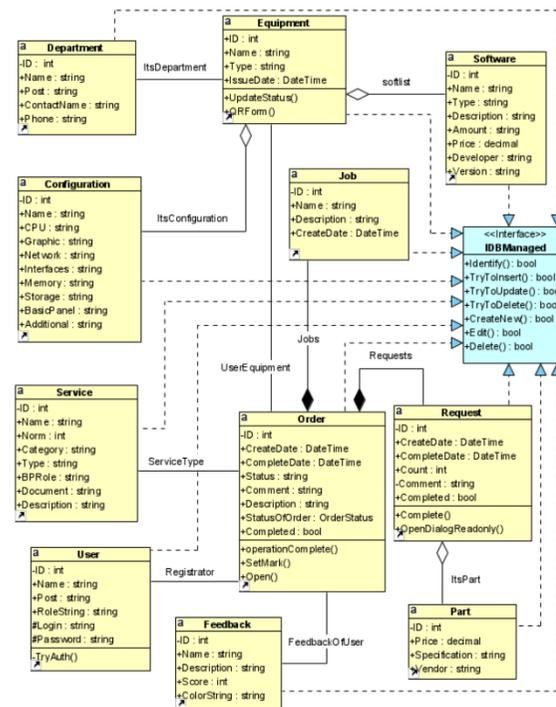


				ВКРБ 2068443.09.03.04.21.008			
	Создан	Изменен	Удален	Доп.	Исп.	Провер.	
Автор работы	Иванов С. С.						Диаграмма данных
Проверитель	Козлов С. В.						Лист С.
Корректор	Чиркова А. А.						Лист В.
				Выполнен «автоматическим работ» базиса»			
				03.09.2019			

Рисунок А.6 – Диаграмма данных



# Диаграмма классов объектной структуры предметной области



ВКР 206844.03.04.21.008			
Создан: И. И.	Таблица: БД	Дата: 01.04.2021	Страницы: 1/1
Автор работы: И. И.	Наименование: Классификация	Диаграмма классов объектной структуры предметной области	Лист 2 из 2
Выполнен «автоматизирован»			03.04.2021

Рисунок А.8 – Диаграмма классов объектной структуры предметной области



# Результаты разработки. Сформированные отчеты. Часть 2



ВКРБ 2068443.09.03.04.21.008			
Исполн. А. А.	Исполн. А. А.	Исполн. А. А.	Исполн. А. А.
Исполн. А. А.	Исполн. А. А.	Исполн. А. А.	Исполн. А. А.
Результаты разработки Сформированные отчеты. Часть 2			Исполн. А. А.
Выполнен «автоматизированная работа бухгалтера»			03/09/2021

Рисунок А.10 – Результаты разработки. Сформированные отчеты. Часть 2



# Заключение

Согласно методологии ITSM ИТ-подразделение любого предприятия рассматривается неразрывно от всей остальной бизнес-структуры предприятия. Так, ИТ-подразделения в соответствии с современными требованиями бизнеса должны интегрироваться в бизнес-структуру предприятия, а понятие ИТ-услуга должно стать частью общих внутренних бизнес-процессов. Для этого в рамках предприятия должна быть организована информационная система, объединяющая в себе клиентскую подсистему, понятия ИТ-услуги, заявки, соглашения по качеству, а также систему визуализации структуры ИТ-сегмента в разрезе общих бизнес-процессов. В ходе настоящей работы была выполнена разработка такой системы – информационной системы поддержки управления ИТ-услугами на предприятии (обозначение системы: ITMS.System).

В ходе работы были решены следующие задачи:

- разработано техническое задание на создание информационной системы поддержки управления ИТ-услугами ITMS.System;
- разработан технический проект ITMS.System с помощью унифицированного языка моделирования программных систем – UML (для разработки ТП была применена система автоматизированного проектирования – CASE-система Visual Paradigm for UML версии 16.2);
- разработан рабочий проект ITMS.System с помощью современных систем разработки программных систем (системы разработки программного обеспечения IDE Visual Studio 2019 C#.NET);
- разработан проект информационного обеспечения системы (базы данных) с применением CASE-системы Sparx Enterprise Architect версии 15.0 с поддержкой технологии forward database-engineering, СУБД MS SQL Server 2017, системы разработки функций базы данных MS SQL Management Studio.

Готовый рабочий проект представлен:

- проектом исходного кода программного обеспечения ITSM.System для IDE Visual Studio 2019;
- файлом дампа базы данных с подготовленными тестовыми данными для SQL Server 2018 (\*.bacpac-файл).

Рабочий проект был подвержен тестированию:

- набор разработанных модульных тестов охватывает все важнейшие функции разработанной ITMS.System: управления данными классов-сущностей, авторизации пользователя, формирования инвентарного QR-кода ИТ-ресурса;
- в рамках системного тестирования были выполнены и проверены все функции информационной системы ITSM.System, заявленные в пользовательских (функциональных) требованиях в техническом задании на разработку системы.

Все проведенные тесты были успешно пройдены, что указывает на работоспособность и корректность проверенных функций. Таким образом, все задачи настоящей работы решены в полном объеме, а исходная цель – достигнута.

				ВКРБ 206844309030421008			
	Создан и в.	Таблица	Вид		Имя	Вид	Имя
Истор работы	Иванов С.С.			Заключение			
Разработчик	Иванов С.С.			Алекс С.			
Проверенный	Иванов А.А.			Выполнен «автоматизация работы бухгалтера»			
				03/09/2019			

Рисунок А.12 – Заключение

## ПРИЛОЖЕНИЕ Б

### Текст программы

#### panelReportITManageCosts.cs

```
using System;
using System.Windows.Forms;

namespace IITM.Subsystems.Boss
{
    /// <summary>
    /// Класс пользовательского интерфейса - панель отчета - стоимость обслуживания
    /// </summary>
    public partial class panelReportITManageCosts : UserControl
    {
        /// <summary>
        /// Конструктор класса
        /// </summary>
        public panelReportITManageCosts()
        {
            // (Автоматическая функция) - Создание и инициализация UI-компонентов
            InitializeComponent();
            // Сброс фильтров по умолчанию
            chkShowCompletedFilter.Checked = false;
            dtpDateFilterFrom.Value = new DateTime(DateTime.Now.Year, DateTime.Now.Month, 1);
            dtpDateFilterTo.Value = new DateTime(DateTime.Now.Year, DateTime.Now.Month,
1).AddMonths(1).AddDays(-1);
            // Обновление данных отчета
            UpdateReport();
        }

        /// <summary>
        /// Применение фильтра
        /// </summary>
        /// <param name="sender">Объект события</param>
        /// <param name="e">Параметры события</param>
        private void lnkFilterOK_LinkClicked(object sender, LinkLabelLinkClickedEventArgs e)
        {
            // Применение фильтра
            UpdateReport();
        }

        /// <summary>
        /// Отмена фильтра
        /// </summary>
        /// <param name="sender">Объект события</param>
        /// <param name="e">Параметры события</param>
        private void lnkFilterCancel_LinkClicked(object sender, LinkLabelLinkClickedEventArgs e)
        {
            // Сброс фильтра
            chkShowCompletedFilter.Checked = false;
            dtpDateFilterFrom.Value = new DateTime(DateTime.Now.Year, DateTime.Now.Month, 1);
            dtpDateFilterTo.Value = new DateTime(DateTime.Now.Year, DateTime.Now.Month,
1).AddMonths(1).AddDays(-1);
            // Обновление таблицы данных
            UpdateReport();
        }

        /// <summary>
        /// Обновить отчет
        /// </summary>
    }
}
```

```

private void UpdateReport()
{
    viweRequestsWithPriceTableAdapter.FillByFilter
        (iITM_DBDataSet.viweRequestsWithPrice,
         dtpDateFilterFrom.Value, dtpDateFilterTo.Value,
         chkShowCompletedFilter.Checked ? 1 : 0);
}

private void chart1_Click(object sender, EventArgs e)
{
}
}
}
}

```

## panelReportITResources.cs

```

using System.Windows.Forms;

namespace IITM.Subsystems.Boss
{
    /// <summary>
    /// Класс пользовательского интерфейса - панель отчета - ИТ ресурсы
    /// </summary>
    public partial class panelReportITResources : UserControl
    {
        /// <summary>
        /// Конструктор класса
        /// </summary>
        public panelReportITResources()
        {
            // (Автоматическая функция) - Создание и инициализация UI-компонентов
            InitializeComponent();
            // Заполнение таблицы "Общая сводка по ИТ-ресурсам"
            FillTable_ITResourcesAll();
            // Заполнение таблицы "Количество ИТ-ресурсов по конфигурациям"
            FillTable_ITResourcesByConfig();
            // Построение диаграммы "Количество ИТ-ресурсов по статусам"
            FillChart_ITResourcesByStatus();
            // Построение диаграммы "Количество ИТ-ресурсов по департаментам"
            FillChart_ITResourcesByDepartment();
        }

        /// <summary>
        /// Заполнение таблицы "Общая сводка по ИТ-ресурсам"
        /// </summary>
        private void FillTable_ITResourcesAll()
        {
            viewEquipmentListTableAdapter.Fill(iITM_DBDataSet.viewEquipmentList);
        }
        /// <summary>
        /// Заполнение таблицы "Количество ИТ-ресурсов по конфигурациям"
        /// </summary>
        private void FillTable_ITResourcesByConfig()
        {

```

```

        viewITResByConfigTableAdapter.Fill(iITM_DBDataSet.viewITResByConfig);
    }

    /// <summary>
    /// Построение диаграммы "Количество ИТ-ресурсов по статусам"
    /// </summary>
    private void FillChart_ITResourcesByStatus()
    {
        viewITResByStatusTableAdapter.Fill(iITM_DBDataSet.viewITResByStatus);
    }
    /// <summary>
    /// Построение диаграммы "Количество ИТ-ресурсов по департаментам"
    /// </summary>
    private void FillChart_ITResourcesByDepartment()
    {
        viewITResByDepartmentTableAdapter.Fill(iITM_DBDataSet.viewITResByDepartment);
    }
}
}

```

## panelReportOrderFunnel.cs

```

using System.Windows.Forms;

namespace IITM.Subsystems.Boss
{
    /// <summary>
    /// Класс пользовательского интерфейса - панель отчета - воронка заявок
    /// </summary>
    public partial class panelReportOrderFunnel : UserControl
    {
        /// <summary>
        /// Конструктор класса
        /// </summary>
        public panelReportOrderFunnel()
        {
            // (Автоматическая функция) - Создание и инициализация UI-компонентов
            InitializeComponent();
            // Обновление данных в таблице
            UpdateDataInTable();
            // Показать воронку заявок
            ShowFunnelChart();
        }

        /// <summary>
        /// Обновление данных в таблице
        /// </summary>
        private void UpdateDataInTable()
        {
            // Заполнение таблицы данными через соответствующий адаптер
            viewOrderListTableAdapter.Fill(iITM_DBDataSet.viewOrderList);
        }

        /// <summary>
        /// Показать воронку заявок
        /// </summary>
        public void ShowFunnelChart()
    }
}

```

```

        {
            viewOrdersFunnelTableAdapter.FillOrdered(iITM_DBDataSet.viewOrdersFunnel);
        }
    }
}

```

## panelReportOrders.cs

```

using System;
using System.IO;
using System.Text;
using System.Windows.Forms;

namespace IITM.Subsystems.Boss
{
    /// <summary>
    /// Класс пользовательского интерфейса - панель отчета - сводка заявок
    /// </summary>
    public partial class panelReportOrders : UserControl
    {
        /// <summary>
        /// Конструктор класса
        /// </summary>
        public panelReportOrders()
        {
            // (Автоматическая функция) - Создание и инициализация UI-компонентов
            InitializeComponent();
            // Сброс фильтров по умолчанию
            chkShowCompletedFilter.Checked = false;
            dtpDateFilterFrom.Value = new DateTime(DateTime.Now.Year, DateTime.Now.Month, 1);
            dtpDateFilterTo.Value = new DateTime(DateTime.Now.Year, DateTime.Now.Month,
1).AddMonths(1).AddDays(-1);
            // Обновление данных отчета
            UpdateReport();
        }

        /// <summary>
        /// Сохранить
        /// </summary>
        using System.Windows.Forms;

namespace IITM.Subsystems.Boss
{
    /// <summary>
    /// Класс пользовательского интерфейса - панель отчета - воронка заявок
    /// </summary>
    public partial class panelReportOrderFunnel : UserControl
    {
        /// <summary>
        /// Конструктор класса
        /// </summary>
        public panelReportOrderFunnel()
        {
            // (Автоматическая функция) - Создание и инициализация UI-компонентов
            InitializeComponent();
            // Обновление данных в таблице
            UpdateDataInTable();
            // Показать воронку заявок
            ShowFunnelChart();
        }
    }
}

```

```

    }

    /// <summary>
    /// Обновление данных в таблице
    /// </summary>
    private void UpdateDataInTable()
    {
        // Заполнение таблицы данными через соответствующий адаптер
        viewOrderListTableAdapter.Fill(iITM_DBDataSet.viewOrderList);
    }

    /// <summary>
    /// Показать воронку заявок
    /// </summary>
    public void ShowFunnelChart()
    {
        viewOrdersFunnelTableAdapter.FillOrdered(iITM_DBDataSet.viewOrdersFunnel);
    }
}

/// <param name="sender">Объект события</param>
/// <param name="e">Параметры события</param>
private void lnkMenuSave_LinkClicked(object sender, LinkLabelLinkClickedEventArgs e)
{
    SaveFileDialog SD = new SaveFileDialog();
    SD.InitialDirectory = Application.StartupPath;
    SD.Filter = "HyperText Markup Language files (*.html)|*.html";
    if (SD.ShowDialog() == DialogResult.OK)
    {
        string FileName = SD.FileName;
        FileStream fs = new FileStream(FileName, FileMode.OpenOrCreate);
        StreamWriter sw = new StreamWriter(fs);
        sw.WriteLine(this.wbbReport.DocumentText);
        sw.Close();
        fs.Close();
    }
}

/// <summary>
/// Печать
/// </summary>
/// <param name="sender">Объект события</param>
/// <param name="e">Параметры события</param>
private void lnkMenuPreview_LinkClicked(object sender, LinkLabelLinkClickedEventArgs e)
{
    wbbReport.ShowPrintDialog();
}

/// <summary>
/// Предпросмотр
/// </summary>
/// <param name="sender">Объект события</param>
/// <param name="e">Параметры события</param>
private void lnkMenuPrint_LinkClicked(object sender, LinkLabelLinkClickedEventArgs e)
{
    wbbReport.ShowPrintPreviewDialog();
}

/// <summary>
/// Применение фильтра
/// </summary>
/// <param name="sender">Объект события</param>
/// <param name="e">Параметры события</param>

```

```

private void lnkFilterOK_LinkClicked(object sender, LinkLabelLinkClickedEventArgs e)
{
    // Применение фильтра
    UpdateReport();
}
/// <summary>
/// Отмена фильтра
/// </summary>
/// <param name="sender">Объект события</param>
/// <param name="e">Параметры события</param>
private void lnkFilterCancel_LinkClicked(object sender, LinkLabelLinkClickedEventArgs e)
{
    // Сброс фильтра
    chkShowCompletedFilter.Checked = false;
    dtpDateFilterFrom.Value = new DateTime(DateTime.Now.Year, DateTime.Now.Month, 1);
    dtpDateFilterTo.Value = new DateTime(DateTime.Now.Year, DateTime.Now.Month,
1).AddMonths(1).AddDays(-1);
    // Обновление таблицы данных
    UpdateReport();
}

/// <summary>
/// Обновить отчет
/// </summary>
private void UpdateReport()
{
    // Отчет
    StringBuilder sb = new StringBuilder();
    sb.Append("<html><head></head>");
    sb.Append("<body>");

    // Временные переменные
    decimal gen_summ = 0;

    // Служебная переменная-счетчик
    int i = 1;

    DateTime __date_in_ = dtpDateFilterFrom.Value;
    DateTime __date_out_ = dtpDateFilterTo.Value;

    // Подключение к таблице сводки по заявкам
    IITM_DBDataSetTableAdapters.viewOrdersGeneralTableAdapter vord
        = new IITM_DBDataSetTableAdapters.viewOrdersGeneralTableAdapter();
    IITM_DBDataSet.viewOrdersGeneralDataTable vodt
        = vord.GetDataByFilter(__date_in_, __date_out_,
        chkShowCompletedFilter.Checked ? 1 : 0);

    // Заголовок отчета
    sb.AppendFormat("<h2 align=center>ОТЧЕТ О ВЫПОЛНЕНИИ ЗАЯВОК
ПОЛЬЗОВАТЕЛЕЙ</font></h2>");
    sb.AppendFormat("<p align=center><font family='courier new'>за период с
<u><i>{0}</i></u> по <u><i>{1}</i></u></font></p>",
        __date_in_.ToLongDateString(), __date_out_.ToLongDateString());

    // Шапка таблицы
    sb.Append("<table width='100%' cellpadding=0 cellspacing=0 border=all
bordercolor=green>");
    sb.Append("<tr>");
    sb.Append("<td align=center><font face='courier new'><b>№ п./п.</b></font></td>");
    sb.Append("<td align=center><font face='courier new'><b>№ Заявки</b></font></td>");

```

```

        sb.Append("<td align=center><font face='courier new'><b>Дата
создания/<br>завершения</b></font></td>");
        sb.Append("<td align=center><font face='courier new'><b>Сервис</b></font></td>");
        sb.Append("<td align=center><font face='courier new'><b>Описание</b></font></td>");
        sb.Append("<td align=center><font face='courier new'><b>Оценка</b></font></td>");
        sb.Append("<td align=center><font face='courier new'><b>Кол-
во<br>работ</b></font></td>");
        sb.Append("<td align=center><font face='courier new'><b>Кол-во<br>комплект-
х</b></font></td>");
        sb.Append("<td align=center><font face='courier new'><b>Сумма, р.</b></font></td>");
        sb.Append("</tr>");

// Данные для отчета
foreach (System.Data.DataRow dr in vodt.Rows)
{
    // Временные переменные
    // Чтение данных
    int id = Convert.ToInt32(dr.ItemArray[0]);
    DateTime? d2 = Convert.IsDBNull(dr.ItemArray[1])
        ? null
        : (DateTime?)Convert.ToDateTime(dr.ItemArray[1]);
    DateTime d1 = Convert.ToDateTime(dr.ItemArray[2]);
    string description = Convert.ToString(dr.ItemArray[3]);
    string service = Convert.ToString(dr.ItemArray[4]);
    string mark = Convert.IsDBNull(dr.ItemArray[5])
        ? "..."
        : Convert.ToString(dr.ItemArray[5]);
    int jobsCnt = Convert.ToInt32(dr.ItemArray[6]);
    int partsCnt = Convert.ToInt32(dr.ItemArray[7]);
    decimal summ = Convert.IsDBNull(dr.ItemArray[8])
        ? 0
        : Convert.ToDecimal(dr.ItemArray[8]);
    gen_summ += summ;

    // Запись в отчет
    sb.Append("<tr>");
    sb.AppendFormat("<td align=center><font face='courier new'>{0}</font></td>",
i++);
        sb.AppendFormat("<td align=center><font face='courier new'>{0}</font></td>",
id.ToString("0000"));
        sb.AppendFormat("<td align=center><font face='courier new'>{0}</font></td>",
        $"{d1.ToShortDateString()}/<br>{(d2 == null ? "-" :
((DateTime)d2).ToShortDateString())}");
        sb.AppendFormat("<td align=left><font face='courier new'>{0}</font></td>",
service);
        sb.AppendFormat("<td align=left><font face='courier new'>{0}</font></td>",
description);
        sb.AppendFormat("<td align=center><font face='courier new'>{0}</font></td>",
mark);
        sb.AppendFormat("<td align=center><font face='courier new'>{0}</font></td>",
jobsCnt);
        sb.AppendFormat("<td align=center><font face='courier new'>{0}</font></td>",
partsCnt);
        sb.AppendFormat("<td align=center><font face='courier new'>{0}</font></td>",
summ);
        sb.Append("</tr>");
}

// Конец таблицы
sb.Append("</table>");

```

```

        // Сводные данные заказа
        sb.AppendFormat("<br><font face='courier new'><b>Всего заявок:
</b><i><u>{0}</u></i></font><br>", i - 1);
        sb.AppendFormat("<font face='courier new'><b>Общая сумма затрат: </b><i>{0}</i>
(руб.)</font><br>", gen_summ);

        sb.Append("</body></html>");

        // Показать
        wbbReport.DocumentText = sb.ToString();
    }
}

```

## panelReportServiceQuality.cs

```

using System;
using System.Windows.Forms;

namespace IITM.Subsystems.Boss
{
    /// <summary>
    /// Класс пользовательского интерфейса - панель отчета - качество услуг
    /// </summary>
    public partial class panelReportServiceQuality : UserControl
    {
        /// <summary>
        /// Конструктор класса
        /// </summary>
        public panelReportServiceQuality()
        {
            // (Автоматическая функция) - Создание и инициализация UI-компонентов
            InitializeComponent();
            // Обновление данных в таблице
            UpdateDataInTable();
            // Показать диаграмму оценок качества
            ShowQValuesChart();
        }

        /// <summary>
        /// Обновление данных в таблице
        /// </summary>
        private void UpdateDataInTable()
        {
            // Заполнение таблицы данными через соответствующий адаптер
            viewOrderListTableAdapter.Fill(iITM_DBDataSet.viewOrderList);
        }

        /// <summary>
        /// Показать диаграмму оценок качества
        /// </summary>
        public void ShowQValuesChart()
        {
            // Предварительная очистка диаграммы
            chart1.Series[0].Points.Clear();
            // Подключение к таблице оценок
            IITM_DBDataSetTableAdapters.viewScoresTableAdapter viewScoresTableAdapter

```

```

        = new IITM_DBDataSetTableAdapters.viewScoresTableAdapter();
IITM_DBDataSet.viewScoresDataTable viewScoresRows
        = viewScoresTableAdapter.GetData();

// Временные переменные
string color = "";
string mark = "";
int score = 0;

// Добавление данных в диаграмму
foreach (System.Data.DataRow dataRow in viewScoresRows.Rows)
{
    color = Convert.IsDBNull(dataRow.ItemArray[2])
        ? "150;150;150"
        : Convert.ToString(dataRow.ItemArray[2]);
    mark = Convert.IsDBNull(dataRow.ItemArray[1])
        ? "Нет оценки"
        : Convert.ToString(dataRow.ItemArray[1]);
    score = Convert.ToInt32(dataRow.ItemArray[0]);

    chart1.Series[0].Points.AddXY(mark, score);
    chart1.Series[0].Points[chart1.Series[0].Points.Count - 1].Color
        = Database.Entities.Feedback.ParseColor(color);
}
}
}
}
}

```

## Order.cs

```

using System;
using System.Windows.Forms;

namespace IITM.Database.Entities
{
    /// <summary>
    /// Статусы заявки
    /// </summary>
    public enum OrderStatus
    {
        Unknown = -1,    // неизвестный
        New,             // новая
        InProcess,      // в процессе
        WaitForParts,   // ожидание комплектующих
        Solved,         // решена (отзыв не оставлен)
        Completed       // завершена (решена и отзыв оставлен)
    }

    /// <summary>
    /// /класс заявки пользователя
    /// </summary>
    public class Order : Database.IDBManaged
    {
        // -----
        // Доступ к атрибутам пользователя
        // -----
        /// <summary>

```

```

/// ID
/// </summary>
public int ID { get; private set; }
/// <summary>
/// Дата создания
/// </summary>
public DateTime CreateDate { get; private set; }
/// <summary>
/// Дата завершения
/// </summary>
public DateTime? CompleteDate { get; private set; }
/// <summary>
/// Регистрирующий пользователь
/// </summary>
public User Registrator { get; private set; }
/// <summary>
/// Статус заявки
/// </summary>
public string Status { get; private set; }
/// <summary>
/// Сервис по заявке
/// </summary>
public Service ServiceType { get; private set; }
/// <summary>
/// Отзыв/оценка работ по заявке
/// </summary>
public Feedback FeedbackOfUser { get; private set; }
/// <summary>
/// Оборудование, по которому создана заявка
/// </summary>
public Equipment UserEquipment { get; private set; }
/// <summary>
/// Комментарий к отзыву
/// </summary>
public string Comment { get; private set; }
/// <summary>
/// Описание
/// </summary>
public string Description { get; private set; }

/// <summary>
/// Статус заявки пользователя как перечислитель
/// </summary>
public OrderStatus StatusOfOrder
{
    get
    {
        switch (Status)
        {
            case "Новая": return OrderStatus.New;
            case "В работе": return OrderStatus.InProcess;
            case "Поставка": return OrderStatus.WaitForParts;
            case "Решена": return OrderStatus.Solved;
            case "Завершена": return OrderStatus.Completed;
            default: return OrderStatus.Unknown;
        }
    }
}

/// <summary>
/// Статус завершенной заявки

```

```

/// </summary>
public bool Completed { get { return CompleteDate != null; } }

/// <summary>
/// Конструктор - Инициализация данных
/// </summary>
/// <param name="__id_">Необязательный идентификатор</param>
public Order(object __id_ = null)
{
    Identify(__id_);
}

/// <summary>
/// Адаптер подключения к таблице БД
/// </summary>
readonly IITM_DBDataSetTableAdapters.TOrderTableAdapter __t_adapter =
    new IITM_DBDataSetTableAdapters.TOrderTableAdapter();

/// <summary>
/// Инициализация данных
/// </summary>
/// <param name="__id_">Необязательный идентификатор</param>
/// <returns>Результат операции</returns>
public bool Identify(object __id_ = null)
{
    try
    {
        // Инициализация по идентификатору
        if (__id_ != null)
        {
            // Получение данных из таблицы
            IITM_DBDataSet.TOrderDataTable __table = __t_adapter.GetDataByID((int)__id_);
            // Чтение полученных данных
            if (__table.Rows.Count > 0)
            {
                ID = Convert.ToInt32(__table.Rows[0].ItemArray[0]);
                CompleteDate = Convert.IsDBNull(__table.Rows[0].ItemArray[1])
                    ? (DateTime?)null
                    : Convert.ToDateTime(__table.Rows[0].ItemArray[1]);
                Status = Convert.ToString(__table.Rows[0].ItemArray[2]);
                Registrator = Convert.ToInt32(__table.Rows[0].ItemArray[3]) ==
IITMS.CurrentUser.ID
                    ? IITMS.CurrentUser
                    : new User(Convert.ToInt32(__table.Rows[0].ItemArray[3]));
                ServiceType = new Service(Convert.ToInt32(__table.Rows[0].ItemArray[4]));
                FeedbackOfUser = Convert.IsDBNull(__table.Rows[0].ItemArray[5])
                    ? null
                    : new Feedback(Convert.ToInt32(__table.Rows[0].ItemArray[5]));
                Comment = Convert.ToString(__table.Rows[0].ItemArray[6]);
                UserEquipment = Convert.IsDBNull(__table.Rows[0].ItemArray[7])
                    ? null
                    : new Equipment(Convert.ToInt32(__table.Rows[0].ItemArray[7]));
                Description = Convert.ToString(__table.Rows[0].ItemArray[8]);
                CreateDate = Convert.ToDateTime(__table.Rows[0].ItemArray[9]);
            }
        }
        else
        {
            MessageBox.Show("Запись не найдена!", Application.ProductName,

```

```

        MessageBoxButtons.OK, MessageBoxIcon.Error);
        // Отказ
        return false;
    }
}
else
{
    // Инициализация по умолчанию
    CompleteDate = null;
    CreateDate = DateTime.Now.Date;
    Status = "Новая";
    Registrator = IITMS.CurrentUser;
    ServiceType = new Service();
    FeedbackOfUser = null;
    Comment = Description = "";
    UserEquipment = null;
}
// Успешно
return true;
}
catch (Exception ex)
{
    MessageBox.Show("Ошибка при выполнении операции: " + ex.Message,
Application.ProductName,
        MessageBoxButtons.OK, MessageBoxIcon.Error);
    // Отказ
    return false;
}
}
// -----
/// <summary>
/// Вставка записи в БД
/// </summary>
/// <returns>Результат операции</returns>
public bool TryToInsert()
{
    try
    {
        // Добавление новой записи в БД
        __t_adapter.InsertQuery(Status, Registrator.ID, ServiceType.ID,
CreateDate);
        (UserEquipment == null ? (int?)null : UserEquipment.ID), Description,
        // Получение автоматически присвоенного ID
        ID = Convert.ToInt32(__t_adapter.GetLastID());
        // Возврат успеха
        return true;
    }
    catch (Exception ex)
    {
        MessageBox.Show("Ошибка при выполнении операции: " + ex.Message,
Application.ProductName,
        MessageBoxButtons.OK, MessageBoxIcon.Error);
        // Возврат неудачи
        return false;
    }
}
// -----
/// <summary>
/// Обновление записи в БД
/// </summary>
/// <returns>Результат операции</returns>

```

```

public bool TryToUpdate()
{
    try
    {
        // Выполнение команды через адаптер таблицы

        // Функция не поддерживается

        // Возврат успеха
        return true;
    }
    catch (Exception ex)
    {
        MessageBox.Show("Ошибка при выполнении операции: " + ex.Message,
Application.ProductName,
        MessageBoxButtons.OK, MessageBoxIcon.Error);
        // Возврат неудачи
        return false;
    }
}
// -----
/// <summary>
/// Удаление записи из БД
/// </summary>
/// <returns>Результат операции</returns>
public bool TryToDelete()
{
    try
    {
        // Выполнение команды через адаптер таблицы
        __t_adapter.DeleteQuery(ID);
        // Возврат успеха
        return true;
    }
    catch (Exception ex)
    {
        MessageBox.Show("Ошибка при выполнении операции: " + ex.Message,
Application.ProductName,
        MessageBoxButtons.OK, MessageBoxIcon.Error);
        // Возврат неудачи
        return false;
    }
}
// -----

// РЕАЛИЗАЦИЯ ИНТЕРФЕЙСА С ПОЛЬЗОВАТЕЛЕМ -----

/// <summary>
/// Диалог ввода данных
/// </summary>
Input.dialogNewOrder Dialog = null;

/// <summary>
/// Команда пользователя - создать новый
/// </summary>
/// <param name="__add_data_">Дополнительные данные</param>
/// <returns>Результат выполнения команды</returns>
public bool CreateNew(object __add_data_ = null)
{
    try

```

```

{
    // Создание диалога ввода данных
    Dialog = new Input.dialogNewOrder(this);

    // Организация диалога с пользователем
    if (Dialog.ShowDialog() == DialogResult.OK)
    {
        // Применение введенных значений
        ServiceType = new Service(Dialog.ServiceID);
        UserEquipment = Dialog.EquipmentID == null
            ? null
            : new Equipment(Dialog.EquipmentID);
        Description = Dialog.Description;
        // Выполнить вставку в БД
        return TryToInsert();
    }
    else
    {
        // Возврат неудачи - отмена действия пользователем
        return false;
    }
}
catch (Exception ex)
{
    MessageBox.Show("Ошибка при выполнении операции: " + ex.Message,
Application.ProductName,
        MessageBoxButtons.OK, MessageBoxIcon.Error);
    // Возврат неудачи
    return false;
}
}
/// <summary>
/// Команда пользователя - редактировать
/// </summary>
/// <param name="__add_data_">Дополнительные данные</param>
/// <returns>Результат выполнения команды</returns>
public bool Edit(object __add_data_ = null)
{
    try
    {
        // Функция не поддерживается

        return true;
    }
    catch (Exception ex)
    {
        MessageBox.Show("Ошибка при выполнении операции: " + ex.Message,
Application.ProductName,
            MessageBoxButtons.OK, MessageBoxIcon.Error);
        // Возврат неудачи
        return false;
    }
}
/// <summary>
/// Команда пользователя - удалить
/// </summary>
/// <returns>Результат операции</returns>
public bool Delete()
{
    try
    {

```

```

        // Запрос на удаление
        if (MessageBox.Show("Подтверждаете удаление выбранной записи ?", "Удаление",
            MessageBoxButtons.YesNo, MessageBoxIcon.Question) == DialogResult.Yes)
        {
            // Удалить из БД
            return TryToDelete();
        }
        // Возврат неудачи - отмена действия пользователем
        return false;
    }
    catch (Exception ex)
    {
        MessageBox.Show("Ошибка при выполнении операции: " + ex.Message,
Application.ProductName,
            MessageBoxButtons.OK, MessageBoxIcon.Error);
        // Возврат неудачи
        return false;
    }
}
// -----

/// <summary>
/// Завершение заявки
/// </summary>
/// <returns>Результат операции</returns>
public bool Complete()
{
    // Завершить можно только заявку со статусом "В работе"
    if (StatusOfOrder != OrderStatus.InProcess)
    {
        MessageBox.Show("Завершить можно только заявку со статусом \"В работе\"!",
Application.ProductName,
            MessageBoxButtons.OK, MessageBoxIcon.Exclamation);
        // Возврат неудачи
        return false;
    }

    try
    {
        // Запрос на удаление
        if (MessageBox.Show("Подтверждаете завершение выбранной заявки ?", "Завершение",
            MessageBoxButtons.YesNo, MessageBoxIcon.Question) == DialogResult.Yes)
        {
            // Завершить
            CompleteDate = DateTime.Now.Date;
            __t_adapter.Complete(CompleteDate, "Решена", ID);
            // Успешно
            return true;
        }
        // Возврат неудачи - отмена действия пользователем
        return false;
    }
    catch (Exception ex)
    {
        MessageBox.Show("Ошибка при выполнении операции: " + ex.Message,
Application.ProductName,
            MessageBoxButtons.OK, MessageBoxIcon.Error);
        // Возврат неудачи
        return false;
    }
}

```

```

}

/// <summary>
/// Оставить оценку качества по заявке
/// </summary>
/// <returns>Результат операции</returns>
public bool SetMark()
{
    // Оставить оценку качества можно только по заявке со статусом "Решена"
    if (StatusOfOrder != OrderStatus.Solved)
    {
        MessageBox.Show("Оставить оценку качества можно только по заявке со статусом
\"Решена\"!",
            Application.ProductName,
            MessageBoxButtons.OK, MessageBoxIcon.Exclamation);
        // Возврат неудачи
        return false;
    }

    try
    {
        // Указать оценку качества
        Input.dialogSetQualityMark sqmd = new Input.dialogSetQualityMark();
        if (sqmd.ShowDialog() == DialogResult.OK)
        {
            // Параметры оценки качества
            FeedbackOfUser = new Feedback(sqmd.FeedbackID);
            Comment = sqmd.Description;
            __t_adapter.SetMark(FeedbackOfUser.ID, Comment, "Завершена", ID);
            // Успешно
            return true;
        }
        else
            // Отменено пользователем
            return false;
    }
    catch (Exception ex)
    {
        MessageBox.Show("Ошибка при выполнении операции: " + ex.Message,
Application.ProductName,
            MessageBoxButtons.OK, MessageBoxIcon.Error);
        // Возврат неудачи
        return false;
    }
}

/// <summary>
/// Открытие формы с подробностями заказа
/// </summary>
/// <returns>Форма с подробностями заказа</returns>
public Form Open() => new Input.dialogOrderContent(this);
}
}

```

## МЕСТО ДЛЯ ДИСКА

