

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Кузбасский государственный технический университет
имени Т. Ф. Горбачева»

Институт информационных технологий, машиностроения
и автотранспорта
Кафедра информационных и автоматизированных производственных
систем

Алексей Николаевич Стародубов
Игорь Сергеевич Кузнецов
Василий Валентинович Зиновьев
Алла Валериевна Кузнецова

УПРАВЛЕНИЕ ЖИЗНЕННЫМ ЦИКЛОМ ИНФОРМАЦИ- ОННЫХ СИСТЕМ

Методические материалы к практическим занятиям

Рекомендовано учебно-методической комиссией направления
09.04.02 «Информационные системы и технологии», в качестве
электронного издания для использования в учебном процессе

Кемерово 2024

Рецензенты: Чичерин И. В. – кандидат тех наук, доцент, заведующий кафедрой информационных и автоматизированных производственных систем ФГБОУ ВО «Кузбасский государственный технический университет имени Т. Ф. Горбачева»

Сыркин И. С. – доцент, кандидат тех. наук кафедры информационных и автоматизированных производственных систем ФГБОУ ВО «Кузбасский государственный технический университет имени Т. Ф. Горбачева»

Стародубов, А.Н., Кузнецов, И.С., Зиновьев, В.В., Кузнецова, А.В. Управление жизненным циклом информационных систем: методические материалы по выполнению практических занятий: для обучающихся направления 09.04.02 «Информационные системы и технологии»/ сост. А. Н. Стародубов, И. С. Кузнецов, В. В. Зиновьев, А. В. Кузнецова, Кузбасский государственный технический университет имени Т. Ф. Горбачева. – Кемерово, 2024. – Текст: электронный.

В данных методических материалах изложено содержание по выполнению практических занятий, цель работ, структура жизненного цикла ИС, автоматизированные системы управления ЖЦ, задания по выполнению, контрольные вопросы.

© Кузбасский государственный
технический университет
имени Т.Ф. Горбачева, 2024

© Стародубов А.Н.
Кузнецов И.С.
Зиновьев В.В.
Кузнецова А.В.
составление 2024

ОГЛАВЛЕНИЕ

ЛАБОРАТОРНАЯ РАБОТА № 1. ЖИЗНЕННЫЙ ЦИКЛ ИНФОРМАЦИОННЫХ СИСТЕМ	5
1. ЦЕЛЬ РАБОТЫ	5
2. СТРУКТУРА ЖИЗНЕННОГО ЦИКЛА ИС	5
2.1 Стадии ЖЦ ИС	5
2.2 Стандарты ЖЦ ИС	6
2.3 Процессы ЖЦ ИС	7
3. АВТОМАТИЗИРОВАННЫЕ СИСТЕМЫ УПРАВЛЕНИЯ ЖЦ	8
3.1 Системы автоматизированного проектирования (CAE/CAD/CAM/PDM)	10
3.2 Автоматизированная система управления цепочками поставок (SCM)	17
3.3 Автоматизированная система управления.....	20
технологическим процессом (SCADA/CNC).....	20
3.4 Автоматизированные системы управления предприятием (ERP/MRP-II)	22
3.5 Автоматизированные системы управления взаимоотношениями с заказчиками (CRM/S&SM)	24
3.6 Производственные исполнительные системы (MES).....	26
3.7 Автоматизированные системы управления ЖЦ продукции (PLM)	28
4. ЗАДАНИЕ К ВЫПОЛНЕНИЮ	30
5. КОНТРОЛЬНЫЕ ВОПРОСЫ.....	31
СПИСОК РЕКОМЕНДУЕМОЙ ЛИТЕРАТУРЫ	32
ЛАБОРАТОРНАЯ РАБОТА №2. ОБЩИЕ СВЕДЕНИЯ ОБ УПРАВЛЕНИИ ПРОЕКТАМИ	33
1. ЦЕЛЬ РАБОТЫ	33
2. ТЕОРЕТИЧЕСКОЕ ПОЛОЖЕНИЕ	33
2.1 Общие понятия управления проектами	33
2.2 Процессы управления проектами	34
2.3 Методологии и стандарты управления проектами	35
2.4 Система управления проектами Битрикс 24.....	37
3. ПРАКТИЧЕСКАЯ ЧАСТЬ	48
4. ПРАКТИЧЕСКОЕ ЗАДАНИЕ	61
5. КОНТРОЛЬНЫЕ ВОПРОСЫ.....	61
СПИСОК РЕКОМЕНДУЕМОЙ ЛИТЕРАТУРЫ	62

ЛАБОРАТОРНАЯ РАБОТА №3. КЛАССИФИКАЦИЯ ПРОЕКТОВ	63
1. ЦЕЛЬ РАБОТЫ.....	63
2. ТЕОРЕТИЧЕСКОЕ ПОЛОЖЕНИЕ	63
2.1 Классификация проектов	63
2.2 Модели и методологии проектирования ИС	66
3. ПРАКТИЧЕСКАЯ ЧАСТЬ	70
4. ПРАКТИЧЕСКОЕ ЗАДАНИЕ.....	75
5. КОНТРОЛЬНЫЕ ВОПРОСЫ.....	76
СПИСОК РЕКОМЕНДУЕМОЙ ЛИТЕРАТУРЫ	76
ЛАБОРАТОРНАЯ РАБОТА №4. CASE-ТЕХНОЛОГИИ РАЗРАБОТКИ АВТОМАТИЗИРОВАННОЙ СИСТЕМЫ	77
1. ЦЕЛЬ РАБОТЫ.....	77
2. ТЕОРЕТИЧЕСКОЕ ПОЛОЖЕНИЕ	77
2.1 Проектирование ИС	77
2.2 Понятие CASE-технологий	78
2.3. Принцип работы CASE-средств на примере программы	79
Ramus Educational	79
3. ПРАКТИЧЕСКАЯ ЧАСТЬ	83
4. ПРАКТИЧЕСКОЕ ЗАДАНИЕ.....	96
5. КОНТРОЛЬНЫЕ ВОПРОСЫ.....	96
СПИСОК РЕКОМЕНДУЕМОЙ ЛИТЕРАТУРЫ	97
ЛАБОРАТОРНАЯ РАБОТА №5. СПЕЦИАЛИЗИРОВАННЫЕ ПРОГРАММА УПРАВЛЕНИЯ ЖИЗНЕННЫМ ЦИКЛОМ: MATHCAD	98
1. ЦЕЛЬ РАБОТЫ.....	98
2. ТЕОРЕТИЧЕСКОЕ ПОЛОЖЕНИЕ	98
2.1 Основы работы в MathCAD.....	98
2.2 Установка и первый запуск MathCAD	98
2.3 Использование MathCAD для моделирования технологических процессов.....	101
3. ПРАКТИЧЕСКАЯ ЧАСТЬ	102
4. ВОПРОСЫ ДЛЯ ЗАЩИТЫ РАБОТЫ	105
СПИСОК РЕКОМЕНДУЕМОЙ ЛИТЕРАТУРЫ	106

ЛАБОРАТОРНАЯ РАБОТА № 1. ЖИЗНЕННЫЙ ЦИКЛ ИНФОРМАЦИОННЫХ СИСТЕМ

1. ЦЕЛЬ РАБОТЫ

Целью данной работы является получение и закрепление теоретических знаний о структуре жизненного цикла информационных систем и автоматизированных систем по его управлению.

2. СТРУКТУРА ЖИЗНЕННОГО ЦИКЛА ИС

Жизненный цикл информационной систем (ЖЦ ИС) – последовательность действий, заключенная в временном промежутке, отсчёт которого начинается с момента определения необходимости в создании информационной системы и заканчивается в момент завершения эксплуатации, и утилизации данной системы.

Информационная система носит предметный характер, который определяется на основе каждого конкретно взятого предприятия, для которого и разрабатывается данная система. Направленность деятельности предприятия и особенности, связанные с ней, оказывают существенное влияние на состав информационной системы, но при этом большинство предприятий, как и большинство ИС имеют общую схожую структуру. Практически любое предприятие представляет собой группу взаимодействующих подразделений, осуществляющих свои функциональные подзадачи в рамках деятельности всей организации. Также и любая информационная система, в независимости от её характера, сложности и масштаба, обязательно проходит через общие ступени развития, которые принято разделять на стадии и этапы, составляющих ЖЦ ИС.

2.1 Стадии ЖЦ ИС

Стадия ЖЦ ИС – ограниченная во времени часть процесса по созданию и функционированию ИС, характеризующаяся формированием конкретного продукта (отчётной документации, моделей, программных модулей) в соответствии с заданными требованиями для этой стадии.

Содержание стадий и их количество может отличаться, в зависимости от особенностей каждой отдельно взятой системы. Однако, можно выделить следующие основные этапы (стадии) присущие каждому ЖЦ ИС:

- Формирование требований или концепции при использовании анализа предметной области;
- Проектирование;
- Реализация;
- Внедрение или ввод системы в эксплуатацию;
- Эксплуатация, дальнейшее сопровождение ИС;

В качестве завершения процессов жизненного цикла ИС принято считать момент вывода её из эксплуатации.

Для успешной организации управления и разработки жизненного цикла ИС для каждой стадии обычно определяется: перечень работ и их последовательность, используемые методы, технологии и средства, ответственные лица, роли и др.

2.2 Стандарты ЖЦ ИС

В настоящий момент времени для разработки современных ИС используются стандарты, в которых обобщается и систематизируется весь ранее накопленный опыт, успешные результаты исследований и работ множества специалистов различных отраслей. Подобные знания способствуют формированию наиболее эффективных методов и средств по созданию и развитию ИС. Обычно в стандартах представлен общий набор этапов и процессов, частных операций, а также их последовательность, с помощью которых осуществляется регламентация работ на всех стадиях ЖЦ ИС. Кроме того, применение такого подхода позволяет достичь не только высокой эффективности для проектируемых систем на и обеспечить возможность их взаимодействия между собой, за счёт обобщенности применяемых методов и средств.

К наиболее распространённым стандартам в мировом сообществе относятся следующие:

ГОСТ 34.601-90 – содержит подробную информацию о перечни работ на каждой стадии жизненного цикла ИС.

ISO/IEC 12207 (International Organization of Standardization /International Electrotechnical Commission) 1995 – базовый стандарт по организации и управлению жизненного цикла ИС.

Rational Unified Process (RUP). Основопологающим для этого стандарта является применение итерационного подхода, включающего разбиение на четыре основных этапа: инициализация, исследование, проектирование и внедрение. Каждый этап может быть разбит на итерации по выполнению которых производится какой-либо конечный продукт, работа над которым может продолжаться в последующих итерациях, вплоть до конечной версии продукта.

Microsoft Solution Framework (MSF). Подход похож на RUP, поскольку также выделяет четыре основные стадии: предварительный анализ, разработка и проектирования, стабилизация. Отличительной стороной является применение итерационного подхода с использованием объектно-ориентированного моделирования.

2.3 Процессы ЖЦ ИС

Процесс представляет собой совокупность связанных действий по осуществлению преобразования входных данных в выходные. Обычно процесс характеризуется наличием решаемых задач, набором исходных данных и получаемых результатов.

Согласно международному стандарту ISO/IEC 12207 процессы ЖЦ ИС подразделяются на следующие три группы: основные, вспомогательные и организационные (Рисунок 1).



Рисунок 1 – Процессы жизненного цикла ИС

1) *Основные процессы* (приобретение, поставка, разработка, эксплуатация, сопровождение).

2) *Вспомогательные процессы* (документирование, управление конфигурацией, обеспечение качества, верификация, аттестация, оценка, аудит, разрешение проблем), помогают в осуществлении выполнения основных процессов ЖЦ.

3) *Организационные процессы* (управление проектами, создание инфраструктуры проекта, усовершенствование, обучение), отвечают за администрирование и формализацию жизненного цикла ИС.

3. АВТОМАТИЗИРОВАННЫЕ СИСТЕМЫ УПРАВЛЕНИЯ ЖЦ

В настоящее время информационные технологии промышленного уровня основываются на системе единого информационного пространства (объединения информационных ресурсов) предприятия. В международном сообществе эта концепция носит

название CALS-технологии (Continuous Acquisition and Lifecycle Support).

CALS-технологии – непрерывная информационная поддержка жизненного цикла продукции, основывающаяся на применении на применении современных информационных технологиях и техники на всех стадиях жизненного цикла продукции.

Непрерывная информационная поддержка позволяет обеспечить наиболее эффективное управление процессами ЖЦ изделия, а также эффективно организовать взаимодействие всех участников этих процессов (заказчиков, поставщиков, исполнителей, ответственных лиц и тд.).

Применение CALS-технологий включает в себя использования различных автоматизированных систем управления продукцией на всех стадиях её жизненного цикла (Рисунок 2).

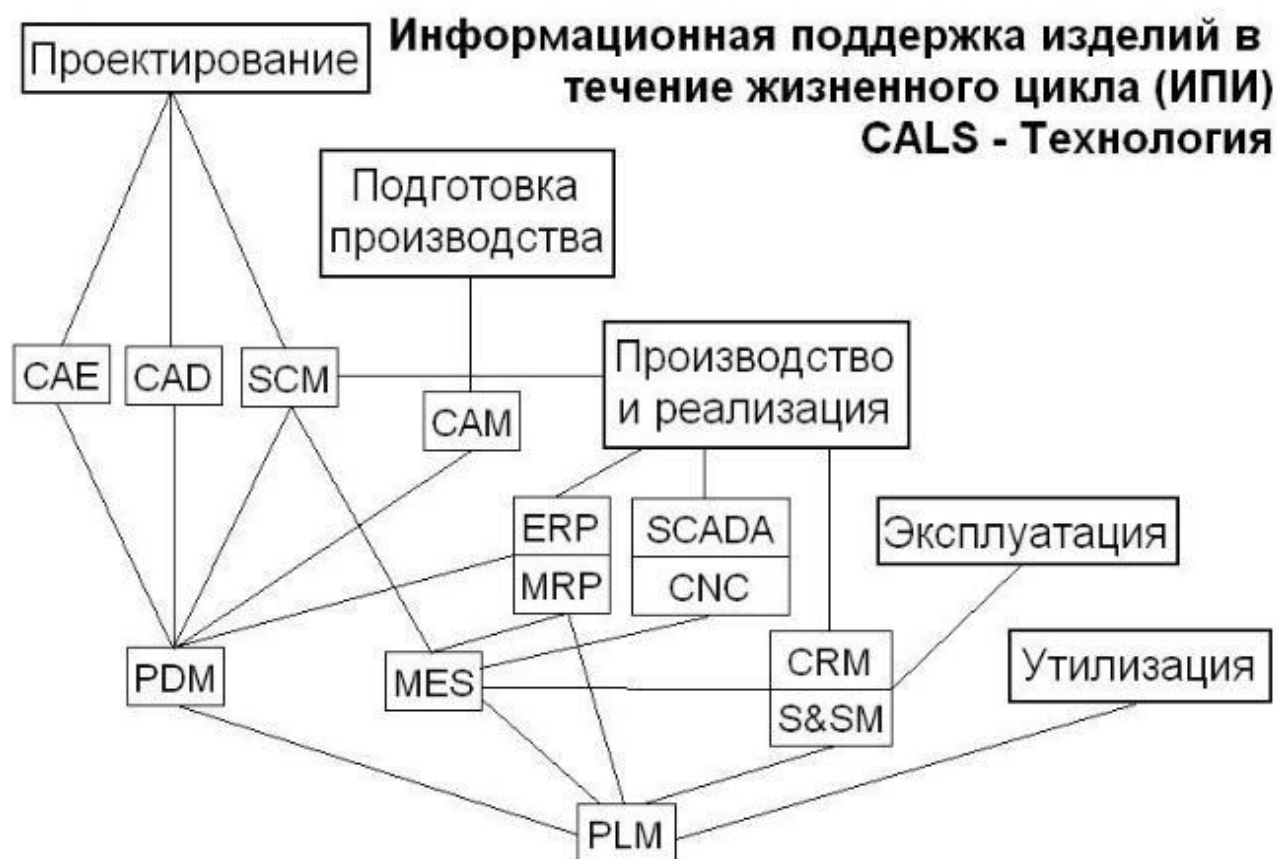


Рисунок 2 – Автоматизированные системы управления этапами ЖЦ

3.1 Системы автоматизированного проектирования (CAE/CAD/CAM/PDM)

Для проектирования различных объектов (сложные механизмы, аппараты, машины, сети, коммуникации и др) в всевозможных областях науки и промышленности используются средства автоматизированного проектирования (САПР) или CAD (*Computer Aided Design/автоматизированное проектирование*).

CAD представляет собой множество модулей, ориентированных на различные проектные задачи (построение трёхмерных моделей деталей, составление чертежей, оформление документации) применяемых для тех или иных объектов. Объединение модулей в организационной структуре позволяет обеспечить практически полное автоматизированное проектирование.

CAM (*Computer Aided Manufacturing/автоматизированная технологическая подготовка производства*) – системы предназначенные для проектирования и обработки данных об изделиях, изготавливаемых на станках с ЧПУ (числовое программное управление), а также подбор и формирования программ для этих станков (токарные, фрезерные, шлифовальные и др). Применение САМ-систем позволяет с высокой точностью производить сложнопрофильные детали и сократить цикл их производства.

CAE-системы (*Computer Aided Engineering/автоматизированные расчеты и анализ*) – комплекс расчётно-аналитических систем направленных на решение любых инженерных задач требующих сложных расчётов.

В основе работ систем САМ/CAE лежит сконструированная трёхмерная модель изделия в системах-CAD.

Широкая направленность решаемых задач САПР безусловно является огромным их достоинством, однако, это она также может служить и причиной возможного возникновения проблем с организацией общего потока данных, построением баз данных, определением протоколов и форматов и др.

Для решения данной проблемы применяются системы PDM.

PDM-система (*Product Data Management/управление проектными данными*) – система поддержки управления совокупной информации о продукте (сложные технические изделия, сети, информационные системы и др). Благодаря PDM-системам возможно организовать обработку большого количества данных и

различной технической и проектной информации, необходимой на многих стадиях ЖЦ. PDM-системы способны воспринимать информацию любого типа, обрабатывать её и выдавать пользователю в необходимом ему виде. Используя системы PDM можно составить различную документацию о конфигурациях системы, маршрутизации изделия, подробную информацию о деталях, используемых материалах и тд.

Одной из главных целей применения PDM-систем является обеспечение возможности работы группы над проектом, за счёт обобщения фрагментов информации и вывода их в реальном режиме времени.

Таблица 1 – Перечень популярных систем CAD/CAM/CAE/PDM

№	Наименование системы	Область / сфера применения	Доступность	Основные достоинства
CAD-системы				
1	AutoCAD	САПР для создания чертежей и трехмерных моделей	<ul style="list-style-type: none"> - Бесплатная пробная версия; - платная подписка 	<ul style="list-style-type: none"> - Простота; - мультиплатформенность; - многоформативность; - широкий функционал и огромный набор инструментов - широкие возможности в 3D моделировании
2	BricsCAD	(САПР), которая объединяет 2D черчение и 3D моделирование в едином формате (.dwg), применяемая для проектирования в области: архитектуры, строительства,	<ul style="list-style-type: none"> - Пробная 30-дневная версия; - платные версии системы 	<ul style="list-style-type: none"> - Есть русский интерфейс; - использование единого формата (.dwg) - огромный набор функциональных возможностей проектирования; - оперативная техническая поддержка (в том числе и на русском языке)

		машиностроении, проектировании инженерных сетей и др.		
3	КОМПАС-3D	Проектирование изделий основного и вспомогательного производств во многих отраслях промышленности (машиностроение, станкостроение, судостроение, авиастроение, приборостроение и др.)	<ul style="list-style-type: none"> - Учебная версия; - бесплатная версия; - платные версии системы 	<ul style="list-style-type: none"> - Включена в Реестр российского ПО; - простота эксплуатации; - многоформативность; - регулярная обновления и техническая поддержка от разработчика; - широкие возможности разработки чертежей и 2D проектирования;
4	SolidWorks	Автоматизация работ промышленного предприятия на этапах конструкторской и технологической подготовки производства. Обеспечивает разработку изделий любой степени сложности и назначения	<ul style="list-style-type: none"> - Учебные версии; - платные версии 	<ul style="list-style-type: none"> - Широкий набор инструментов; - есть русский интерфейс; - возможность лёгкой интеграции с АСУ другого типа из семейства Dassault Systemes
5	FreeCAD	Задачи параметрического трёхмерного моделирования, объектов реального ми-	<ul style="list-style-type: none"> - Полностью бесплатная версия 	<ul style="list-style-type: none"> - Доступность; - использование полностью открытой, гибкой платформы - наличие русского интерфейса;

		ра любого размера		- хороший набор инструментов проектирования
CAE-системы				
1	Mathlab	Вычислительные технические задачи, математическое моделирование в области проектирования	- Учебная версия; - демоверсия; - платная версия	- Широкие возможности в области математического моделирования; - высокая точность; - высокая скорость работы
2	T-FLEX Анализ	Математическое моделирование распространённых физических явлений и решение практических задач проектирования	- Демоверсия; - платная локальная версия; - платная сетевая версия	- Гибкость подхода при решении задач; - включён в Реестр российского ПО; - легкая интеграция с другими системами T-FLEX семейства; - простота и удобность эксплуатации
3	SimulationX	Моделирование физико-технических объектов и систем (в особенности выделяется изучение работы автомобильных приводов)	- Демоверсия с существенными ограничениями; - платная версия	- Легкая интеграция с другими CAE/CAD системами; - широкий пакет инструментов; - поддерживает язык моделирования Modelica; - применение метода гармонического баланса; - надёжность системы
4	ANSYS	Решение пространственных задач механики деформации	- Несколько вариантов учебных версий; - несколько вариантов бес-	- Интуитивно понятное удобное управление; - доступность

		руемого твёрдого тела и механики конструкций, задач механики жидкости и газа, теплопередачи и теплообмена, электродинамики, акустики	платных версий; - платные версии	
5	APM WinMachine	Автоматизированный расчет и проектирование механического оборудования и конструкций в области машиностроения	- Пробная версия (30 дней) - бесплатная ограниченная версия; - платная версия	- Включена в Реестр российского ПО; - доступность; - простота интерфейса; - огромное количество дополнительных возможностей системы
САМ-системы				
1	PowerMill	Подготовки высокоэффективных управляющих программ для фрезерных станков с ЧПУ, быстрое создание траектории УП используя 2.5D обработку	- Пробная бесплатная версия без ограничений (30 дней) - Платная версия	- Многофункциональность; - высокая точность; - надёжность;
2	MasterCAM	Проектирование эффективно управляющих про-	- Учебная версия; - Демоверсия; - Платные вер-	- Широкий набор инструментов; - надёжность; - быстрая скорость рас-

		грамм для обработки по двум - пяти осям на фрезерных, токарных и проволочных электроэрозионных станках, штамповку, гибку, а также полное конструирование моделей в 3D	сии	чѐтов; - интегрированные библиотеки из CAD-систем; - наличие русского интерфейса;
3	EdgeCAM	Создания управляющих программ обработки для станков с ЧПУ, токарной, фрезерной и электроэрозионной групп	- Ограниченная пробная версия; - Платные версии системы под заказ	- Возможность интеграции с любой CAD-системой; - высокая точность; - многофункциональность; - наличие русского интерфейса
4	ArtCAM Pro	Пространственное моделирование и механическая обработка. Позволяет автоматически создавать трехмерные модели из плоских изображений с последующим производством высококачественных изделий на станках с ЧПУ	- Демоверсия; - платная версия	- Широкий набор инструментов; - проектирование эскизов любой сложности; - широкий спектр дополнительных возможностей - наличие русского интерфейса

5	SprutCAM	Программирование станков с ЧПУ и промышленных роботов	<ul style="list-style-type: none"> - Пробная бесплатная версия без ограничений (30 дней) - учебная версия; - платные версии 	<ul style="list-style-type: none"> - Широкий функционал; - простота и удобность интерфейса - доступность - внесён в Реестр российского ПО
PDM-системы				
1	Solid Edge	Твердотельное и поверхностного моделирования производства: моделирования деталей и сборок, создания чертежей, управления конструкторскими данными	<ul style="list-style-type: none"> - Пробная бесплатная версия (30 дней); - платные версии 	<ul style="list-style-type: none"> - Широкий набор инструментов; - наличие русского интерфейса; - применение уникальной «синхронной технологии»; - оперативная, высокоэффективная техническая поддержка - многоформативность
2	Rhinoceros (Rhino)	Промышленный дизайн, архитектура, корабельном проектировании, ювелирном и автомобильном дизайне, мультимедиа	<ul style="list-style-type: none"> - Пробная бесплатная версия (30 дней); - платная учебная версия (скидка 90%); - платные версии 	<ul style="list-style-type: none"> - Разнообразии функциональной применимости; - быстрая адаптация и обучаемость системы; - поддержка большого числа форматов; - доступность, относительно невысокая цена
3	Creo Elements/Pro	Твёрдотельное и параметрическое моделирование, черчение, функции ЧПУ, хранение и управление данными	<ul style="list-style-type: none"> - Демоверсия; - платные версии 	<ul style="list-style-type: none"> - Многофункциональность; - эффективная интеграция CAD/CAM/CAE семейства PTC;

		ми об изделии		
4	PDM STEP Suite	Управление инженерными данными об изделиях различного вида и сложности	- Учебная версия; - демоверсия - платная версия	- Внесена в Реестр российского ПО; - доступная; - простота эксплуатации
5	SmarTeam	Создание информационной базы о продукции, конструкторских отделах, малых и средних предприятий	- Демоверсия; - платная версия	- Создание структуры хранения данных любой сложности; - многофункциональность; - лёгкая интеграция

3.2 Автоматизированная система управления цепочками поставок (SCM)

Потребность в организации поставок оборудования и сырья возникает уже на ранних процессах этапа проектирования. Для обеспечения и автоматизирования данной потребности можно воспользоваться системами SCM.

SCM – система (Supply Chain Management/управление цепочками поставок) – автоматизированная система, предназначенная для организации и контроля товародвижения на всех этапах жизненного цикла продукции.

С помощью SCM-систем осуществляется организация цепей поставок не только необходимых производству ресурсов, но и уже полностью готовой продукции.

Таблица 2 – Перечень популярных АСУ цепочками поставок

№	Наименование системы	Область / сфера применения	Доступность	Основные достоинства
1	Oracle SCM	Управление цепочек поставок для компаний среднего и крупного бизнеса	<ul style="list-style-type: none"> - Платная версия; - онлайн демонстрация; 	<ul style="list-style-type: none"> - Широкий основной и дополнительный функционал; - возможность простой интеграции с другими АС семейства Oracle; - наличие большой базы дополнительного ПО и других информационных решений от Oracle (в том числе есть и бесплатные)
2	1С:Транспортная логистика	Любые компании, где требуется организовать транспортные и экспедиторские процессы, управление цепочек поставок	<ul style="list-style-type: none"> - Наличие бесплатной учебной версии; - наличие платных версий различной конфигурации (единовременная покупка или платная подписка); - демоверсия 	<ul style="list-style-type: none"> - Включена в Реестр российского ПО; - простая адаптация к условиям любого предприятия; - широкий функционал; - доступность и разнообразие сборок и различных конфигураций систем;
3	Infor SCM	Автоматизация управления цепочек поставок	Бесплатные и демоверсии системы отсут-	- Детализация и многофункциональность в

		крупных корпораций	ствуют, для приобретения необходимо напрямую обратиться к разработчику	рамках управления цепочек поставок; - использование ИИ для прогнозирования
4	Streamline	Автоматизация управления цепочек поставок малого, среднего и крупного бизнеса	- Платная версия; - демоверсия с ограниченным пробным периодом	- Простой и интуитивно понятный интерфейс; - возможность интеграции с любыми базами данных и ERP-системами; - наличие дискретного моделирования; - использование ИИ для прогнозирования спроса
5	Visary SCM	Автоматизация деятельности государственных и муниципальных компаний, а также компаний среднего и крупного бизнеса	Бесплатные и демоверсии системы отсутствуют, для приобретения необходимо напрямую обратиться к разработчику	- Включена в Реестр российского ПО; - встроенные инструменты ИИ для решения прикладных задач; - гибкая лицензионная политика; - использование открытых отечественных технологий

3.3 Автоматизированная система управления технологическим процессом (SCADA/CNC)

АСУ ТП (автоматизированная система управления технологическим процессом) – комплексный подход, состоящий из программно-технических средств, направленный на обеспечение автоматизации управления технологическим оборудованием на промышленных предприятиях.

Целью использования АСУ ТП является автоматизирование основных технологических операций на определённом участке или производстве в целом, выполнение которого приводит к выпуску относительно завершённого продукта.

Система *SCADA (Supervisory Control And Data Acquisition)* необходима для осуществления сбора и обработки данных о состоянии оборудования и происходящих технологических процессах на производстве. Помимо этого, системы SCADA необходимы для разработки и внедрения программного обеспечения для внутреннего оборудования, входящего как объект системы управления технологическими процессами производством.

Прямое программное управление технологическим оборудованием осуществляется при помощи *CNC (Computer Numerical Control)* на базе контроллеров, интегрированных в это оборудование.

Таблица 3 – Перечень популярных АСУ ТП

№	Наименование системы	Область / сфера применения	Доступность	Основные достоинства
1	MasterSCADA	Задачи учета и диспетчеризации объектов промышленности, ЖКХ и зданий.	- Демоверсия; - ограниченная бесплатная версия; - платные версии разной конфигурации	- Внесён в Реестр российского ПО; - кроссплатформенность (Microsoft Windows, Linux, Unix, Qnx, Android, Эльбрус и др); - неограниченная гибкость вычислительных возмож-

				<p>ностей;</p> <ul style="list-style-type: none"> - мощная трехмерная графика и мультимедиа; - расширение поддержки языков МЭК-61131-3 - простая интеграция; - надёжность
2	TRACE MODE	Автоматизации технологических процессов, диспетчеризации, телемеханики, учета ресурсов и автоматизации зданий	<ul style="list-style-type: none"> - Бесплатная версия; - платные версии разной конфигурации 	<ul style="list-style-type: none"> - Внесён в Реестр российского ПО; - полная поддержка всех языков стандарта МЭК 61131-3; - интегрированная среда разработки; - простота освоения; - интегрированная среда разработки - оперативная техническая поддержка
3	OpenSCADA	Любые задачи диспетчеризации и контроля	<ul style="list-style-type: none"> - Полная бесплатная версия 	<ul style="list-style-type: none"> - Открытость; - многоплатформенность; - модульность; - масштабируемость
4	Simatic WinCC	Автоматизация технологических процессов крупных предприятий и зданий	<ul style="list-style-type: none"> - Бесплатная пробная версия - платная версия 	<ul style="list-style-type: none"> - Наличие интерфейса на русском язык; - широкие функциональные возможности; - надёжность; - многолетнее присутствие на рынке
5	Vijeo Citect	Автоматизация технологических процессов, дис-	<ul style="list-style-type: none"> - Бесплатная ограниченная версия; 	<ul style="list-style-type: none"> - Наличие интерфейса на русском язык;

		петчеризация и контроль	- Платные версии разной конфигурации	- гибкость в решении поставленных задач; - Простота освоения; - многолетнее присутствие на рынке
--	--	-------------------------	--------------------------------------	--

3.4 Автоматизированные системы управления предприятием (ERP/MRP-II)

Автоматизация управления на верхних иерархических уровнях предприятия осуществляется за счёт систем стандарта MRP-II и ERP.

MRP-II (Manufacturing (Material) Requirement Planning/планирование производственных ресурсов) – система, обеспечивающая управление бизнес-процессами предприятиями, в рамках которой осуществляется: планирование, анализ и контроль использования ресурсов предприятия (сырьевые, энергетические, оборудование, сотрудники и другие). Концепция MRP-II позволяет осуществлять автоматизированное управление производственными ресурсами на протяжении всего жизненного цикла продукта.

ERP (Enterprise Resource Planning/планирование ресурсов предприятия) – система, обеспечивающая управление бизнес-процессами предприятия, как внутренними, так и внешними, которые необходимы для производства продукции, принятия заказов, продаж, осуществления взаимодействия с партнёрами и клиентами.

Упрощённо, ERP-система воплощает в себе все функции MRP-II, а также планирование внешними ресурсами, управление финансами и маркетинг. Применение ERP-системы фактически позволяет управлять всеми необходимыми ресурсами для производства и реализации продукции предприятия, а также даёт возможность динамически анализировать и оперативно вносить изменения в любой части цепочки планирования.

Таблица 4 – Перечень популярных АСУП

№	Наименование системы	Область / сфера применения	Доступность	Основные достоинства
1	1С:ERP	Автоматизация бизнес-процессов, повышение эффективности менеджмента среднего и крупного бизнеса	<ul style="list-style-type: none"> - Учебная версия; - демоверсия; - платные версии 	<ul style="list-style-type: none"> - Многофункциональность; - лёгкая интеграция и дополнительные возможности за счёт семейства систем и ПО от 1С; - внесена в Реестр российского ПО; - регулярные обновления и развитие систем
2	Галактика ERP	Управление бизнес-процессами предприятия с учётом национальной специфики	<ul style="list-style-type: none"> - Платные версии 	<ul style="list-style-type: none"> - Внесена в Реестр российского ПО; - гибкая настройка под отечественное законодательство и стандарты; - модульность; - поддержка открытых стандартов разработки
3	Компас	Автоматизация управления предприятия любой направленности	<ul style="list-style-type: none"> - Бесплатная пробная версия (14 дней); - платные версии 	<ul style="list-style-type: none"> - Внесена в Реестр российского ПО; - модульность; - удобная настройка систем
4	ERPNext	Управление и планирование операци-	<ul style="list-style-type: none"> - Самодостаточная бес- 	<ul style="list-style-type: none"> - Наличие русского интерфей-

		онной деятельности бизнеса	платная версия; - платные версии	са; - интеграции с крупными международными сервисами; - доступность; - современный и удобный интерфейс
5	Парус	Автоматизация управления среднего и крупного бизнеса, государственных и муниципальных корпораций	- Пробная версия; - платная версия	- Внесена в Реестр российского ПО; - эффективная интеграция с ERP-системами;

3.5 Автоматизированные системы управления взаимоотношениями с заказчиками (CRM/S&SM)

Как было уже сказано в предыдущем разделе системы ERP могут обеспечивать некоторые взаимодействия с внешними участниками производственной цепочки (в основном с партнёрами), однако, имеют существенные ограничения в организации эффективных коммуникационных методах связи с клиентами. Для решения этого вопроса можно воспользоваться специализированными CRM-системами.

CRM-система (Customer Relationship Management) – система автоматизированного управления взаимодействий с клиентами, в основе которой лежит использование современных информационных технологий и технических средств, сохранение подробной и информации о клиентах и их заказах, а также удобный вывод и анализ этой информации. CRM-системы способствуют повышению уровня продаж продукции, улучшению качества обслуживания клиентов, развития маркетинговых стратегий компании и др.

Маркетинговые функции иногда возлагаются на систему *S&SM (Sales and Service Management)*, которая, кроме того, служит для решения проблем обслуживания.

**Таблица 5 – Перечень популярных АСУ взаимоотношениями
с заказчиками**

№	Наименование системы	Область / сфера применения	Доступность	Основные достоинства
1	Битрикс24	Автоматизация продаж и маркетинга практически любой продукции и для любого предприятия	<ul style="list-style-type: none"> - Бесплатная версия с ограничениями; - полная платная версия (ежемесячная подписка) 	<ul style="list-style-type: none"> - Включён в Реестр российского ПО; - обладает широким набором инструментов в области crm; - огромный пакет дополнительного функционала; - большие возможности интеграции с другими системами ПО через внутренний магазин системы; - доступность; - простота в использовании
2	bpm'online	Автоматизация управления продажами, маркетингом	<ul style="list-style-type: none"> - Бесплатная пробная версия; - платные версии 	<ul style="list-style-type: none"> - Широкий набор инструментов crm и интерфейса; - чёткое разделение версий системы для предприятий малого, среднего и крупного бизнеса; - детализация работы процесса продаж; - возможность интеграции со множеством ПО - наличие русского интерфейса
3	АmoCRM	Авторизация управления маркетингом и продажами	<ul style="list-style-type: none"> - Бесплатная пробная версия; - платные версии (ежемесячные подписки) 	<ul style="list-style-type: none"> - Включён в Реестр российского ПО; - широкий функционал; - удобность и прак-

				тичность системы; - доступность
4	SberCRM	Управление маркетингом, продажи и взаимодействие с клиентами оптовой торговли и сферы услуг	- Бесплатная версия с ограничениями; - платные версии (ежемесячная подписка)	- Включён в Реестр российского ПО; - доступность, низкая стоимость; - простота и удобность; - большой набор дополнительных услуг; - быстро развивающаяся система
5	Sales Creatio	Автоматизация процессов и ускорение полного цикла продаж	- Бесплатная пробная версия; - платные версии	- Простота в освоении; - детализация процессов; - высокая скорость и надёжность системы

3.6 Производственные исполнительные системы (MES)

Системы MES (Manufacturing Execution System) – система автоматизированного управления производством, являющаяся связующим звеном между производственными и бизнес-процессами предприятия. Позволяет *оперативно* запрашивать информацию о предприятии, проводить анализ и принимать эффективные решения, контролировать их и документировать от момента поступления заказа на создание новой продукции до её конечного выпуска.

Как правило, применяются на конкретном производственном участке, но могут быть использованы для всего предприятия.

Таблица 6 – Перечень популярных автоматизированных производственных исполнительных систем

№	Наименование системы	Область / сфера применения	Доступность	Основные достоинства
1	PolyPlan	Оперативно-календарное (цехового) планирование, формирование опе-	- Пробная версия; - платная версия	- Возможность работы с неавтоматизированными система-

		ративных план-графиков работы оборудования для автоматизированных производственных систем и для производственных систем с невысоким уровнем автоматизации		ми; - наличие русского интерфейса; - высокая точность построения расписания
2	ФОБОС	Оперативное планирование и диспетчерский контроль, мониторинг состояния производственных заказов, контроль состояния и распределение ресурсов	- Пробная версия; - платная версия	- Высокая скорость и точность работы системы; - наличие русского интерфейса - достойный функционал
3	YSB.Enterprise.Mes	Формирование портфеля заказов, управление складом, расчет заработной платы для малых предприятий	- Пробная версия; - платная версия	- Наличие русского интерфейса; - Простота эксплуатации; - Низкая ценовая политика
4	SAP Manufacturing Execution	Управление и контроль производственных операций и операций производственного участка	- Пробная версия; - платная версия	- мультифункциональность; - точность и надёжность системы; - создание иерархичной структуры хранения информации; - наличие русского интерфейса
5	DIAMES	Автоматизация управления производством на предприятиях машино-	- Пробная версия; - платная версия	- Широкий функционал; - наличие русского интерфей-

		строения, дискретного производства, непрерывного производства		са
--	--	---	--	----

3.7 Автоматизированные системы управления ЖЦ продукции (PLM)

PLM-система (Product Lifecycle Management/ управление жизненным циклом продукции) – автоматизированная система управления жизненным циклом продукции на всех его стадиях, представляющая собой комплекс программного обеспечения, технических средств, инфраструктуру, методов организации, анализа и контроля. PLM-системы представляют собой эффективную совокупность автоматизированных участков производства, в виде интеграции ранее перечисленных системы в единую.

Таблица 7 – Перечень популярных
автоматизированных систем управления ЖЦ продукции

№	Наименование системы	Область / сфера применения	Доступность	Основные достоинства
1	Siemens Teamcenter	Обеспечение организации коллективной работы сотрудников предприятия (группы предприятий) с данными об изделии и связанных с ним процессах на всех этапах его жизненного цикла	- Сильно ограниченная демоверсия; - Платная версия	- Наличие русского интерфейса; - модульность; - масштабируемость; - универсальность; - широкие возможности управления составом изделия; - лёгкая интеграция с производственными системами
2	Arena PLM	Объединение информации о продук-	- Платные версии;	- Простота использования;

		тах, людях и процессах в единой корпоративной платформе для ускорения проектирования и разработки продуктов с помощью облачного программного обеспечения	- Запрос демонстрации напрямую у производителей и дистрибьюторов	- эффективная интеграция с ERP-системами
3	PTC Windchill PLM	Автоматизированное управление ЖЦ изделия	- Демоверсия; - платная версия	- Многофункциональность; - простота и удобность эксплуатации;
4	Оракул Agile	Управления жизненным циклом продукта от начального сбора ресурсов до проектирования, создание прототипов, получения сертификации, производства, настройки, обслуживания и вывода из эксплуатации	- Демоверсия; - платная версия	- Широкий функционал; - гибкость в управлении ЖЦ изделия; - универсальность
5	T-FLEXPLM	Решение задач в области управления жизненным циклом изделий и организации деятельности предприятий	- Пробная версия; - платная версия	- Включена в Реестр российского ПО; - широкие функциональные возможности системы; - универсальность; - возможность интеграции с другими системами; - относительно невысокая цена

4. ЗАДАНИЕ К ВЫПОЛНЕНИЮ

1. Выбрать и согласовать с преподавателем изделие, управление ЖЦ которого будет автоматизирован с помощью АСУ ЖЦ. Описать его назначение, сферу применения, отрасль промышленности.

2. На основе изученного материала в данной работе, а также открытых источников информации, в том числе сети интернет, заполнить Таблицу 9 для выбранной ИС, используя набор автоматизированных систем управления ЖЦ представленный в Таблице 8 (необходимо выбрать наиболее подходящие системы к вашему изделию):

Таблица 8 – Список АСУ ЖЦ

АСУ ЖЦ
Arena PLM; AutoCAD; T-FLEXPLM; Visary SCM; Streamline; Infor SCM; 1C:Транспортная логистика; Oracle SCM; SmarTeam; PDM STEP Suite; Creo Elements/Pro; Rhinoceros (Rhino); Solid Edge; SprutCAM; ArtCAM Pro; EdgeCAM; MasterCAM; PowerMill; APM WinMachine; ANSYS; SimulationX; T-FLEX Анализ; Mathlab; FreeCAD; SolidWorks; КОМПАС-3D; BricsCAD; MasterSCADA; TRACE MODE; OpenSCADA; Simatic WinCC; Vijeo Citect; 1C:ERP; Галактика ERP; Компас; ERPNext; Парус; Битрикс24; bpm'online; AmoCRM; SberCRM; Sales Creatio; PolyPlan; ФОБОС; YSB.Enterprise.Mes; SAP Manufacturing Execution; DIAMES; Siemens Teamcenter; PTC Windchill PLM; Оракул Agile.

Таблица 9 – Распределение АСУ по стадиям ЖЦ

№	Стадия ЖЦ	Вид АСУ	Наименование системы	Преимущественные характеристики
1	Проектирование	CAD	FreeCAD	Доступная, полностью бесплатная система, обладающая широким набором инструментов и русским интерфейсом. Применяется для моделирования любых объектов реального мира. Идеально подойдёт для обучения пользо-
		

				вателя обладающего некоторым уровнем знанием в области IT, но не работающего с данным видом систем ранее.
2	Подготовка производства
3	Производство и реализация
		crm	Битрикс24	Универсальная система, включённая в Реестр российского ПО. Обладает широким набором инструментов и дополнительных возможностей. Доступная, есть бесплатная версия. Подойдет к любому виду продукции, также обучению начинающих пользователей.
...				

3. Написать вывод о составленной таблице, привести аргументы в пользу произведённого выбора систем управления жизненным циклом.

4. Сформировать отчёт о проделанной работе.

5. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Понятие ЖЦ ИС?

2. Перечислите основные стадии (этапы) ЖЦ ИС?

3. Назовите группы процессов ЖЦ ИС выделяемые в стандарте ISO/IEC 12207?
4. Что представляют собой CALS-технологии?
5. Расшифруйте аббревиатуру «САПР», дайте определение данному понятию?
6. На чём основывается работа систем САМ/САЕ?
7. Расшифруйте аббревиатуру «АСУ ТП», для чего применяется система SCADA?
8. Для чего используют системы MES?
9. Что такое системы PLM, какие системы могут входить в состав PLM?

СПИСОК РЕКОМЕНДУЕМОЙ ЛИТЕРАТУРЫ

1. Избачков, С. Ю., Петров, В. Н. Информационные системы-СПб.: Питер, 2012. – 655 с.
2. Зараменских, Е. П. Управление жизненным циклом информационных систем / Е. П. Зараменских. – Новосибирск: Изд-во ЦРНС, 2014. – 270 с.
3. ГОСТ Р ИСО/МЭК 15288-2005. Системная инженерия. Процессы жизненного цикла систем.
4. ГОСТ 34.601-90. Информационная технология. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Стадии создания.
5. Аверченков, В. И. Информационные системы в производстве и экономике: учеб. пособие / В. И. Аверченков, Ф. Ю. Лозбинев, А. А. Тищенко; науч. ред. В. И. Аверченков. – 2-е изд., стереотип. – М.: ФЛИНТА, 2011. – 274 с
6. Этапы жц промышленных изделий и системы их автоматизации // StudFiles – сайт. [Электронный ресурс]. – URL: <https://studfile.net/preview/3564600/page:143/>
7. Иванов, А. А. Автоматизация технологических процессов и производств: Учебное пособие / А. А. Иванов. — М.: Форум, 2016. — 224 с.
8. Вдовенко, Л. А. Информационная система предприятия / Л. А. Вдовенко. - М.: Вузовский учебник, Инфра-М, 2010. - 240 с.
9. Криницкий, Н. А. Автоматизированные информационные системы / Н. А. Криницкий, Г. А. Миронов, Г. Д. Фролов. - М.: Наука, 2009. - 382 с.

ЛАБОРАТОРНАЯ РАБОТА №2. ОБЩИЕ СВЕДЕНИЯ ОБ УПРАВЛЕНИИ ПРОЕКТАМИ

1. ЦЕЛЬ РАБОТЫ

Целью данной работы является получение общих теоретических знаний об управлении проектами и практических навыков их реализации в системе Битрикс 24.

2. ТЕОРЕТИЧЕСКОЕ ПОЛОЖЕНИЕ

2.1 Общие понятия управления проектами

Проект – это совокупность процессов и мероприятий, ограниченных заданными временными периодами, внутренними и внешними ресурсами, которые направлены на осуществление поставленных целей или удовлетворение конкретно возникших потребностей.

Соответственно этому определению управление проектом – это деятельность по организации, планированию, контролю и согласованию методов, инструментов и членов команды, направленная на удовлетворение существующих требований и поставленных целей в рамках проекта.

Каждый проект характеризуется наличием одной и более целей. При этом в качестве целей могут восприниматься не только итоговые результаты проекта, но и пути их достижения, используемые методы и технологии. В рамках управления проектами должна обеспечиваться логическая связь применяемых областей знаний и процессов, протекающих в проекте.

Управление проектом рассматривается с точки зрения системного подхода. Несмотря на всё различие и многообразие классификации видов проектов, для управления каждого из них характерно наличие применяемой методологии, команды, последовательности обязательных операций и оценки полученных результатов.

В качестве оценки полученных результатов как правило выступает достижение или не достижение поставленных целей, сроки выполнения задач, ресурсные затраты.

Особенности методологий и операций, присущих управлению проектами будут рассмотрены в разделах ниже.

2.2 Процессы управления проектами

Управление проектами характеризуется наличием ряда обязательных процессов и процедур, сформированных в следующие группы:

- Инициализация;
- Планирование;
- Исполнение;
- Мониторинг и контроль;
- Завершение.



Рисунок 1 – Структура взаимодействия групп процессов управления проектами

Инициализация. Группа процессов при помощи которых осуществляется санкционирование и запуск проекта или новой стадии его жизненного цикла.

Под инициализацией обычно понимают следующие процессы: разработка концепции проекта, рассмотрение, утверждение или отклонение этой концепции, принятие ключевых решений о запуске проекта (выделение ресурсов для запуска проекта, назначение руководителя проекта).

Планирование. Процессы планирования направлены на определение и формирование целей проекта, способов их достижения, а также критериев оценки результата проекта. Одним из основных результатов процессов планирования является создание плана проекта. Планирование проекта характеризуется непре-

рывностью своих процессов, поскольку на любом этапе жизненного цикла проекта, предполагается возможность внесения, как незначительных корректировок, так и серьезных изменений в ранее сформированные планы.

Планирование проекта характеризуется многофункциональностью и многопроцедурностью и может включать в себя ряд следующих операций, затрагивающих разные функциональные области проекта: планирование целей и задач проекта, календарное планирование, планирование финансирования проекта, планирование управления рисками и другие.

Исполнение. Группа процессов управления проектом, отвечающая за осуществление плана проекта. В рамках процессов исполнения происходит организация выполнения задач и необходимых работ, а также координация закреплённых исполнителей. К процессам исполнения проекта относятся: распределение обязанностей исполнителей, создание системы учёта и контроля выполнения работ, организация контроля качества, управление изменениями проектами, обеспечение взаимодействия и координации команды и другие.

Мониторинг и контроль. Данная группа процессов призвана осуществлять сравнение фактических и плановых показателей проекта, выполнять анализ текущего состояния проекта, возникших отклонений (если есть) и их причин, разрабатывать и производить оценку возможных альтернативных решений, на основе которых можно внести корректировки и изменения в проект.

Завершение. Процессы формального и фактического завершения проекта. К процессам завершения проекта можно отнести: предоставление итоговых результатов проекта заказчику, формирование итоговой отчётной документации, формирование финансовой оценки проекта, завершение всех работ по выполнению проекта и другие.

2.3 Методологии и стандарты управления проектами

Методология управления проектами – это совокупность научных подходов, методов и правил, способов постановки и решения задач, с помощью которой обеспечивается процесс исследования управления проектами.

Список методологий управления проектами довольно обширен и многообразен. Он включает в себя и так называемые общие методы менеджмента (планирование, оценка эффективности), и узкоспециализированные методы, характеризующие управление проектами, как отдельную, самостоятельную дисциплину. Среди таких методов можно выделить сетевое планирование, анализ критического пути, рациональное управление (Lean) и многие другие.

Методологии управления проектами определяются на основе стандартов.

В мировой практике принято выделять следующие типы стандартов управления проектами:

- *Международные*: вид стандартов, которые обрели статус международного значения, в процессе своего развития или которые используются к применению в большинстве стран мира.

- *Национальные*: специально созданные или получившие свой статус в процессе развития стандарты, применяемые внутри одной страны.

- *Общественные*: разработанные специалистами разных отраслей стандарты, предназначенные для общественного применения.

- *Частные*: стандарты, разработанные и используемые на частной основе. Как правило, применяются для спецпроектов.

- *Корпоративные*: стандарты, разработанные и применяемые в рамках одного предприятия или объединённой группы предприятий.

Основополагающими можно считать международные стандарты. Они имеют обширную распространённость и признание. Такие стандарты представляют собой комплексные системы, которые помимо основного набора методов и правил могут включать в себя такие элементы как аудит, тестирование и другие.

К наиболее популярными международным стандартам относятся:

- 1) *PMBOK (Project Management Body of Knowledge)*. Данный стандарт является олицетворением процессного подхода к управлению проектами. В стандарте содержатся основные подходы и принципы проектного менеджмента, которые можно использовать для большинства современных проектов. Процессное проек-

тирования предполагает наличие чётких формальных входных данных, а также подразделение процессов в соответствии с направленностью областей исследований управления проектами.

2) IPMA Competence Baseline. Данный стандарт является международным документом, содержащим основные требования компетентности, которым должен соответствовать менеджер проекта. Также в этом стандарте определены критерии оценки менеджеров, их мастерства и профессионализма.

3) ISO 10006. Этот стандарт является основным документом из перечня «Управления качеством и обеспечение качества». ISO 006 заостряет внимание именно на процессах проектирования и их методах контроля, а не на конечном результате проекта.

4) OPM3 (Organization Project Management Maturity Model). В данном стандарте содержатся методы и критерии оценки зрелости организации по управлению проектами. Зрелость организации подразумевает способность организации эффективно отбирать, управлять и выпускать проекты, в соответствии со стратегическими целями компании.

OPM3 включает в себя три основных взаимосвязанных элемента:

- *Знание*. Информация об успешных практиках по управлению проектами;
- *Оценка*. Инструменты оценки уровня зрелости управления проектами предприятия;
- *Улучшение*. Инструменты эффективного создания и изменения схем развития проекта.

2.4 Система управления проектами Битрикс 24

Существует огромное множество систем и различных программных обеспечений по управлению проектами. Они включают в себя разный набор инструментов и решений, позволяющий в той или иной степени вести пользователям деятельность по управлению проектами. Среди наиболее известных систем и программ можно выделить следующие: *Microsoft Project*, *Jira*, *YouGile*, *Asana*, *Wrike* и многие другие.

Рынок отечественных разработок в области управления проектами также достаточно обширен и многообразен. В качестве наиболее востребованной, многофункциональной и доступной

можно выделить комплексную систему по управлению предприятием Битрикс 24. Данная система представляет собой облачный сервис, что позволяет вести совместную работу командой, с возможностью дистанционного участия практически с любого удобного устройства пользователя (система поддерживается Windows, MacOS, Linux, iOS, Android).

Битрикс 24 совмещает в себе набор классических инструментов по управлению предприятием и элементы социальной сети, что даёт возможность легко обеспечить взаимодействие и координирование между всеми участниками команды.

Подробное рассмотрение возможностей Битрикс 24 следует начать со стартовой станицы, которая выглядит следующим образом:

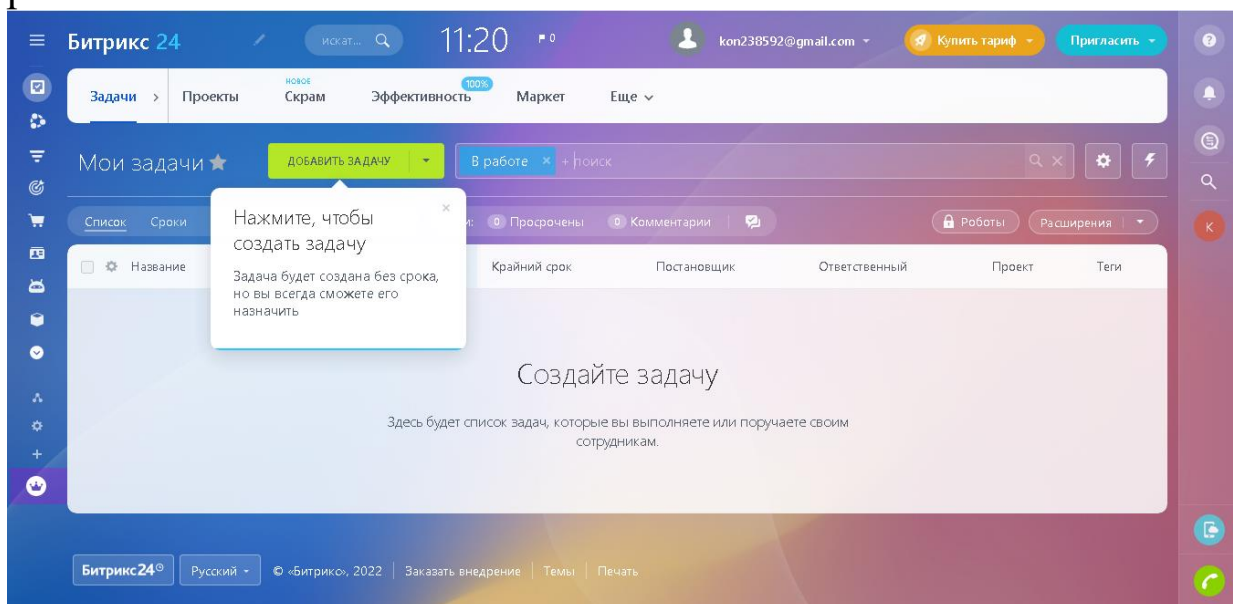


Рисунок 2 – Стартовая страница Битрикс 24

Меню инструментов Битрикс 24 располагается в левой части экрана в виде вертикальной последовательности значков и включает в себя:

1. Совместная работа, в которой располагаются все внутренние средства взаимодействия пользователей системы (почта, звонки и чаты, группы, новости, ведение онлайн документации и др.);
2. Задачи и проекты;
3. CRM;
4. Маркетинг;

5. Конструктор сайтов и онлайн-магазинов;
6. Структура компании;
7. Роботизация бизнеса;
8. Встроенные магазин приложений;
9. Раздел интеграции продукции;
10. Остальные дополнительные возможности системы.

Обширность возможностей Битрикс 24 может удовлетворить потребности в области управления практически любой компании. А разнообразие инструментов и наличие современных элементов социального взаимодействия отлично подойдут для предприятий малого и среднего бизнеса.

Отдельно следует выделить возможности Битрикс 24 в рамках изучаемой тематики данной работы – управлении проектами.

Как уже было сказано выше в Битрикс 24 существует раздел *Задачи и проекты* при помощи которого можно создавать и работать над проектом, создавать и работать над задачами, как в рамках проекта, так и независимо.

Создание нового проекта начинается с определение его названия, добавления описания и выбора дизайнерских решений оформления (Рисунок 3).

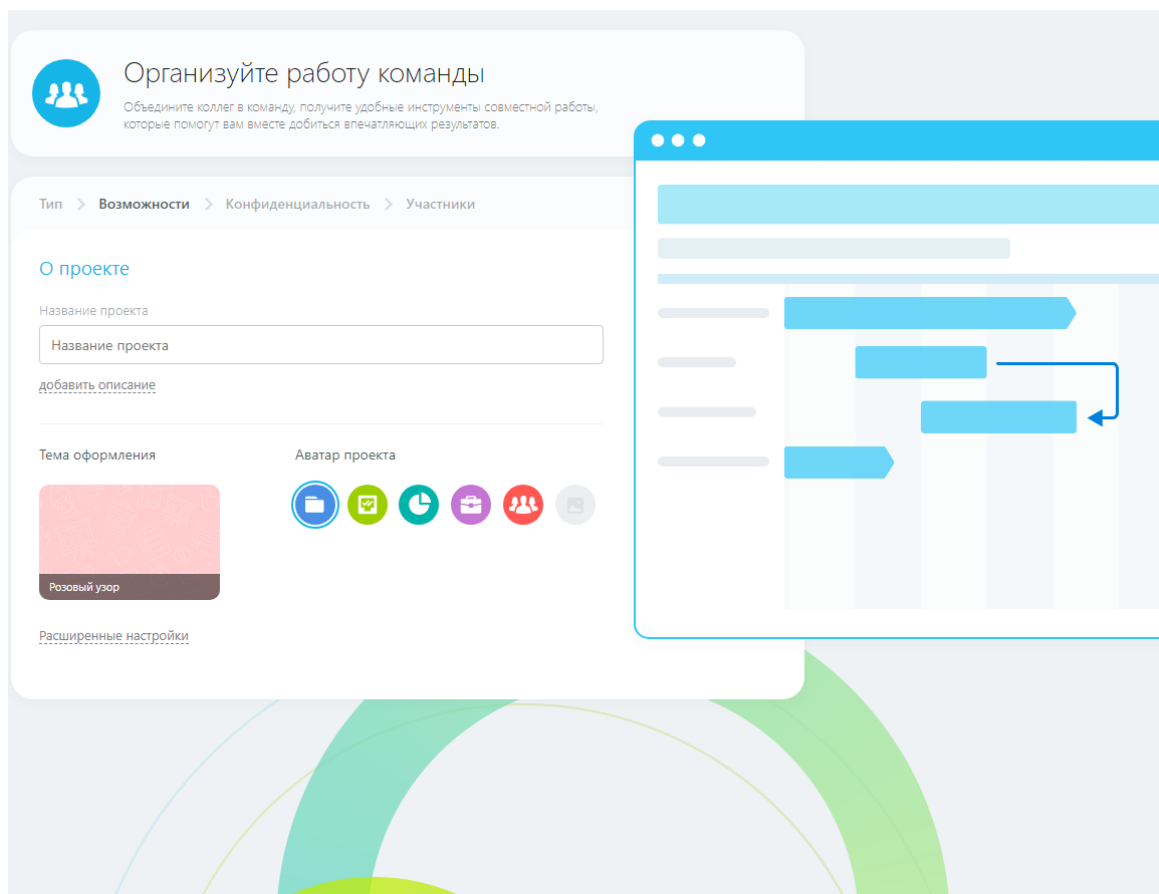


Рисунок 3 – Создание нового проекта

Здесь же можно произвести дополнительные настройки нового проекта. Можно задать сроки проекта, определить тип проекта, уровень доступа, инструменты проекта (задачи, календарь, чат, сообщения и другие):

Параметры проекта

Срок

 12

—










 12

Теги

+ Добавить тег

Возможности

Инструменты в проекте ?

- | | |
|---|---|
| <input checked="" type="checkbox"/> Задачи  | <input checked="" type="checkbox"/> Календарь  |
| <input checked="" type="checkbox"/> Диск  | <input checked="" type="checkbox"/> Чат  |
| <input checked="" type="checkbox"/> Сообщения  | <input type="checkbox"/> Фотогалерея  |
| <input type="checkbox"/> Wiki  | <input checked="" type="checkbox"/> Маркет  |
| <input checked="" type="checkbox"/> База знаний  | |

Тип проекта

- ☐ Для публикации

Внешние пользователи

- ☐ Разрешить доступ внешним пользователям


Рисунок 4 – Расширенные настройки нового проекта

На следующем этапе создания предоставляется возможность определить конфиденциальность проекта, то есть определить уровень открытости информации проекта и возможности к нему присоединения:


Тип > Возможности > **Конфиденциальность** > Участники

Уровень конфиденциальности


Конфиденциальность влияет на то, кто видит информацию внутри команды и как новые участники могут присоединиться к команде.



Открытый
Команда видна всем сотрудникам, все могут зайти, посмотреть информацию и свободно присоединиться к ней



Закрытый
Команда видна в списке всем сотрудникам, доступ есть только у участников, присоединиться к ней можно по запросу



Секретный
Команда никому не видна в списке, присоединиться к ней можно только по запросу

Рисунок 5 – Определение конфиденциальность нового проекта

Заключительным шагом создания проекта является назначение руководителя проекта и формирование команды проекта (Рисунок 6).

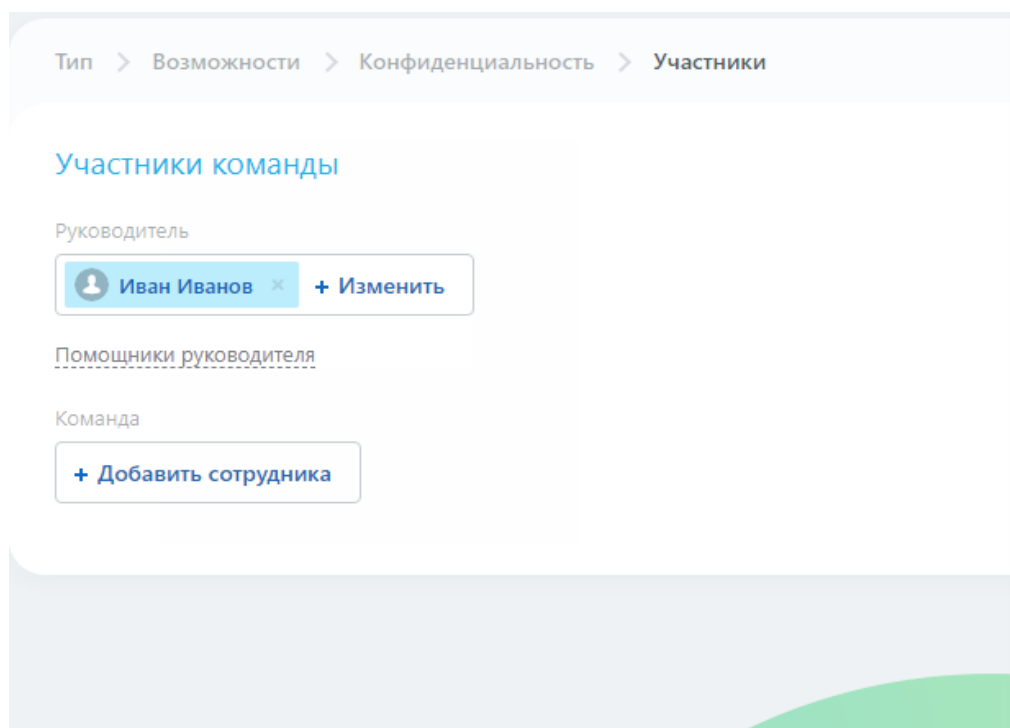


Рисунок 6 – Определение участников нового проекта

В разделе *Проекты и задачи* можно ознакомиться со всей актуальной информацией о проекте, внести необходимые изменения или исправить его статус.

Также в этом же разделе можно создать задачу (Рисунок 7).

Новая задача

Введите название задачи

☐ Это важная задача

файл Создать документ Ответить человеку Цитата Чек-лист в чек-лист

Ответственный: Иван Иванов + Добавить еще

Постановщик: Соисполнители Наблюдатели

Крайний срок: Планирование сроков Ещё

Результат задачи: ☐ Обязательный отчет при завершении задачи

Дополнительно (Проект, Учет времени, Напоминать, Повторять, Гант, CRM, Подзадача, Теги, Поля)

Проект: + Добавить Создать проект

Учет времени: ☐ Время для выполнения задачи

Напоминать о задаче: + Добавить напоминание сообщением или по e-mail

Повторять задачу: ☐ Сделать задачу регулярной

Рисунок 7 – Создание новой задачи

В рамках создания новой задачи необходимо указать её название, сформулировать описание (при необходимости), назначить ответственные лица, сроки задачи и указать обязательность формирования отчёта при завершения задачи.

К дополнительным возможностям относятся:

- Причастность к проекту;
- Задание учёта времени;
- Установление напоминание о задаче, сроков напоминаний и каким способом это будет сделано (системные сообщения, отправка уведомлений на почту);
- Повтор задачи (сделать задачу цикличной);
- Отметить задачу на диаграмме Ганта;
- Добавить CRM элементы;
- Сделать подзадачей другой задачи;
- Указать теги и дополнительные поля к заполнению.

Все созданные задачи отображаются в *списке задач*, где кратко представлена вся основная информация о задачах и статусе участия пользователя в них:

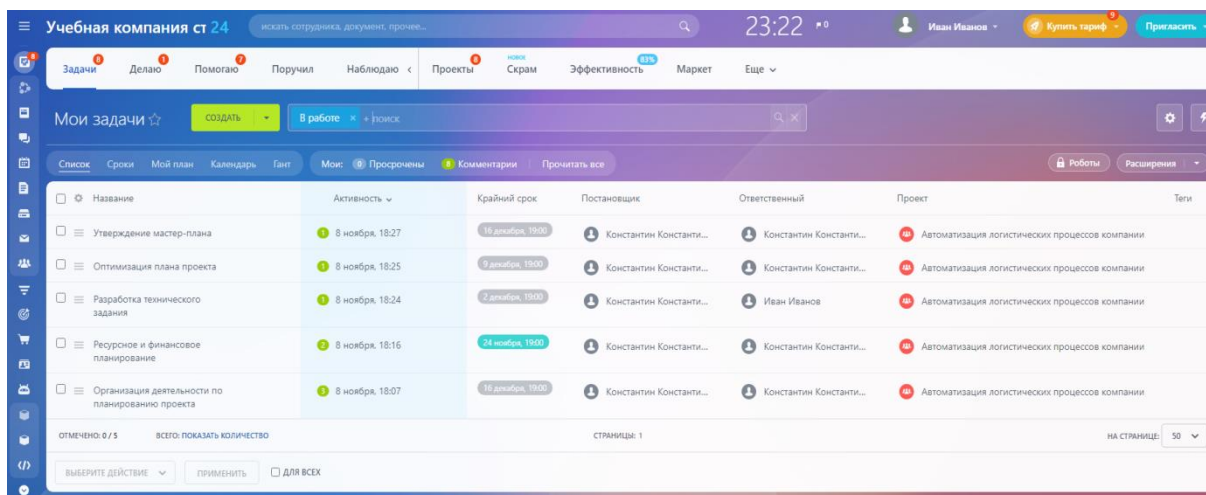


Рисунок 8 – Список задач

Также в Битрикс 24 существуют отдельные вкладки, где указываются: *сроки задач* (Рисунок 9), *личный план пользователя* (Рисунок 10), *календарь* (Рисунок 11) и *диаграмма Ганта* (Рисунок 12).

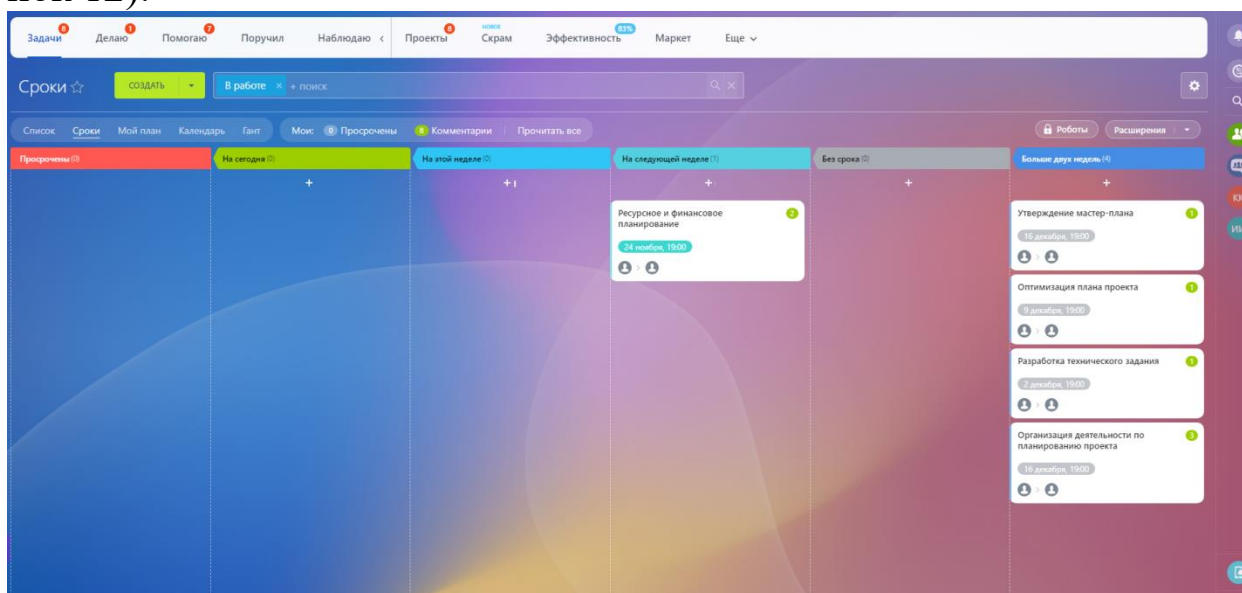


Рисунок 9 – Установленные сроки

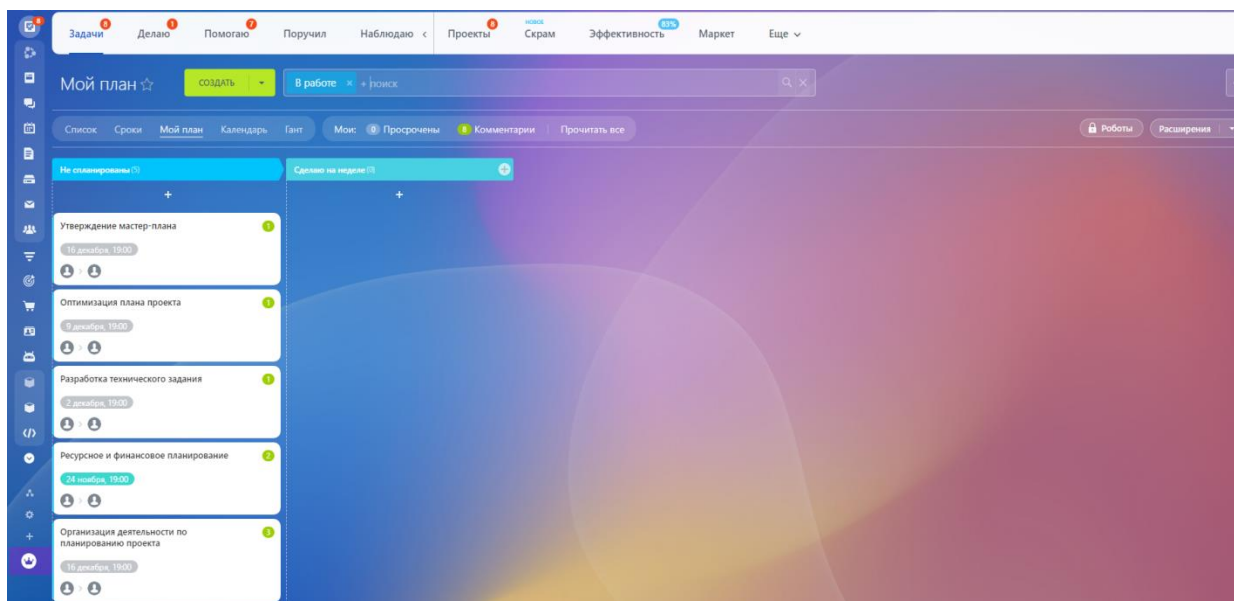


Рисунок 10 – План пользователя

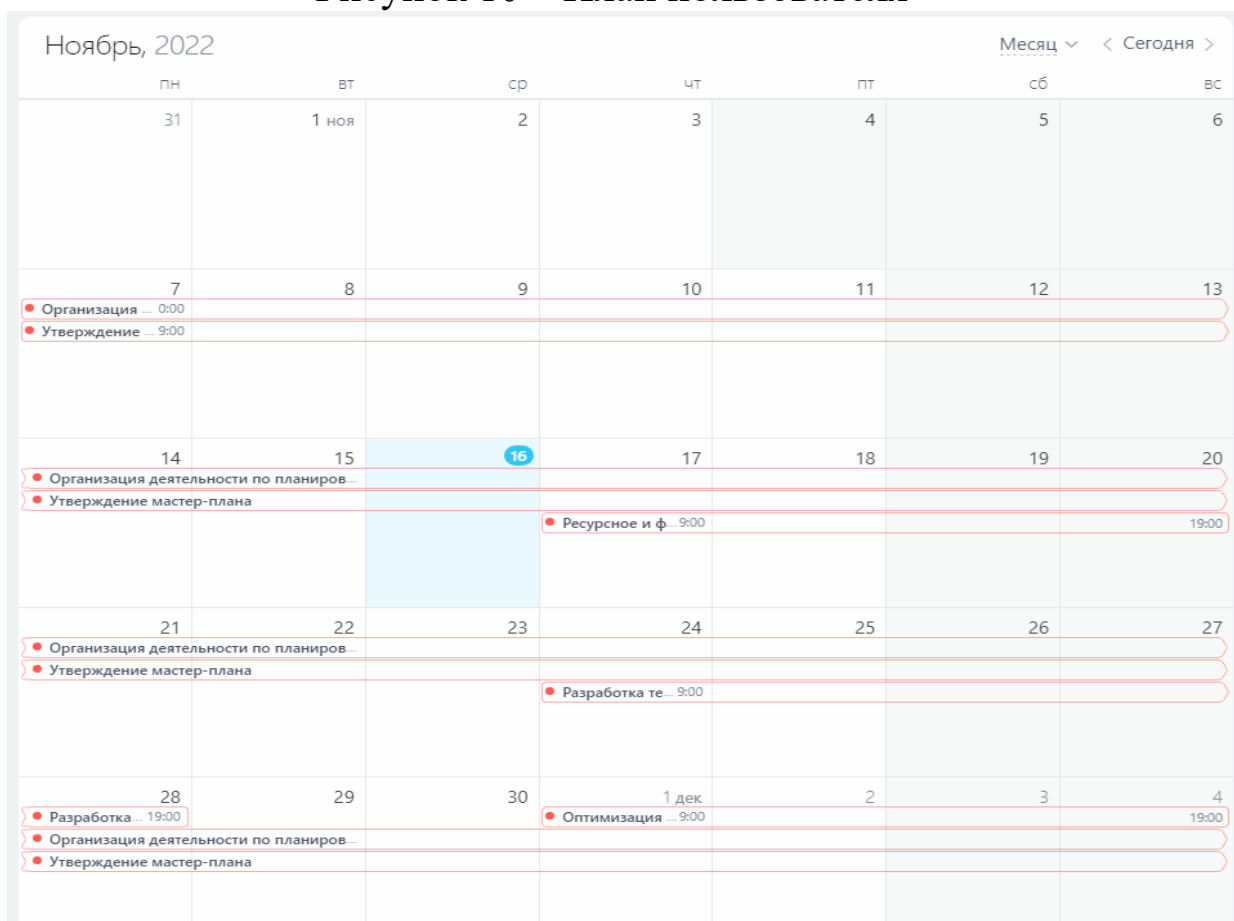


Рисунок 11 – Календарь пользователя

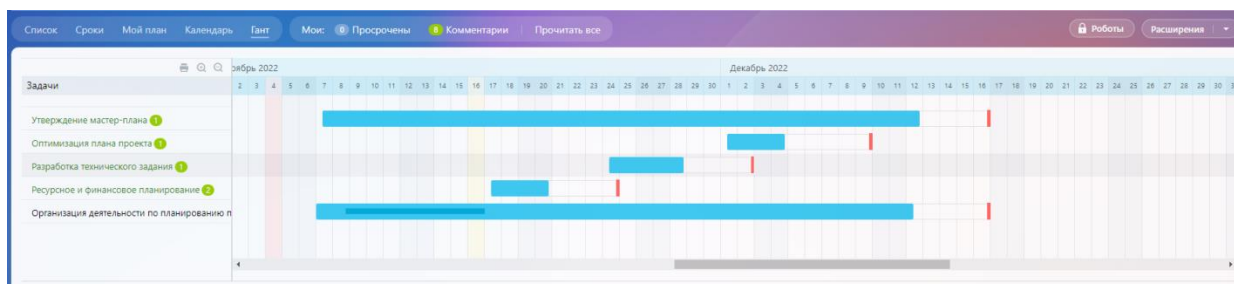


Рисунок 12 – Диаграмма Ганта

Диаграмма Ганта – это график работ и задач проекта, включающий в себя указание их этапов, сроков, пояснения и ответственных за их выполнение. Основой диаграммы Ганта является список поставленных задач (по вертикальной оси) и сроки выполнения этих задач (по горизонтальной оси).

Управление проектами в Битрикс 24 предоставляет возможность ознакомиться с *эффективностью* работы по выполнению поставленных задач, как лично для пользователя, так и в рамках определенного проекта (Рисунок 13).

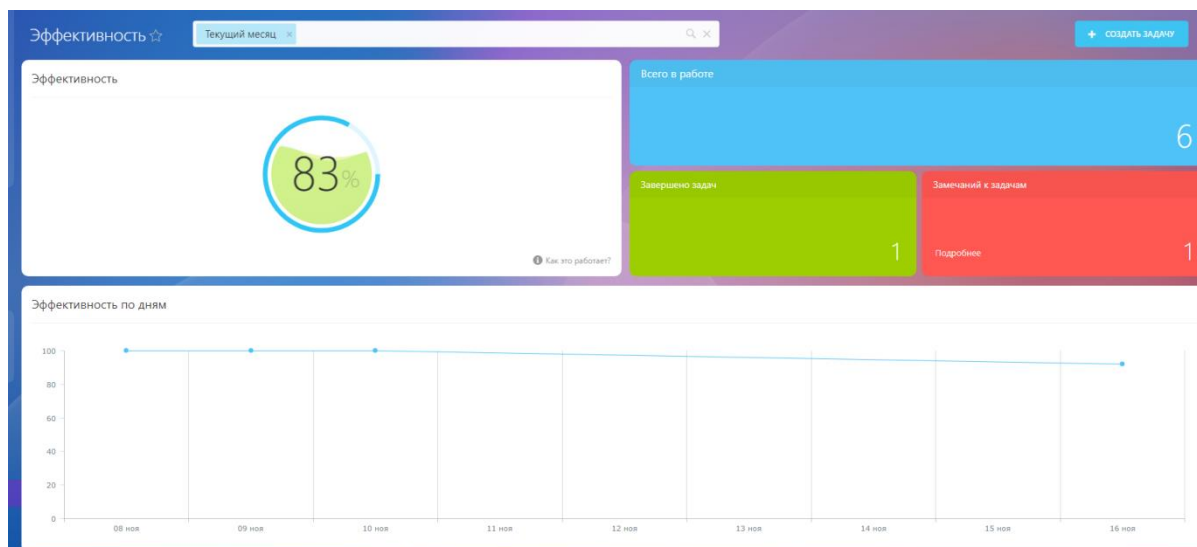


Рисунок 13 – Эффективность выполнения задач

Эффективность рассчитывается на основе суммарного времени работы в системы (или в рамках одного проекта), суммарного количества поставленных и выполненных задач (или в рамках одного проекта), а также сроков их выполнения.

На любой стадии выполнения проекта и задачи можно пригласить новых членов команды на соответствующей вкладке *Пригласить*:

Рисунок 14 – Приглашение новых членов команды

В Битрикс 24 реализовано множество вариантов приглашения новых пользователей. Это и возможность перехода по «быстрой ссылке», и отправка письменного приглашения на почтовый ящик, массовая рассылка приглашений, приглашение с привязкой к отдельному отделу и проекту и многие другие варианты.

Для удобной работы внутри команды Битрикс 24 предоставляет возможность создание разных видов документов онлайн, их редактирование, выгрузку и загрузку уже созданных документов:

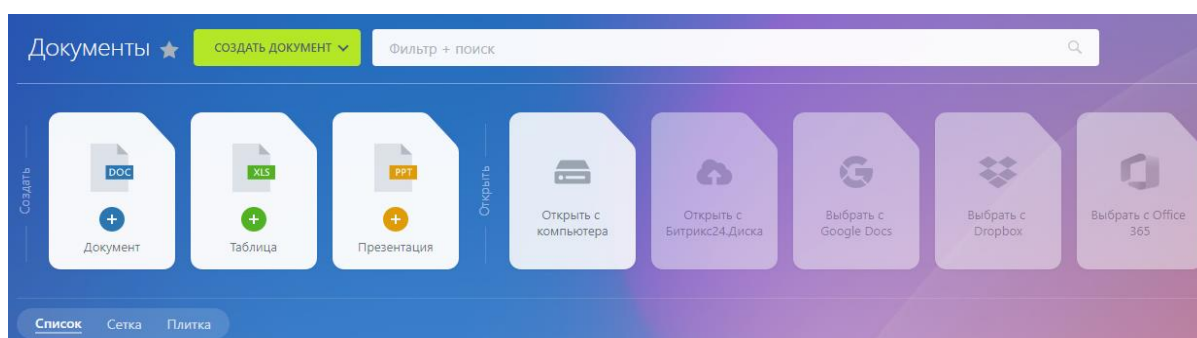


Рисунок 15 – Работа с документами онлайн

По завершению работы над проектом или задачей в Битрикс 24 можно сформировать *отчёт*. В базе Битрикс 24 хранится уже набор стандартных отчётов, однако, реализована возможность

сформировать и в любой другой удобной пользователю форме (Рисунок 16).

Занятость по проектам ☆

ВЕРНУТЬСЯ К СПИСКУ ОТЧЕТОВ

И	Ответственный	Проект: Название	Затраченное время (устаревшее) (Сумма)	В статусе: В работе (Сумма)
1	Иван Иванов	Автоматизация логистических процессов компании	0:00	0
	Константин Константинов		0:00	0
	Константин Константинов	Автоматизация логистических процессов компании	0:00	1
Всего:				
	—	—	0:00	1

Фильтр

- ☒ Мои задачи
- ☐ Задачи подчиненных
- ☐ Задачи из групп

Отчет за период: этот месяц

Проект: "равно"

Применить Отмена

Рисунок 16 – Формирование отчётов

3. ПРАКТИЧЕСКАЯ ЧАСТЬ

Прежде чем начать работу с Битрикс 24 необходимо предварительно определиться с концептуальной частью проекта. Для этого нужно выбрать направленность проекта, сформулировать тему, разработать цели и задачи проекта. После этого также необходимо произвести структурную декомпозицию работ проекта, на основе которой сформировать перечень задач и работ проекта с сроками их выполнения.

В качестве примера будет использован проект со следующей тематикой: «Автоматизация логистических процессов компании».

Целями проекта является повышение эффективности работы логистического подразделения компании путём частичного автоматизирования выполнения части их функциональных задач.

Поскольку управление проектом характеризуется наличием ряда обязательных операций, проведём на их основе декомпозицию задач и работ проекта. Для стандартизации примера будут использованы обобщённые задачи (Рисунок 17).

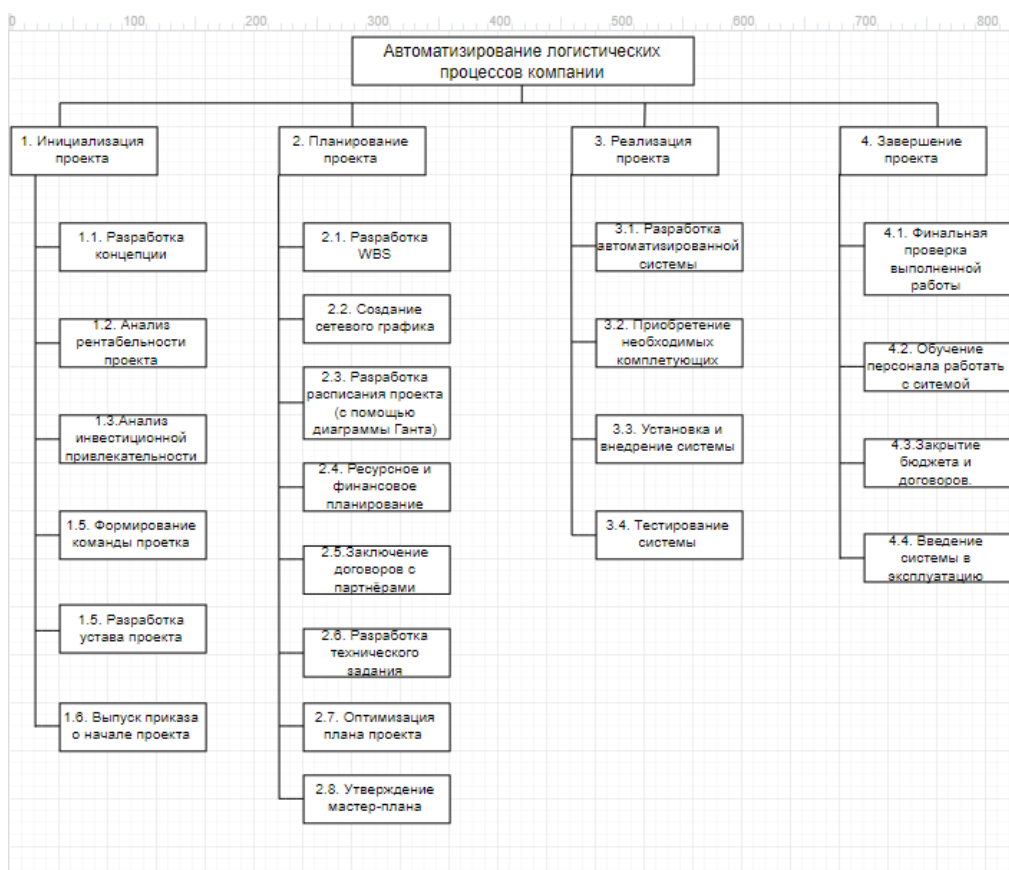


Рисунок 17 – Структурная декомпозиция процессов проекта

На основе произведённой декомпозиции работ проекта составим их перечень и определим сроки выполнения работ.

Таблица 1 – Перечень работ проекта

№ п/п	Операция	Предшествующая операция	Длительность, дней
1	Инициализация проекта	-	21
2	Разработка WBS	1	2
3	Создание сетевого графика	2	2
4	Разработка расписания проекта	3	2
5	Ресурсное и финансовое планирование	4	2
6	Заключение договоров с партнёрами	5	2
7	Разработка технического задания	6	2
8	Оптимизация плана проекта	7	2

9	Утверждение проекта	8	2
10	Подготовка к реализации	9	2
11	Разработка автоматизированной системы	10	21
12	Приобретение комплектующих	11	4
13	Установка и настройка технического оборудования	12	7
14	Установка и настройка ПО	12	7
15	Тестирование системы	13, 14	4
16	Финальная проверка работ	15	1
17	Обучение персонала	16	10
18	Переход на новую систему	16	10
19	Заккрытие бюджета и договоров	17,18	2
20	Ввод системы в эксплуатацию	19	2

Поскольку эта таблица содержит предварительные данные изменять задачи и сроки их выполнения можно в процессе выполнения задания.

Имея все необходимые входные данные можно начинать работу с Битрикс 24.

Для этого сначала регистрируемся в данной системе. Сделать это можно по следующей ссылке: <https://www.bitrix24.ru/>

Битрикс24 ← вернуться на сайт

Помогаем бизнесу работать

Битрикс24 – CRM №1 в России

← Исследование рынка CRM

Русский

Регистрация Битрикс24

Используйте для регистрации свой аккаунт в соцсети

VK Odnoklassniki Yandex Mail.ru Telegram

или

Введите e-mail

Регистрируясь, вы подтверждаете, что принимаете [Пользовательское соглашение](#), даете [Поручение и Согласие](#) на обработку персональных данных.

☐ Я соглашаюсь получать рекламные информационные материалы в соответствии с [Соглашением](#)

☐ Я хочу получать приглашения на бесплатные вебинары

Рисунок 18 – Регистрация в Битрикс 24

Теперь можно сформировать команду проекта, пригласив второго члена группы. Для этого отправим ему приглашение на почту:

Приглашение

ОБРАТНАЯ СВЯЗЬ

Приглашение по ссылке

Email или номер телефона

Массовое приглашение

С привязкой к делу и г...

Зарегистрировать

Экстранет-пользователь

Пригласить интегратора

Active Directory

На вашем тарифном плане доступно пользователей неограниченно

Сейчас в вашем Битрикс24 зарегистрировано уже 1

Введите e-mail или телефон

Имя

Фамилия

ivanichka007@gmail.com

Иван

Иванов

Введите e-mail или телеф...

Имя

Фамилия

Введите e-mail или телеф...

Имя

Фамилия

Введите e-mail или телеф...

Имя

Фамилия

Введите e-mail или телеф...

Имя

Фамилия

+ ДОБАВИТЬ ЕЩЕ или Пригласить массово

ПРИГЛАСИТЬ

ЗАКРЫТЬ

Рисунок 19 – Приглашение нового пользователя (Ивана Иванова)

После принятия приглашения, регистрации и авторизации второго пользователя, оставим сообщение в чате команде о присоединение нового члена команды:

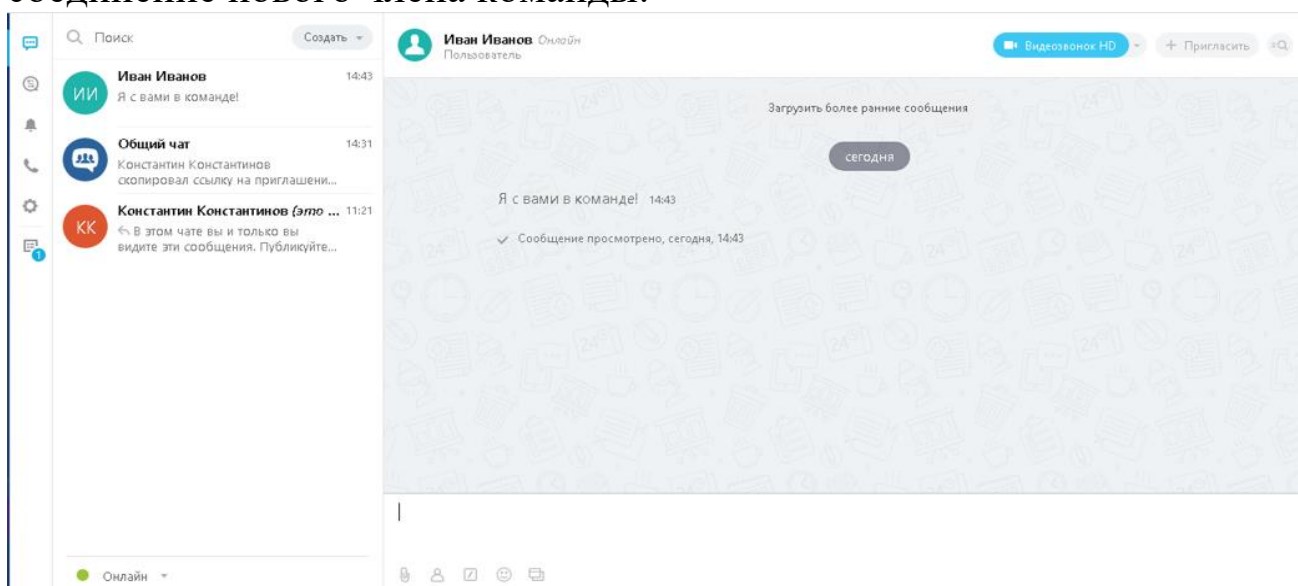


Рисунок 20 – Общий чат команды

Создадим новую независимую задачу для нового члена команды. Обозначим задачу по заполнению профиля пользователя, укажем сроки выполнения, ответственного и исполнителя:

Новая задача

Заполнить профиль

Заполните свой профиль

Файл Создать документ Отметить человека Цитата Чек-лист В чек-лист

Ответственный: Константин Константинов + Добавить еще

Постановщик: Константин Константинов Сменить

Соисполнители: Иван Иванов + Добавить еще

Крайний срок: 10.11.2022 19:00 Планирование сроков Еще

Результат задачи: ☐ Обязательный отчет при завершении задачи

Дополнительно (Проект, Учет времени, Напомнить, Повторять, Гант, CRM, Подзадача, Теги, Поля)

Рисунок 20 – Создание задачи для членов команды

Зайдём в систему от пользователя *Иван Иванов*, ознакомимся с уведомлением (Рисунок 21).

Заполнить профиль

Задача №3 - выполняется, просрочена

Заполните свой профиль

+ добавить чек-лист

Приостановить Завершить Ещё

Комментарии История 5 Времени 00:00:00 Замечания 1

Константин Константинов 6 ноября 17:58
Иван Иванов, вы назначены соисполнителем.

Константин Константинов 10 ноября 07:04
Иван Иванов, задача почти просрочена. Завершите задачу или передвиньте срок в течение суток, иначе эффективность будет снижена.

Константин Константинов 20:51
Иван Иванов, задача просрочена, эффективность снижена. Завершите задачу или передвиньте срок.

Добавить комментарий

Выполняется с 06.11.2022 18:03

Крайний срок: 10.11.2022 19:00

Задача просрочена!

Напоминание: Напомнить

Поставлена: 08.11.2022 17:58

Оценка: Нет оценки

ВИДЕОЗВОНОК HD

Постановщик: Константин Константинов

Ответственный: Константин Константинов

Соисполнители: Иван Иванов

Наблюдатели: наблюдать

Рисунок 21 – Уведомление о новой задаче

Выполним поставленную задачу, заполнив свой профиль:

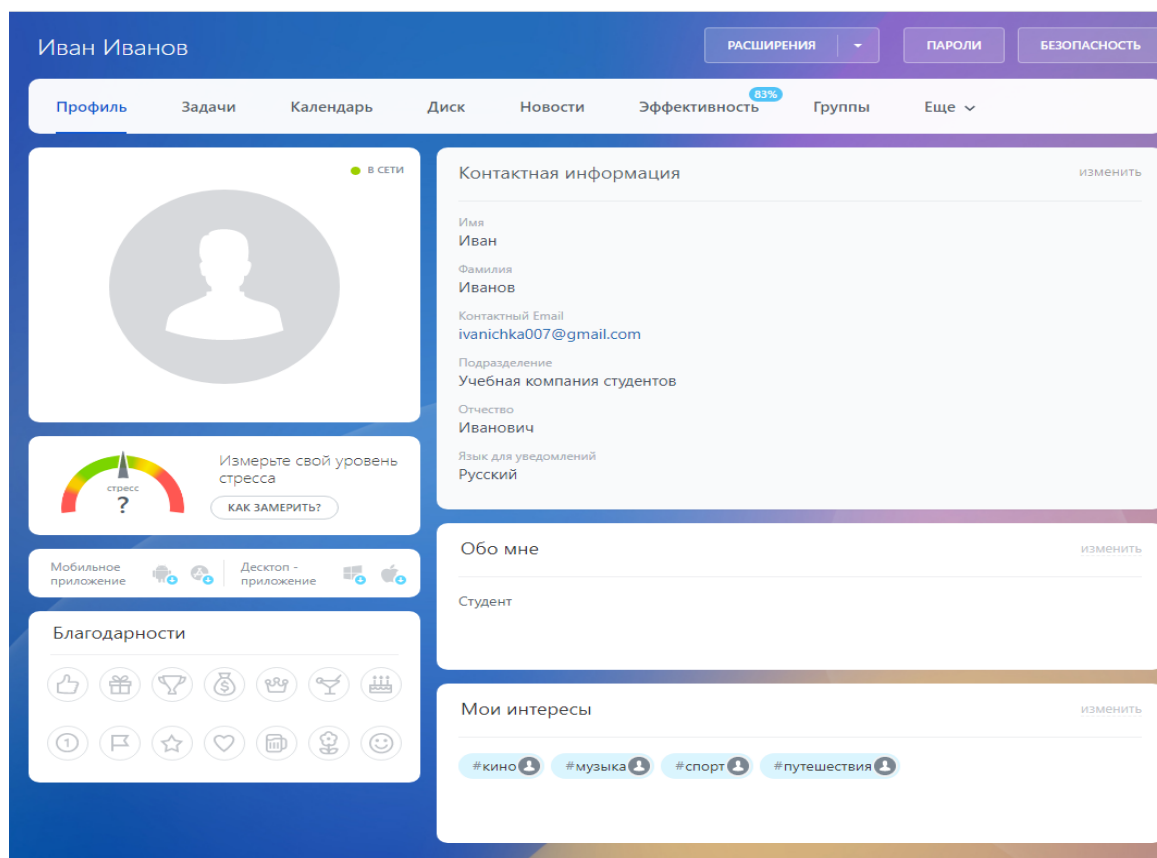


Рисунок 22 – Заполнение профиля пользователя

После выполнения задачи изменим её статус:

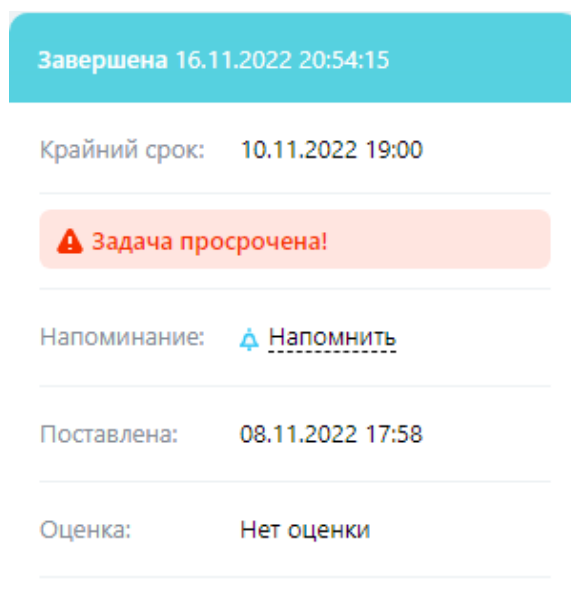


Рисунок 23 – Изменение статуса задачи

Можно обратить внимание, что задача была просрочена, так как были нарушены сроки её выполнения. Это повлияет на эффективность работы пользователя в системе.

Для расширения возможностей нового пользователя предоставим ему *права администратора*. Для этого вернёмся к первому пользователю (*Константин Константинович*) и проведём соответствующую процедуру:

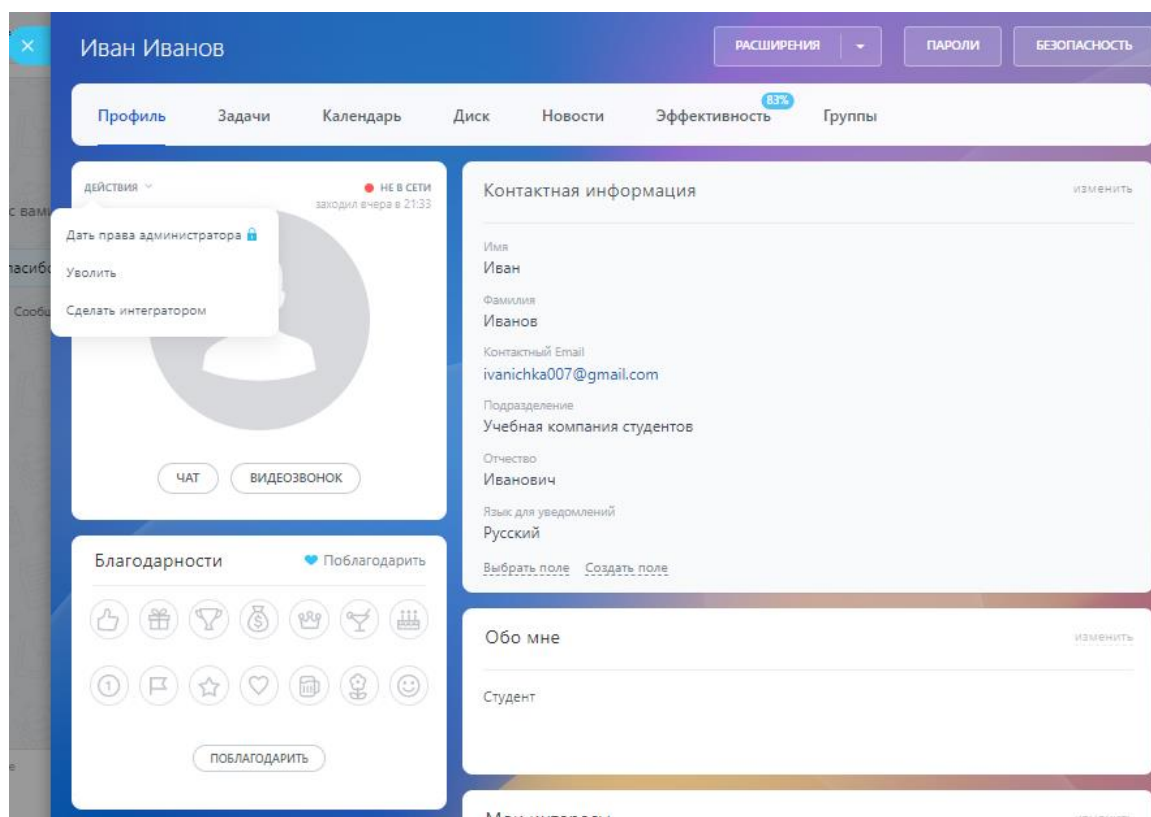


Рисунок 24 – Предоставление прав администратора

Теперь можно начать создание нового проекта (Рисунок 25).

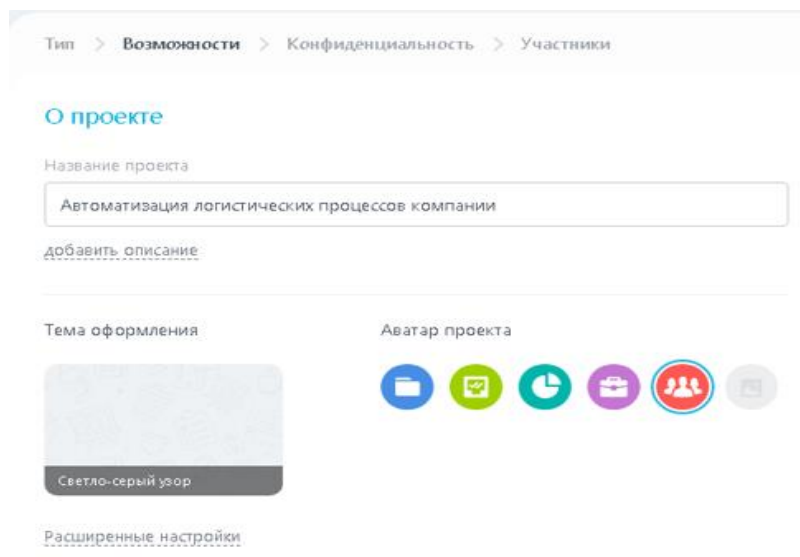


Рисунок 25 – Создание проекта

«Автоматизация логистических процессов компании»

Сделаем проект закрытого типа и добавим к участникам второго пользователя.

После завершения создания проекта он отобразится в списке проектов (Рисунок 26).

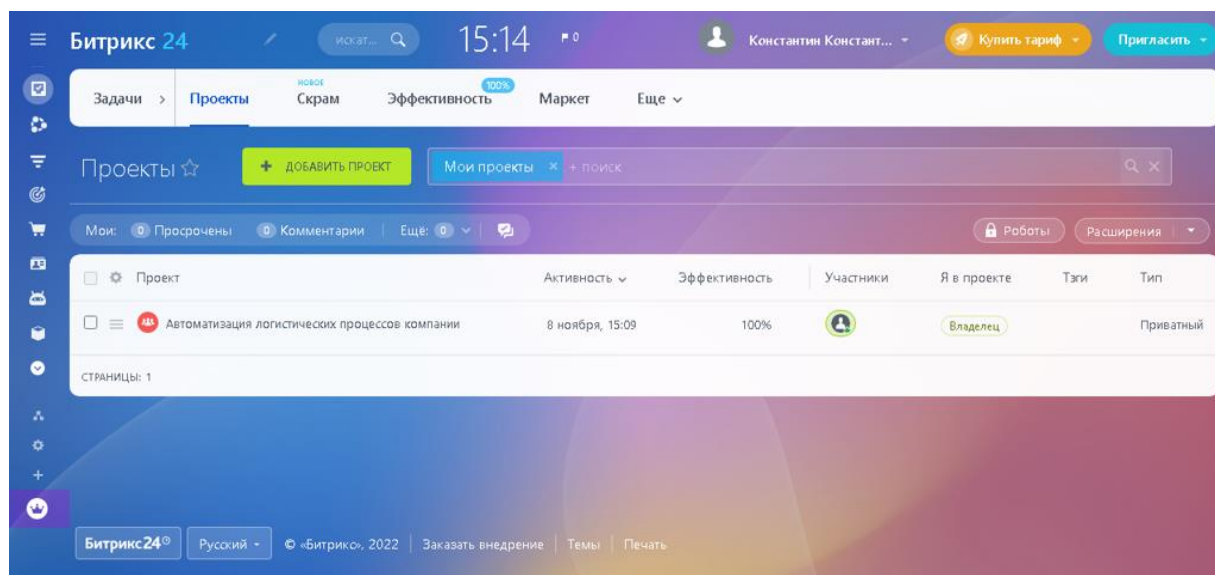


Рисунок 26 – Список проектов

Перейдём к задачам проекта. Детальнее разберём группу задач по планированию проекту (Рисунок 17). Создадим одну крупную задачу *Организация деятельности по планированию проекта*, включающую в себя ряд подзадач.

Рисунок 27 – Создание задачи Организация деятельности по планированию проекта

Рисунок 28 – Дополнительные настройки задачи Организация деятельности по планированию проекта

В настройках задачи укажем сроки её выполнения, ответственных лиц, установим напоминание, добавим в диаграмму Ганта.

К задаче Организация деятельности по планированию проекта создадим следующий ряд подзадач:

- Разработка WBS;

- Создание сетевого графика;
- Разработка расписания проекта;
- Ресурсное и финансовое планирование;
- Заключение и подписание договоров с партнёрами;
- Разработка технического задания;
- Оптимизация плана проекта;
- Утверждение мастер-плана проекта.

Для упрощения примера, сделаем так, что срок выполнения для каждой подзадачи составит два дня. Все задачи выполняются последовательно, начало работ происходит только после завершения предшествующей задачи. Режим работы предприятия – без выходных дней.

Создадим подзадачу *Разработка расписания проекта*:

The screenshot shows a web interface for creating a new task. At the top, there's a header 'Новая задача' (New task) with a settings icon and a 'ШАБЛОНЫ ЗАДАЧ' (Task templates) dropdown. Below this, the task title 'Разработка расписания проекта' (Project scheduling development) is entered. A toolbar contains icons for file, document, person, quote, checklist, and checklist. The main form has several sections: 'Ответственный' (Responsible) with a dropdown showing 'Константин Константинов' and a '+ Добавить еще' (Add more) button; 'Крайний срок' (Deadline) with a date-time picker set to '18.11.2022 19:00' and a 'Планирование сроков' (Schedule planning) button; 'Начать задачу с' (Start task at) with a date-time picker set to '17.11.2022 09:00'; 'Длительность' (Duration) with a numeric input set to '34' and units 'дни' (days), 'часов' (hours), and 'минут' (minutes); 'Завершение' (Completion) with a date-time picker set to '18.11.2022 19:00'. At the bottom, there are checkboxes for permissions: 'Разрешить ответственному менять сроки задачи' (Allow responsible to change task deadlines) is checked, 'Пропустить выходные и праздничные дни' (Skip weekends and holidays) is unchecked, 'Проконтролировать задачу после завершения' (Control task after completion) is checked, 'Добавить себе в избранное' (Add to favorites) is unchecked, and 'Добавить в план рабочего дня' (Add to workday plan) is checked. A 'настроить' (configure) link is next to the second checkbox. A watermark 'Активация Windows' (Windows Activation) is visible at the bottom right.

Рисунок 29 – Создание подзадачи «Разработка расписание проекта»

Как и в главной задаче в настройках укажем сроки её выполнения, ответственных лиц, установим напоминание, добавим в диаграмму Ганта.

Аналогичным способом создадим все остальные подзадачи.

Все созданные задачи можно увидеть в *списке задач* (Рисунок 30). При помощи этого списка можно увидеть основную и

дополнительную информацию о задаче, её статус, сроки, ответственных лиц или же внести нужные нам изменения.

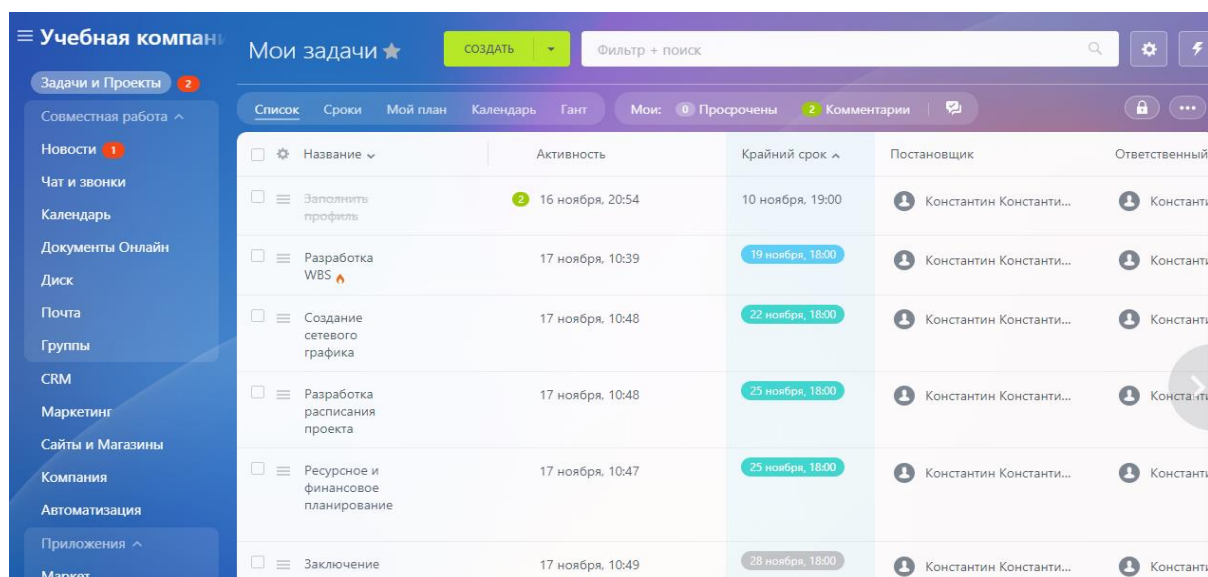


Рисунок 30 – Список задач пользователя

Ознакомиться со сроками и статусами задачи можно также при помощи вкладок *Сроки* (Рисунок 31) или, например, вкладок *Календарь* (Рисунок 32).

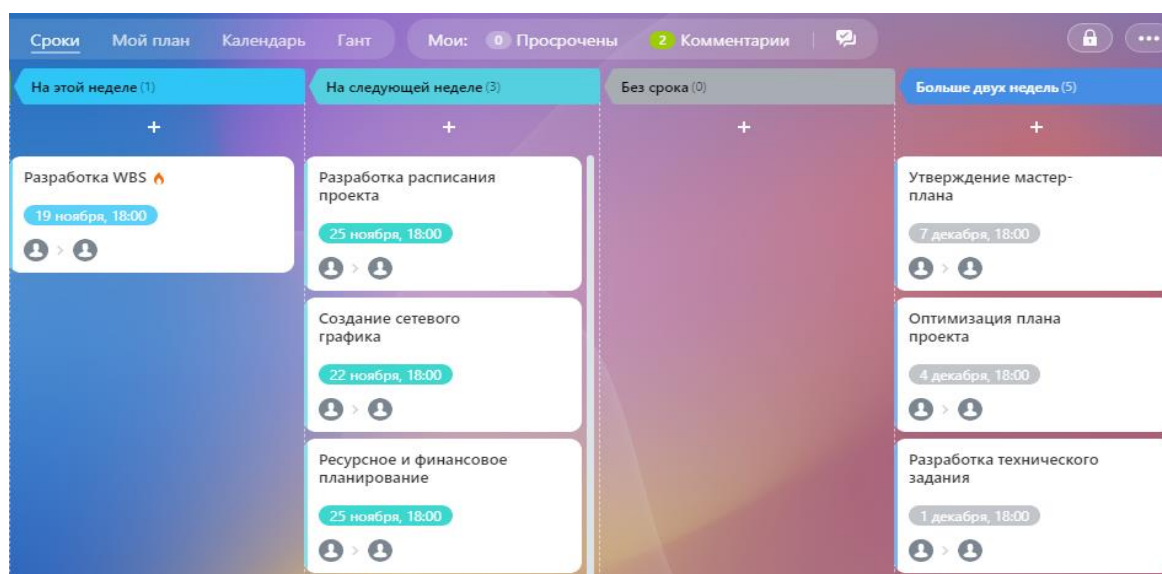


Рисунок 31 – Сроки выполнения задач пользователя

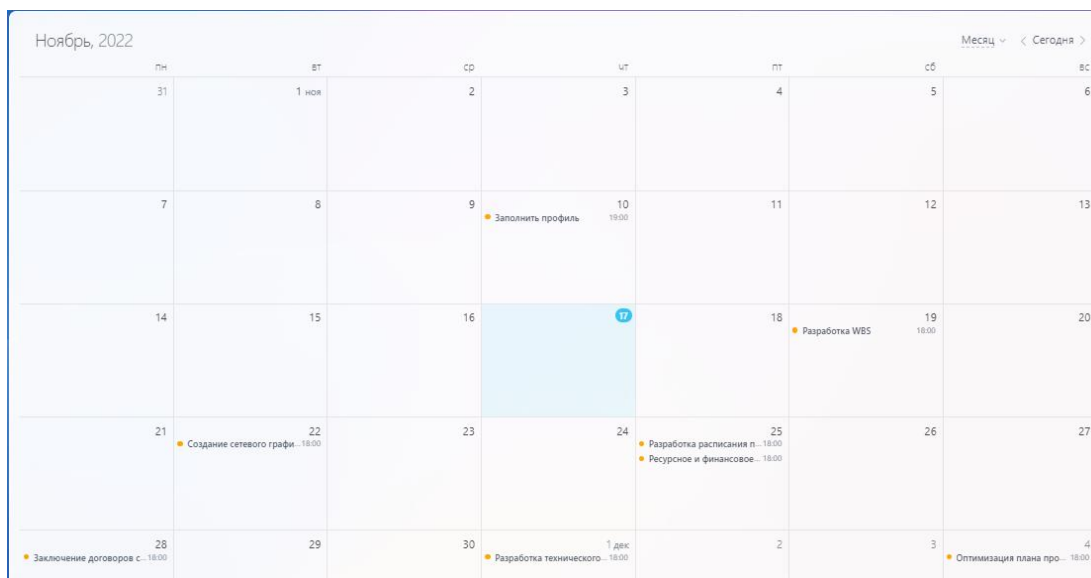


Рисунок 32 – Календарь выполнения задач пользователя

Поскольку в дополнительных настройках всех задач мы выбирали возможность добавления их на диаграмму Ганта, все задачи и подзадачи нашего проекта будут там отображены:

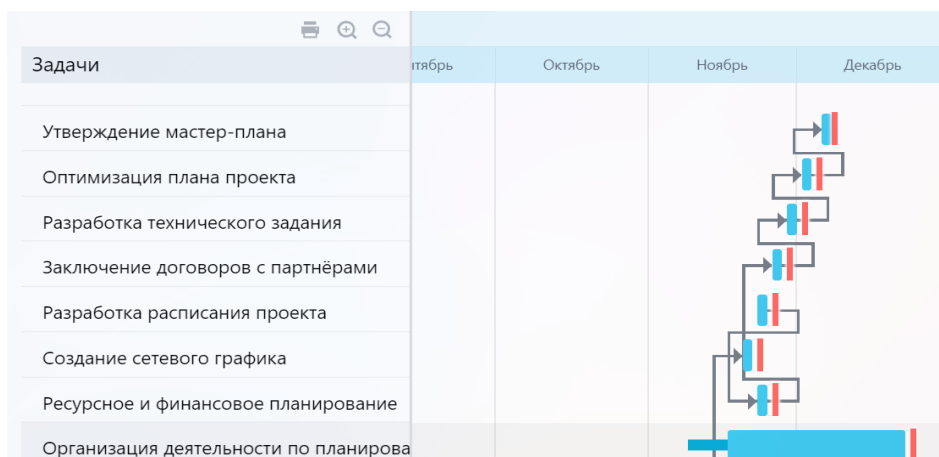


Рисунок 33 – Диаграмма Ганта управляемого проекта

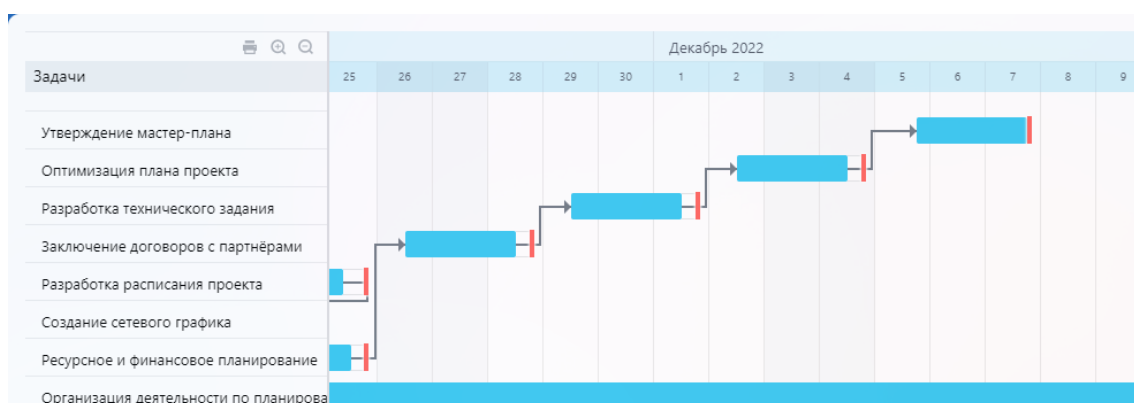


Рисунок 34 – Масштабированный вид диаграмма Ганта управляемого проекта

Изменим статус некоторых задач на *завершённые*. После чего сформируем отчёт о задачах нашего проекта. Для этого можно воспользоваться или шаблоном отчётов или самостоятельно сформировать отчёт нужного нам вида.

Личные отчеты	
Название отчета	Дата создания
Задачи проекта	17.11.2022

Стандартные отчеты	
Название отчета	Дата создания
Занятость по проектам	08.11.2022
Задачи на месяц	08.11.2022
Отчет по эффективности	08.11.2022
Задачи за прошлый месяц	08.11.2022
Ресурсный учёт по задачам	08.11.2022
Ресурсный учёт по исполнителям	08.11.2022

Рисунок 35 – Список доступных шаблонов отчётов

↑ Ответственный	Название	Статус	Важность	Дата создания	Дата начала	Дата завершения	Крайний срок
Константин Константинов	Организация деятельности по планированию проекта	Завершена	Нет	08.11.2022	08.11.2022	17.11.2022	23.12.2022
Константин Константинов	Создание сетевого графика	Завершена	Нет	08.11.2022		17.11.2022	22.11.2022
Константин Константинов	Заключение договоров с партнёрами	Ждет выполнения	Нет	08.11.2022			28.11.2022
Константин Константинов	Заполнить профиль	Завершена	Нет	08.11.2022	08.11.2022	16.11.2022	10.11.2022
Константин Константинов	Разработка расписания проекта	Завершена	Нет	08.11.2022	17.11.2022	17.11.2022	25.11.2022
Константин Константинов	Утверждение мастер-плана	Ждет выполнения	Нет	08.11.2022			07.12.2022
Константин Константинов	Разработка WBS	Завершена	Да	08.11.2022	08.11.2022	17.11.2022	19.11.2022
Константин Константинов	Ресурсное и финансовое планирование	Завершена	Нет	08.11.2022		17.11.2022	25.11.2022

Рисунок 36 – Отчёт о назначенных задачах пользователю за месяц

4. ПРАКТИЧЕСКОЕ ЗАДАНИЕ

1. Студенты делятся на группы по 2 человека. Каждая группа разрабатывает тему проекта в области информационных технологий. Необходимо кратко описать цели и задачи проекта, особенности его реализации. Произвести структурную декомпозицию работ проекта на основе которой сформировать предварительный перечень работ проекта с сроками их выполнения.

2. Каждый студент должен зарегистрироваться в системе Битрикс 24. В системе на группу создаётся один проект: один студент создаёт проект и приглашает второго студента в команду, предоставляя ему равные права и доступ.

3. Необходимо добавить задачи проекта в соответствии с ранее сформированным перечнем работ. Задачи должны быть включены в состав проекта, иметь сроки начала и завершения. Задачи распределяются между членами группами, назначаются ответственные лица, создаются напоминания в виде системных сообщений.

4. График выполнения задач проекта нужно отобразить на диаграмме Ганта.

5. Создать табличный документ в Битрикс 24, где отобразить перечень работ проекта, сформированный в первом пункте.

6. Сформировать отчёт о выполненных задачах и статусе проекта в Битрикс 24.

7. Сформировать отчёт о проделанной работе.

5. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Что такое проект? Понятие управление проектами?

2. Какие группы процессов управления проектами вы знаете?

3. Какие существуют виды стандартов управления проектами? Перечислите известные вам международные стандарты?

4. Перечислите известные вам программы и системы управления проектами?

5. Перечислите основные возможности Битрикс 24?

6. В каком разделе располагаются основные инструменты управления проектами в системе Битрикс 24?

7. Какие уровни конфиденциальности проекта существуют в Битрикс 24?

8. Каким способом можно осуществить приглашение новых пользователей в Битрикс 24?

9. Что такое диаграмма Ганта? Как можно создать или изменить диаграмму Ганта в Битрикс 24?

10. Каким образом можно сформировать отчёт о выполненных задачах и завершённом проекте в Битрикс 24?

СПИСОК РЕКОМЕНДУЕМОЙ ЛИТЕРАТУРЫ

1. Бедердинова, О. И. Автоматизированное управление IT-проектами: учебное пособие / О. И. Бедердинова, Ю. А. Водовозова. – Москва: ИНФРА-М, 2021. – 92 с.

2. Котляр, Е. В. Система управления проектами Канбан / Е. В. Котляр, Е. М. Пушкарева // Бизнес-образование в экономике знаний. – 2020. – № 1(15). – С. 57-59.

3. Вейнберг, Р. Р. Применение стандартов управления проектами в IT-индустрии: PRINCE2 и PMBoK / Р. Р. Вейнберг, Н. А. Моисеев, С. М. Сахарова // Вестник Российского экономического университета имени Г. В. Плеханова. – 2020. – № 1(109). – С. 56-66.

4. Бахтин, И. В. Руководство по выбору лучшего программного обеспечения для управления проектами / И. В. Бахтин // Научный электронный журнал Меридиан. – 2020. – № 7(41). – С. 60-62.

5. Задачи и проекты // Битрикс 24 – сайт. [Электронный ресурс]. – URL: <https://www.bitrix24.ru/features/tasks.php>

ЛАБОРАТОРНАЯ РАБОТА №3. КЛАССИФИКАЦИЯ ПРОЕКТОВ

1. ЦЕЛЬ РАБОТЫ

Целью данной работы является получение общих теоретических знаний об классификации проектов, методологиях проектирования ИС и практических навыков их эффективного выбора.

2. ТЕОРЕТИЧЕСКОЕ ПОЛОЖЕНИЕ

2.1 Классификация проектов

Проект – это организованная совокупность взаимосвязанных процессов целью которых является выполнение поставленных задач, создание принципиально нового, уникального продукта, в рамках имеющихся требований, ресурсных и временных ограничениях. Наличие временных и ресурсных ограничений сводится к тому, что у проекта есть точки старта и завершение проекта. Точкой начала проекта считается момент времени, когда появилась потребность в его создании. Завершение проектом считается, выполнение целей, поставленных в рамках проекта, или наоборот, подтверждение факта невозможности их достижения.

Выбор подхода к управлению и ведению проекта в значительной степени зависит от типа проекта. Классификация основных типов проектов представлена в Таблице 1.

Таблица 1 – Классификация проектов

Классификационные признаки	Типы проектов				
По уровню проекта	Проект	Программа		Система	
По масштабу (размеру) проекта	Малый	Средний		Мегапроект	
По сложности	Простой	Организационно сложный	Технически сложный	Ресурсно сложный	Комплексно сложный
По срокам реализации	Краткосрочный	Средний		Мегапроект	
По требованиям к качеству и способам его обеспечения	Бездефектный	Модульный		Стандартный	
По требованиям к ограниченности ресурсов совокупности проектов	Мультипроект		Монопроект		
По характеру проекта/уровню участников	Международный (совместный)		Отечественный: —государственный —территориальный —местный		
По характеру целевой задачи проекта	Антикризисный		Реформирование/реструктуризация		
	Маркетинговый		Инновационный		
	Образовательный		Чрезвычайный		
По объекту инвестиционной деятельности	Финансовый		Реальный		
	Инвестиционный		Инвестиционный		
По главной причине возникновения проекта	Открывшиеся возможности		Необходимость структурно функциональных преобразований	Реорганизация	
	Чрезвычайная ситуация			Реструктуризация	
				Ренжиниринг	

Определение верного типа классификации проекта довольно сложный процесс. Как правило, проект обладает несколькими классификационными признаками, а соответственно может быть отнесен к различным типам. При этом можно выделить следующие наиболее значимые и часто встречающиеся типы проектов:

По уровню проекта:

Помимо стандартного понимания проекта, то есть типа проекта, соответствующего определению и применяемому к любой общественной сфере и отрасли промышленности, по уровню проекта можно выделить следующие: ***программа, система***. Проекты по разработке программ и систем, в том числе информационных, в последнее время набрали существенную популярность и распространённость. На данный момент существует множество подходов и методик управления таким проектированием, правильное использование которых, способствует получению высокой эффективности разрабатываемых программ и систем.

По масштабу проекта:

Малые проекты – простые проекты, существенно ограниченные объёмами. Как правило, малыми являются проекты по

оптимизации каких-либо существующих процессов или модернизации уже действующих производств. Малые проекты характеризуются простотой организации, проектирования и реализации, а также низкой затратностью.

Средние проекты в сравнении с малыми обладают более детально проработанной организационной и финансовой структурой, а также длительным периодом (2-5 лет) и большими затратами. В реальной повседневной практике средние проекты являются наиболее распространёнными.

Мегапроекты – многоструктурированные разработки, состоящие из нескольких взаимосвязанных проектов разного масштаба, направленных на достижение единой проектной цели. Мегапроекты характеризуются высокой стоимостью, трудоёмкостью и задействованными ресурсами.

По срокам реализации проекта:

Краткосрочные – до 3 лет.

Среднесрочные – от 3 до 5 лет.

Долгосрочные – более 5 лет.

По требованиям к ограниченности ресурсов совокупности проектов:

Монопроект – самостоятельный проект любого типа.

Мультипроект – совокупность проектов, программ или систем, рассматривающихся как единый комплекс, который требует мультипроектного управления. Все мультипроекты состоят из монопроектов.

Помимо выше перечисленных типов проектов можно выделить международный.

Международные проекты характеризуются исключительно важной ролью в экономических, политических и других сферах общества стран-участников проекта. Такие проекты являются особо сложными и обладают высокой стоимостью, а также основываются на принципах взаимодополняемости и взаимовыгодности участников.

В научной литературе выделяются также ***инновационные проекты***. Такие проекты направлены на разработку и создание новых технологий, изделий, систем. Одним из главных отличий инновационных проектов является достижение целей инноваций, то есть получение нематериальных активов. При этом стоит от-

метить, что отсутствие финансовых целей проекта, не исключает выделения на него инвестиций. Поэтому разделение проектов на **инвестиционные** и инновационные принято считать достаточно условным.

Не смотря на сложность процесса классификации проекта, он считается очень важным и неотъемлемым на ранних стадиях создания проекта. При помощи правильно определённого типа проекта можно определить его перспективность и реализуемость, сферу применения и значимость, что способствует постановки выполнимых целей и сроков проекта, а также подбора правильной стратегии его управления.

2.2 Модели и методологии проектирования ИС

Как уже было сказано в предыдущем разделе, рассмотреть проект можно с точки зрения создания программного продукта или информационной системы. К ведению такого проекта применяется понятие проектирования информационной системы (ИС).

Проектирование ИС – это процесс, в ходе которого принимаются проектно-конструкторские решения, способствующие созданию описания ИС, которое будет удовлетворять установленным требованиям. Проектирование ИС производится на основе моделей и методологий.

Модель информационной системы представляет собой описание ИС, выполненное в формализованном виде. Модели отвечают за конкретные аспекты системы, определяют используемый набор документации, диаграмм, методы визуализации и др., необходимого формата, используемые на протяжении всего жизненного цикла системы.

Методология проектирования информационных систем представляет собой набор методов, приёмов, способов организации, технических и программных средств по принятию проектно-конструкторских решений для создания необходимой системы, характеризующийся многократным успешным применением для систем подобного типа.

В настоящее время применяются следующие модели и методологии проектирования ИС:

Каскадная модель (или модель водопада, Waterfall) – в основе данной методологии лежит строгая последовательность вы-

полнения этапов жизненного цикла ИС, таких как: анализ, проектирование, реализация и тд. Начало каждого нового этапа происходит только в случае полного завершения предыдущего. Каскадный подход способствует быстрому и экономичному созданию системы. Однако, стоит учесть, что эффективно модель водопада может быть применена только в случае, когда все требования известны и утверждены, а также являются стабильными, то есть не будут подвержены изменениям в ходе проектирования.

V-Model – разновидность каскадной модели проектирования, суть которой заключается в последовательности выполнения этапов жизненного цикла и объединением фазы тестирования с каждым из них. Такая методология подходит для систем со сложными требованиями или разработки кардинально нового программного продукта. При этом все эти сложные требования должны быть определены заранее и не изменять в процессе всего жизненного цикла ИС для достижения максимальной эффективности данного подхода.

Спиральная модель (Spiral) – модель ИС, сочетающая в себе итерационный подход и поэтапность процессов. Данную модель эффективно используется для систем в которых присутствуют серьёзные риски. Процесс разработки происходит в виде спирали, где каждый виток спирали – это итерация. Каждая итерация состоит четырёх обязательных этапов: планирование, анализ рисков, разработка и тестирование, оценка созданного прототипа и планирование следующей итерации. По результатам каждого витка спирали создаётся фрагмент или версия ИС. По мере движения по спирали, происходит углубление и детализация разработки системы, а также выбирается один готовый, наиболее подходящий вариант системы. Спиральная модель характерна к применению для различных исследовательских проектов. Данный подход является очень дорогим в использовании и оправдан только для тех систем, у которых имеются серьёзные риски.

Итеративная модель – модель с разбиением проекта на этапы, каждый из которых характеризуется прохождением жизненного цикла ИС. Как правило один этап (итерация) имеет длительность не более 6 недель. Важной отличительной чертой итеративной модели, является низкая стоимость вносимых изменений и исправления ошибок. Однако, расходы на саму организацию

процессов довольно высоки, по сравнению с той же каскадной моделью. Итерационный подход используется во многих современных методологиях проектирования ИС, например, таких как: Scrum, XP и другие.

Методология Скрам (Scrum) – популярная и довольно лёгкая методология разработки ИС и ПО, основанная на итерационном подходе. Итерация именуется спринтом (Sprint). В течении спринта ведется работа над разработкой программ или систем, а по его окончании должна быть получена рабочая версия программной продукции. Спринт всегда имеет временные ограничения, которые составляют не более 4 недель и остаются равными на протяжении всего жизненного цикла продукции. До начала каждого спринта проводятся митинги планирования (Sprint Planning), на котором производится оценка текущей версии продукта, а также определяется бэклог спринта (Sprint Backlog), то есть содержание и распределение задач на предстоящий спринт. Заканчивается итерация формированием демонстрационной версии продукции, митингом ретроспективы спринта, на котором происходит оценка полученного продукта и предлагаются возможные направления его улучшения. Команда методологии скрам обязательно включает в себя 3 роли: заказчика или его представителя, скрам-мастера – управляющего проектом, а также остальных членов команды.

Методология экстремального программирования (XP) – состоит из 12 практик: парное программирование, разработка через тестирование, рефакторинг, простая архитектура, коллективное владение кодом, непрерывная интеграция, заказчик в команде, частые релизы, игра в планирование, 40-часовая рабочая неделя, стандарты кодирования, метафора системы. Обязательно использование всех 12 практик.

Методология экстремального программирования (XP) – представляет собой гибкую методологию, объединяющую в себе достоинства методов и традиционных практик разработки систем и ПО, выводя их на новый «экстремальный уровень». Методология XP состоит из 12 практик разработки программной продукции:

- парное программирование;

- применения тестирования, до написания кода/ (тест определяет код, а не наоборот);
- постоянная оптимизация кода/системы с целью его упрощения (рефакторинг);
- простая архитектура системы;
- коллективное владение кодом;
- непрерывная интеграция новых частей и версий программного продукта;
- присутствие заказчика в команде разработчика;
- регулярный выпуск новых версий;
- применение стандартов кодировки;
- обязательное командное планирование процессов;
- стандартная 40-часовая рабочая неделя;
- простота и понятность в наименованиях и аббревиатурах.

Методология Канбан (Kanban) – представляет собой конвейер по постановке и выполнению задач. Сущность методологии основывается на применении 3-х основных правил: процесс разработки системы визуализируется при помощи специальных канбан-досок, количество задач на каждом этапе жизненного цикла ИС ограничено, работа команды подвергается регулярной оценке и изменениям, направленным на повышение её эффективности. Применяется в основном для построения систем простой архитектуры и/или решения локальных задач разработки.

Методология RAD (Rapid Application Development) – методология, основанная на итерационном подходе, направленная на быструю разработку систем, не требующих сложных архитектурных решений. В своём арсенале RAD использует мощный инструментарий уже готовых решений и компонентов, что помогает максимально снизить издержки, связанные с процессом проектирования и разработки системы. В качестве недостатка, следует выделить, что методология RAD рассчитана на проекты только длиной в 60 – 90 дней.

Прочие фреймворки процессов – методологии, в основе которых лежит большое количество правил и описаний процессов жизненного цикла информационной системы. Придерживаться всех правил необязательно, достаточно выделить подходящие для своей системы и адаптировать их в процессе проектирования. RUP, EssUp и другие.

3. ПРАКТИЧЕСКАЯ ЧАСТЬ

Выбор наиболее выигрышной методологии проектирования ИС достаточно сложный процесс. Он требует детального анализа, а также должен учитывать множество факторов. Для упрощения и визуализации этого процесса можно представить его в виде блок-схемы (Рисунок 1)

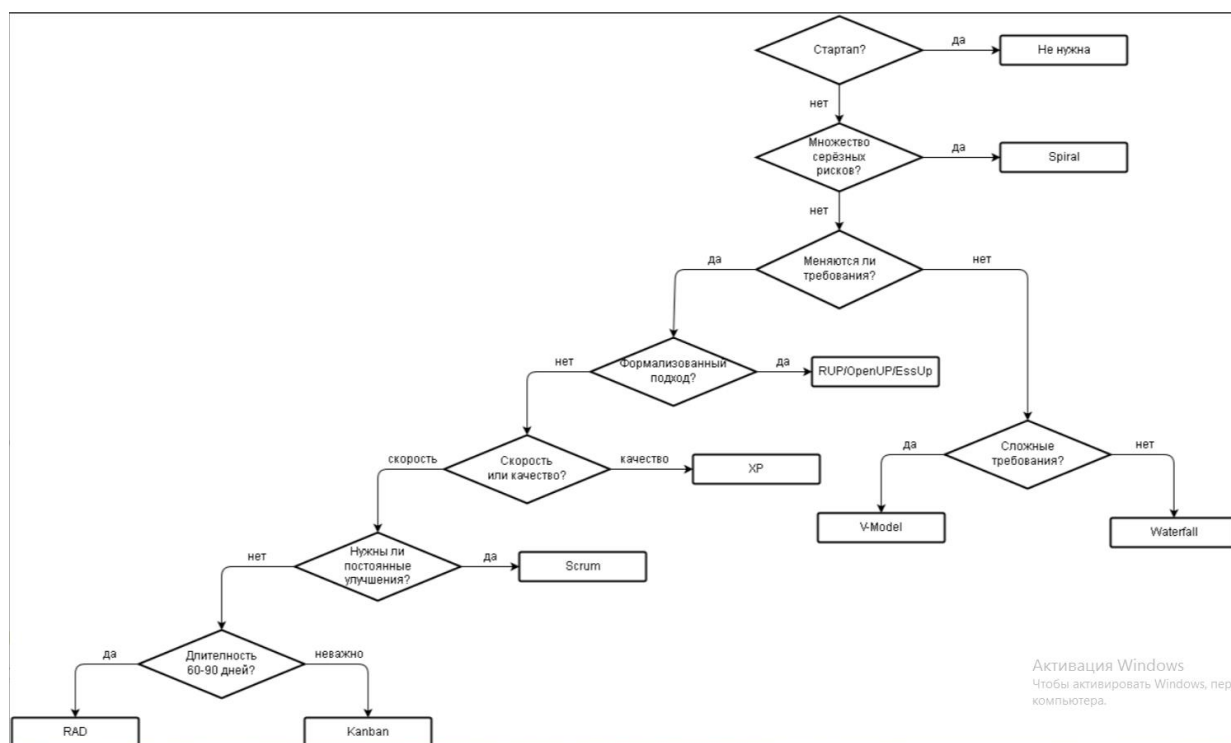


Рисунок 1 – Блок-схема выбора методологии проектирования ИС

Стоит отметить, что универсальных правил, которые подошли бы к каждой системе для выбора эффективной методологии проектирования ИС не существует. Каждый проект является по определению уникальным процессом, а соответственно и выбор методологии будет зависеть от конкретно взятой системы и имеющихся условий её проектирования и реализации. Однако, данная блок-схема позволит рассмотреть распространённые факторы выбора и методологии проектирования, наиболее подходящие в тех или иных случаях.

Составим пошаговое описание блок-схемы, в которой отразим все ключевые факторы, которые влияют на предпочтительность и эффективность применения методологии проектирования.

Шаг №1 (Старт проекта)

Не стоит торопиться с выбором методологии при стартапе. Нарботав какую-то базу, более чёткую направленность, собрав информацию и проанализировав её, будет намного проще подобрать наиболее выигрышную методологию.

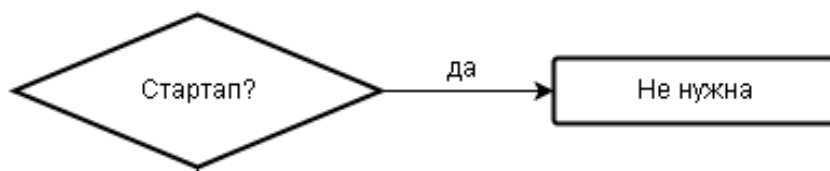


Рисунок 2 – Шаг №1 блок-схемы

Шаг №2 (Анализ рисков)

В каждом проекте предполагается наличие рисков. В некоторых случаях риски являются настолько серьёзными, что уже на ранних этапах проектирования возможность реализации задуманной системы ставится под вопросом. В таком случае следует обратиться к спиральной модели, в рамках которой проектирование начнётся с разработкой концептов, моделей и прототипов системы. Такой подход поможет снизить потенциальные риски, а также установить факт возможности реализации системы. Как правило, применение спиральной модели характерно для всех исследовательских моделей.

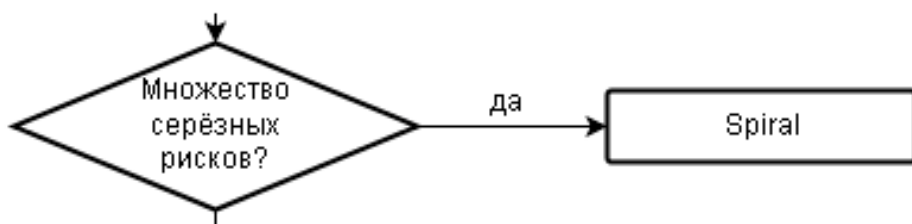


Рисунок 3 – Шаг №2 блок-схемы

Шаг №3 (Разработка и анализ изменчивости требований)

В случае отсутствия серьёзных рисков, следующим пунктом станет анализ изменчивости требований. Если все требования определены и чётко прописаны заранее, их существенное изменение не предполагается в рамках разработки проекта, то следует обратить внимание на каскадную модель. Каскадная модель хо-

рошо подходит если большая часть деталей проекта заранее определены и согласованы, а также реализации проекта не должна растянуться на длительное время. В таком случае каскадная модель позволит сократить расходы на реализацию проекта, а также позволит сосредоточиться на непосредственном проектировании ПО.

Если же проект займёт длительное время и предполагает внесение частых изменений, стоит обратить внимание на другие подходы.



Рисунок 4 – Шаг №3 блок-схемы

Шаг №4 (Анализ сложности требований)

Сложность требований ещё один серьёзный фактор в вопросе выбора методологии проектирования. Серьёзные и грандиозные задумки и, соответственно им, сложные требования вызывают потребность в более детальном анализе особенностей будущего тестирования системы ещё на ранних стадиях её проектирования. Поэтому стоит заранее разработать стратегии тестирования, подготовить тест-планы и тест-кейсы. Как раз при таких условиях наиболее выгодной к использованию методологией станет V-Model, являющаяся разновидностью каскадной модели. Для более простых требований она является не целесообразной, поскольку будет приумножать дополнительную документацию, необходимость детального анализа и лишние этапы тестирования, что приведет к увеличению проектировочных расходов.



Рисунок 5 – Шаг №4 блок-схемы

Шаг №5 (Анализ соответствия проекта нормативам и стандартам)

Когда система разрабатывается для общественно важных сфер, таких как: (медицинская, энергетическая, военная, транспортная и др.), она должна быть выполнена строго в соответствии с отраслевыми стандартами. Такие системы подлежат тщательному тестированию и лицензированию. Все стадии проектирования, промежуточные и итоговые результаты, решения, принятые при их достижении должны быть задокументированы. Такая детальная регламентация соответствует применению формализованного подхода, который характерен для методологий (фреймворк процессов): RUP, OpenUp, EssUp.



Рисунок 6 – Шаг №5 блок-схемы

Шаг №6 (Определение предпочтительности скорости выполнения проекта и глубины, уникальности его проработки)

При отсутствии необходимости формализации рассматривают гибкие методологии. Главные факторы выбора здесь это скорость создания системы и её детализация, уникальность решений. Для создания кардинально новой продукции, требующей сложные приёмы программирования, высокого уровня организа-

ции процесса проектирования и профессиональной команды применяют метод экстремального программирования (XP). Кроме того, практики методологии XP часто применяются и в других методологиях, где для разработки системы требуется более высокий уровень инженерии. Также довольно часто методология XP применяется в паре с другими методологиями.

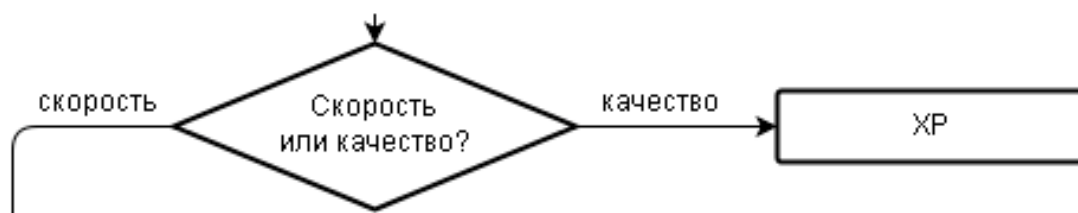


Рисунок 7 – Шаг №6 блок-схемы

Шаг №7 (Определение необходимости постоянных улучшений)

Несмотря на приоритет быстроте проектированию системы, есть случаи, когда она имеет потребность в регулярных усовершенствованиях процесса. Для этого есть Скрам методологии, где применяется митинг ретроспективы, которые проводятся в конце спринта. В рамках анализа спринта определяется успешность и не успешность тех или иных действий, определяются направления, которые нужно будет улучшить. Обычно Скрам применяется в тех случаях, когда используются незнакомые ранее технологии или над проектом работает неопытная и/или неслаженная команда.



Рисунок 8 – Шаг №7 блок-схемы

Также, часто методология Скрам применяется при запуске нового проекта, а в процессе его развития, отладки и стабилизации происходит плавный переход к другим методологиям.

Шаг №8 (Определение длительности проекта)

При отсутствии необходимости в постоянных улучшениях, стоит обратиться к методологиям, которые позволят сконцентрироваться на выполнении поставленных задач в рамках проектирования ПО. Среди таких методологий можно выделить RAD и Kanban. RAD подойдёт в случаях когда длительность проекта не составит более 60-90 дней. Для длительных проектов можно воспользоваться методологией Kanban. Данная методологию можно сравнить с непрерывным конвейером, который при необходимости может работать бесконечно. Ориентирован в основном на проекты поддержки и не очень подходит для систем сложной архитектуры, поскольку направлен на решение конкретных локальных задач.



Рисунок 9 – Шаг №7 блок-схемы

4. ПРАКТИЧЕСКОЕ ЗАДАНИЕ

1. Выбрать ИС для проектирования и согласовать с преподавателем. Кратко описать её назначение, перечислить требования к системе.

2. На основе изученного материала в данном методическом указании составить блок-схему выбора методологии проектирования ИС с пошаговым пояснением.

3. Определить наиболее выигрышную методологию проектирования для своей системы. Сделать вывод о выбранной методологии и привести аргументы своего выбора.

4. На основе блок-схемы определить альтернативную методологию проектирования ИС. Аналогично п.№3 выполнить задания для вновь выбранной методологии.

5. Сделать отчёт о проделанной работе.

5. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Что такое проект? Как определяется начало и завершение проекта?
2. Какие классификационные признаки проектов вы знаете?
3. Какие существуют типы проекта по уровню?
4. Чем характеризуется инновационный проект?
5. Что такое методология проектирования ИС?
6. Перечислите известные вам методологии проектирования ИС?
7. Что такое модели ИС? Какие модели ИС вам известны?
8. В чём заключается основной смысл ХР методологии?
9. Какую лучше выбрать методологию если на ранних стадиях проектирования ИС известно о серьёзных рисках?
10. В каких случаях целесообразно применять методологию V-модели?

СПИСОК РЕКОМЕНДУЕМОЙ ЛИТЕРАТУРЫ

1. Аньшин, В. Управление проектами. Фундаментальный курс / В. Аньшин, О. Ильина. – Москва: Высшая школа экономики, 2013. – 624 с.
2. Бусеро А. Руководитель проектов. Все навыки, необходимые для работы / А. Бусеро, Р. Инглунд. – Москва: Манн, Иванов и Фербер, 2018. – 384 с.
3. Круглов, М. Г. Инновационный проект. Управление качеством и эффективностью / М. Г. Круглов. – Москва: Дело НХ, 2014. – 336 с.
4. Мазур И. И., Шапиро В. Д. и др. Управление проектами / Справочное руководство. Москва.: Высшая школа, 2001 875 с.: ил. Меридиан. – 2020. – № 7(41). – С. 60-62.
5. Емельянова, Н. З. Проектирование информационных систем: Учебное пособие / Н. З. Емельянова, Т. Л. Партыка, И. И. Попов. – Москва.: Форум, 2013. – 432 с.
6. Заботина, Н. Н. Проектирование информационных систем: Учебное пособие / Н. Н. Заботина. – Москва.: НИЦ ИНФРА-М, 2013. – 331 с.

ЛАБОРАТОРНАЯ РАБОТА №4. CASE-ТЕХНОЛОГИИ РАЗРАБОТКИ АВТОМАТИЗИРОВАННОЙ СИСТЕМЫ

1. ЦЕЛЬ РАБОТЫ

Целью данной работы является получение теоретических знаний и практических навыков проектирования информационных систем при помощи CASE-средств.

2. ТЕОРЕТИЧЕСКОЕ ПОЛОЖЕНИЕ

2.1 Проектирование ИС

Современные информационные системы (ИС) являются сложным продуктом совокупной деятельности специалистов различных отраслей и уровней. Создание ИС представляет собой ряд задач, для решения которых необходимо применение специальных методик, средств и инструментов.

Под технологией проектирования (создания) ИС обычно принято понимать набор методов и методик, технических средств, которые упорядочены в логической последовательности и нацелены на поиск путей реализации общей концепции системы, её создания или изменения.

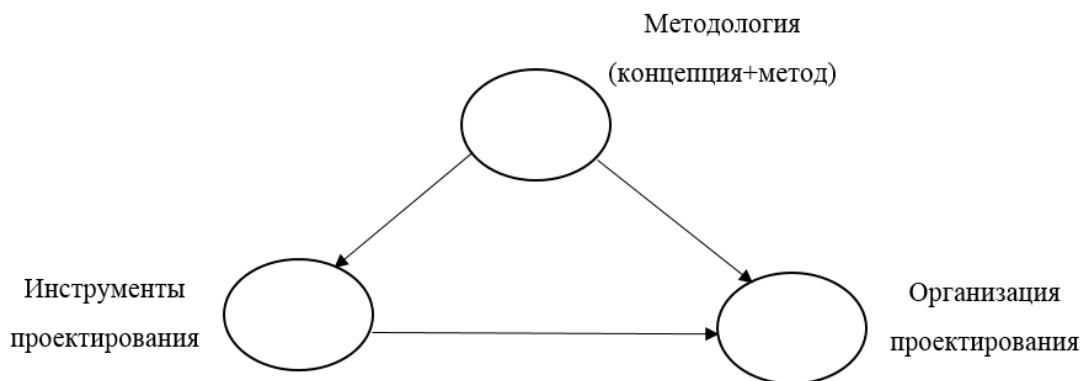


Рисунок 1 – Компоненты проектирования ИС

Выбор технологии проектирования ИС производится на основе следующих требований, которые она должна удовлетворять:

- Конечный проект должен соответствовать всем требованиям заказчика;

- Технология должна охватывать максимум этапов жизненного цикла проекта;
- Должна обеспечиваться минимизация ресурсных затрат в процессе проектирования;
- Технология должна обеспечивать надежность и безопасность процесса проектирования и последующего использования проекта;
- Должна обеспечиваться простота и доступность на всех этапах проектирования, в том числе и для готового продукта (проекта).

В настоящее время принято выделять две основных технологии проектирования ИС: каноническую и индустриальную.

Индустриальная технология в свою очередь подразделяется на два класса: автоматизированное – применение CASE-технологий, типовое – проектирования на основе параметрического и модельного ориентирования. При этом не исключается совместное использование данных технологий.

2.2 Понятие CASE-технологий

CASE-технологии (Computer-Aided Software Engineering) – методология системной инженерии, а также совокупность средств и инструментов, позволяющая производить моделирование предметной области в наглядной, доступной форме, анализировать полученную модель на всех этапах разработки и функционирования ИС.

Современные CASE-средства способны удовлетворять множество потребностей, возникших при проектировании ИС: от элементарных средств анализа и документирования до крупномасштабных средств по автоматизации, способных затрагивать абсолютно весь жизненный цикл информационной системы.

Основой большинства современных CASE-средств являются методологии структурного или объектно-ориентируемого анализа, при использовании спецификации в виде различных методов визуального представления информации, таких как диаграммы или текстовые вставки. Такие графические средства позволяют в наглядном виде смоделировать и изучить проектируемую ИС,

описать её характер поведения и архитектуру, а также изменять её в соответствии с требованиями и целями данной системы.

С помощью CASE-средств можно создать практически полностью готовый к применению продукт, при этом соблюдая все основные парадигмы процесса проектирования системы. Основной целью данной технологии является: облегчение работы проектировщика, повышение производительности его труда, а также выделение основополагающей информации о проектируемой системе в простом и доступном виде.

В основном, к CASE-средствам принято относить любую программную продукцию, способную автоматизировать тот или иной этап, или совокупность этапов жизненного цикла ИС. Классификация CASE-средств основывается на их функциональной направленности на определённые стадии жизненного цикла информационных систем. В настоящее время наиболее популярными программными продуктами, представляющими CASE-средства, и направленными на анализ и проектирование ИС, являются:

- Silverrun;
- CASE.Аналитик;
- Vantage Team Builder;
- BPwin;
- DESIGNER/2000;
- Ramus Educational.

2.3. Принцип работы CASE-средств на примере программы Ramus Educational

В подавляющем большинстве основной функционал различных программных CASE-средств не имеет кардинальных отличий. Различия заключаются в дизайнерских решениях интерфейсного модуля, а также в дополнительном функционале программы. Выбор проектировщика основывается на его дизайнерских и архитектурных предпочтениях, а также на наличии необходимых ему дополнительных возможностей программы.

Поэтому типовое устройство и процесс проектирования можно разобрать в любом из перечисленных в предыдущем разделе программных комплексов.

Ramus Educational – программное обеспечение, которое относится к CASE-средствам и является инструментом для анализа, моделирования и проектирования системы.

Ramus обладает всем необходимым функционалом для проектирования ИС, а также является доступным и простым в использовании. Кроме того, в Ramus Educational есть возможность переключения русского языка, что в совокупности с вышеперечисленными достоинствами, делает его прекрасно подходящим для ознакомления и обучения работой с CASE-средствами при проектировании ИС.

Главным предназначением Ramus является описание бизнес-процессов предприятий и создание различных информационных систем.

К основным функциональным возможностям Ramus Educational относятся:

- Конструирование диаграмм стандарта IDEF0;
- Конструирование DFD диаграмм;
- Формирование классификаторов;
- Проведение импорта и экспорта в файлы IDL формата (Файлы разработчика).

Применение методологии IDEF0 подразумевает создание иерархической системы диаграмм. С помощью контекстной диаграммы описывается система в целом, её взаимодействия с внешним миром. Далее происходит разбиение системы на подсистемы, с помощью которых описывается функциональная часть ИС (диаграммы декомпозиции). Каждая вновь спроектированная подсистема может разделяться на ещё более мелкие подсистемы, и так, пока не будет достигнута нужная проектировщику подробность описания системы.

Рабочее пространство Ramus выглядит следующим образом (Рисунок 2)

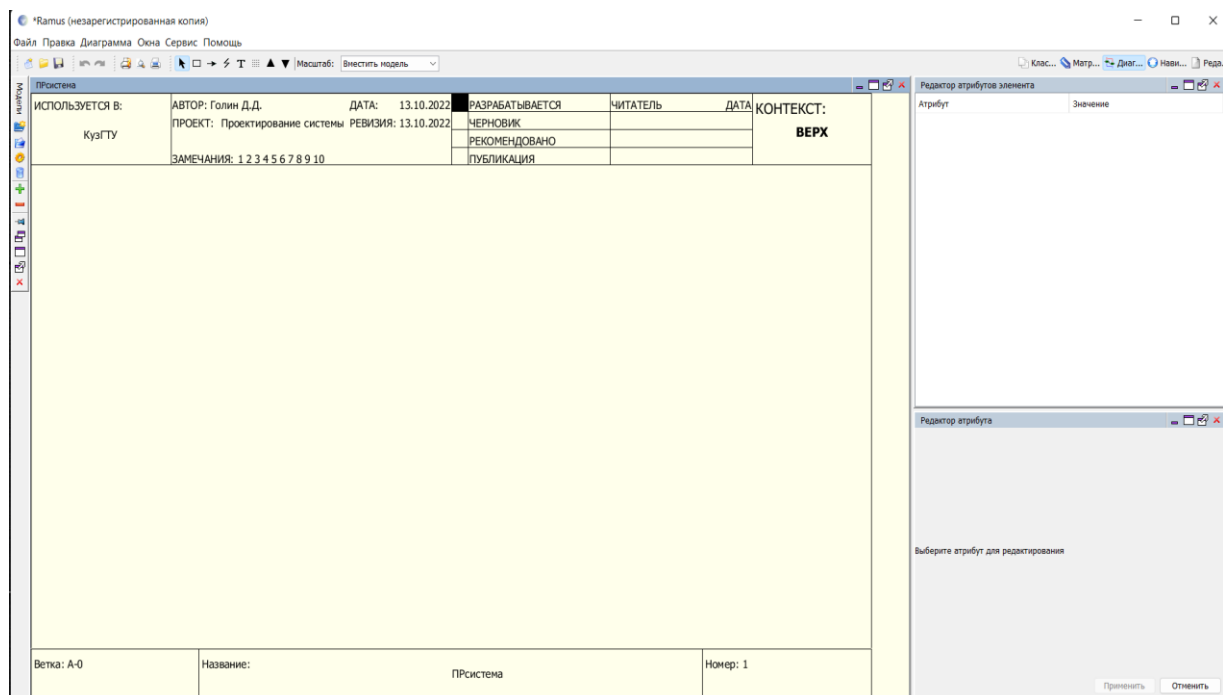


Рисунок 2 – Рабочее пространство Ramus

Помимо классического меню, позволяющего производить манипуляции с файлами, вносить изменения, взаимодействовать с окнами и получать подсказки по работе с программой, рабочее пространство Ramus может также похвастаться дополнительной вкладкой, отведённой созданным моделям, а также панелью инструментов и областью работы с диаграммами.

Работа с моделью осуществляется при помощи инструментов, указанных на Рисунке 3.



Рисунок 3 – Панель инструментов

Так как диаграмма IDEF0 должна содержать блоки, которые являются функциями моделируемой системы, и дуги, которые

отвечают за связи и взаимодействие блоков, эти элементы, соответственно, присутствуют и в Ramus.

Блоки изображаются с помощью прямоугольников, которые обозначают одноимённые процессы (работы), функции или задачи, которые должны протекать в рамках исследуемой системы. Добавить блок на диаграмму можно с помощью кнопки «Режим добавления функциональных блоков».

В качестве дуг на диаграммах выступают стрелочки. В методологии IDEF0 существуют стрелочки следующих видов:

Вход – определяет объект, который используется процессом для получения конечного результата (**выхода**). Стрелки входа располагаются по направлению к левой грани процесса.

Управление – представляет собой информацию о правилах и ограничениях на основании которых выполняется функция. Данный вид стрелок располагается над функциональным блоком и соответственно входят в его верхнюю грань. В отличие от стрелок типа вход, управляющие дуги являются обязательными на диаграмме.

Выход – объекты входа, которые преобразовались в результате выполнения процесса. Стрелка выхода является обязательной, поскольку для того, чтобы функция обладала смыслом, она должна иметь какой-либо результат. Изображается как исходящая от правой грани операции.

Механизм – это ресурсы с помощью которых выполняется функция. В качестве ресурсом могут выступать, оборудование, ПО, специалисты и др. Стрелка механизма не носит обязательного характера для диаграммы и входит в нижнюю грань блока.

Вызов – стрелка при помощи которой указывается ссылка на другую модель. Она также не является обязательной для данного типа диаграммы и исходит вниз из нижней грани функции.

При помощи перечисленных выше элементов проектируется контекстная диаграмма, которая в общем случае, должна обладать видом, как показано на Рисунке 4.

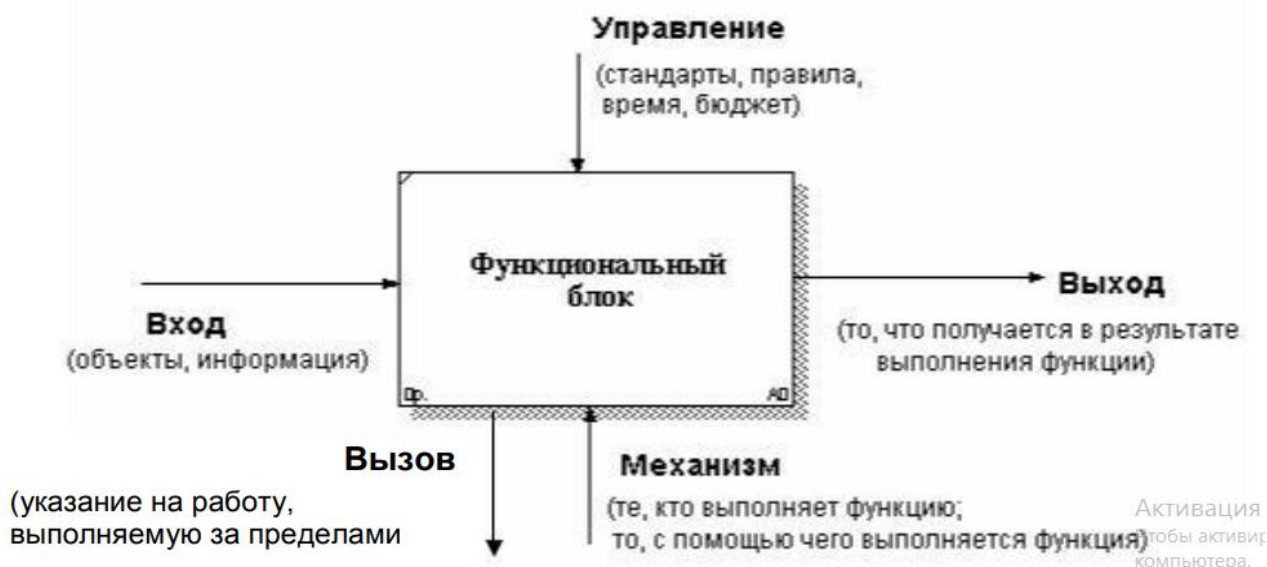


Рисунок 4 – Общий вид контекстной диаграммы

Диаграммы декомпозиции создаются аналогичным образом, что и контекстные диаграммы. Однако, характеризуются большим количеством процессов и взаимосвязей между ними.

3. ПРАКТИЧЕСКАЯ ЧАСТЬ

Процесс проектирования модели начинается с изучения системы, сопроводительной документации, целей её создания, её функциональных возможностей, а также пожеланий заказчиков.

Изучив исходные материалы, необходимо определить и сформировать цель моделирования, а также точки зрения относительно которых и будет рассматриваться будущая модель. Теперь можно приступать к созданию модели.

В качестве примера рассмотрим «классическую» пользовательскую систему. Суть данной системы заключается в наличии и обслуживании пользователей разного уровня (допуска), которые могут обращаться в систему, делать некоторые запросы, изменять данные и получать отчёты в качестве выходных продуктов.

Модель для данной системы спроектируем при помощи программного средства Ramus Educational.

*Создание проекта в **Ramus***

1. Необходимо произвести процедуру запуска приложения Ramus Educational, в результате чего откроется диалоговое окно, как показано на Рисунке 5.

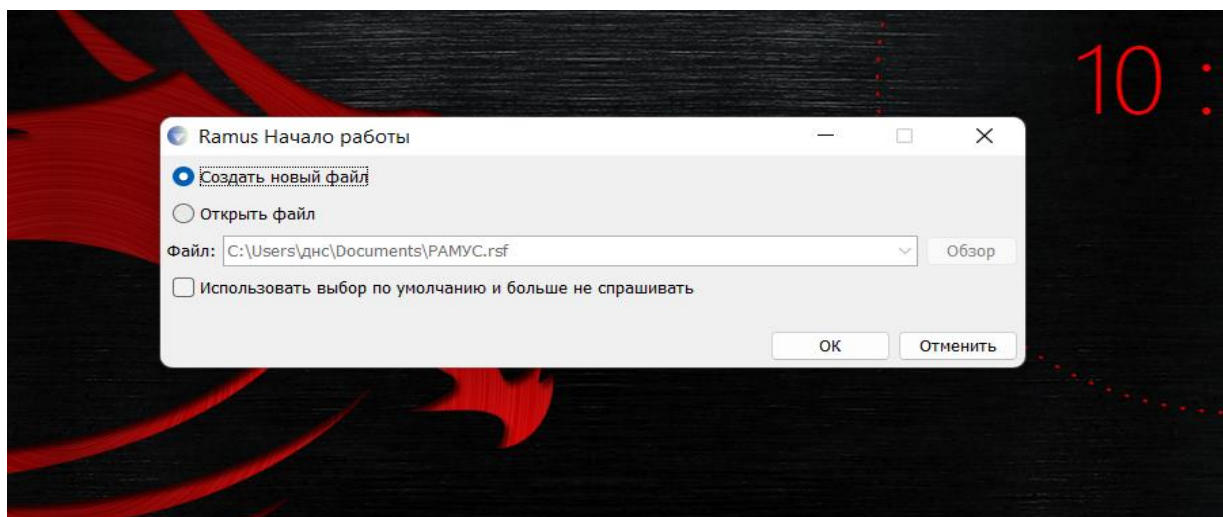


Рисунок 5 – Окно Начало работы

Выберем **Создать новый файл** и нажмем кнопку **ОК**.

2. Далее откроется мастер **Свойства проекта** (Рисунок 6). В данном окне необходимо указать автора, название проекта, название модели. Также необходимо выбрать тип диаграммы, в нашем случае это **IDEF0**. Нажимаем кнопку **Дальше**.

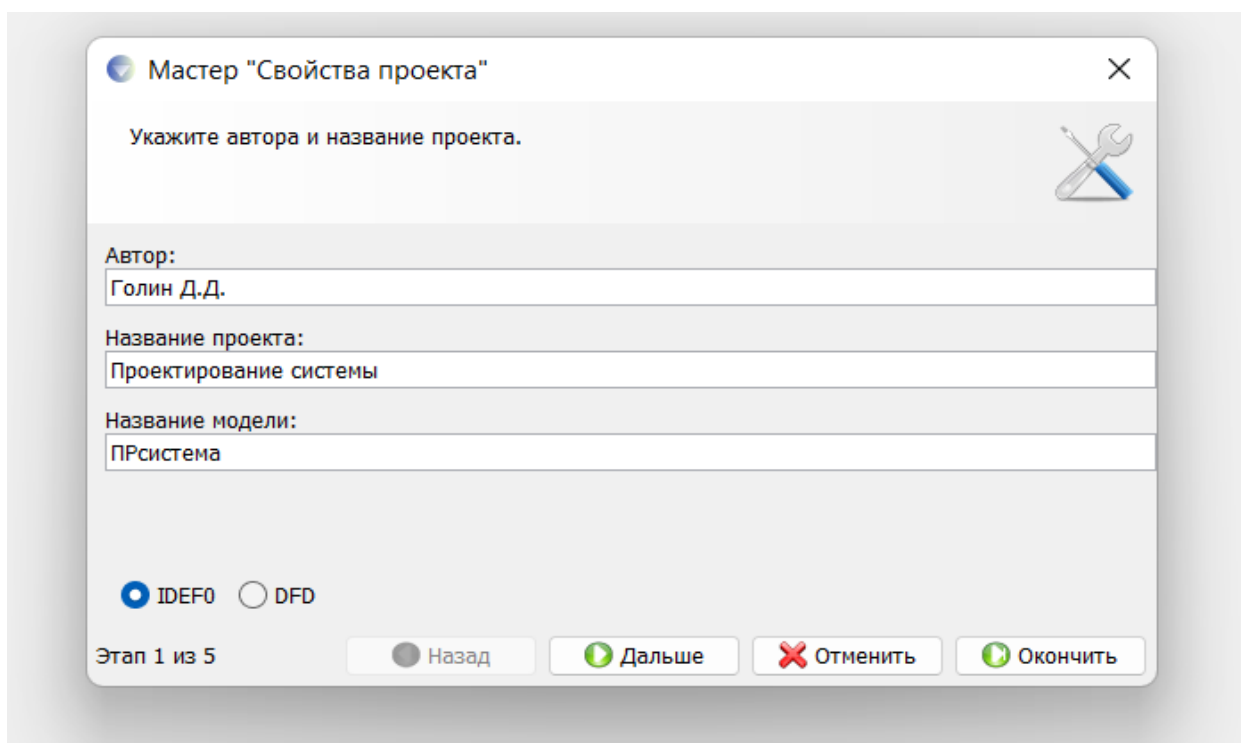


Рисунок 6 – Окно определения автора и названия проекта

3. Следующим этапом станет определение места, где используется проект (Рисунок 7).

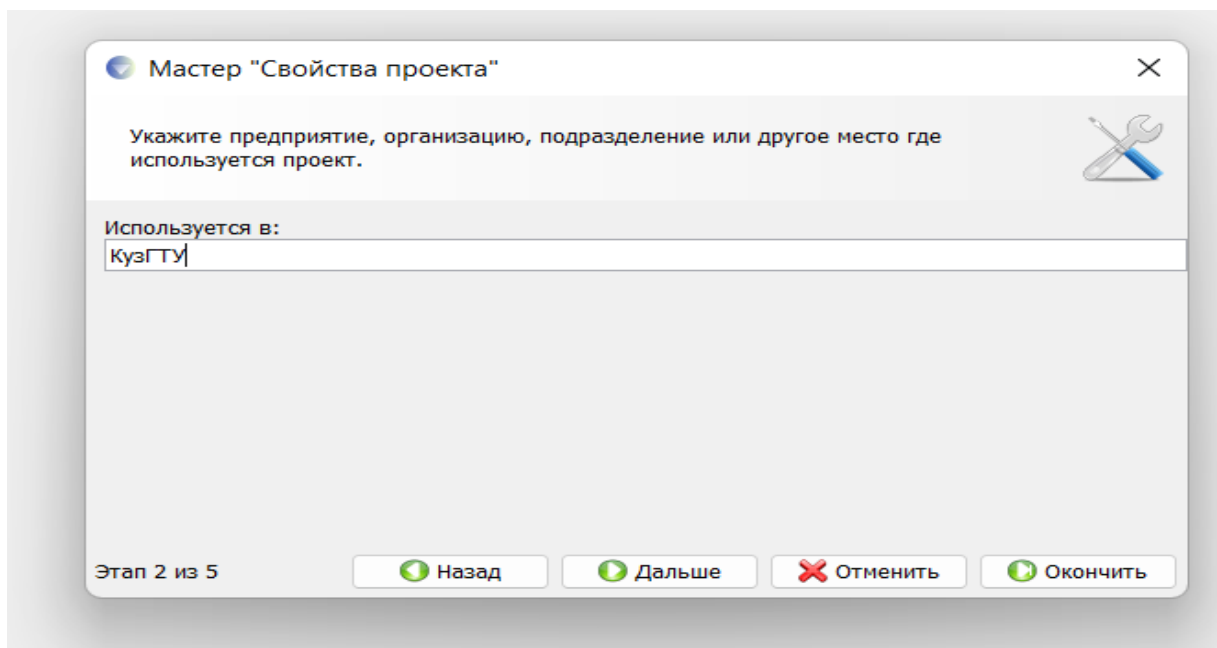


Рисунок 7 – Окно определения места использования проекта

4. Теперь мы можем дать краткое описание нашего проекта (Рисунок 8)

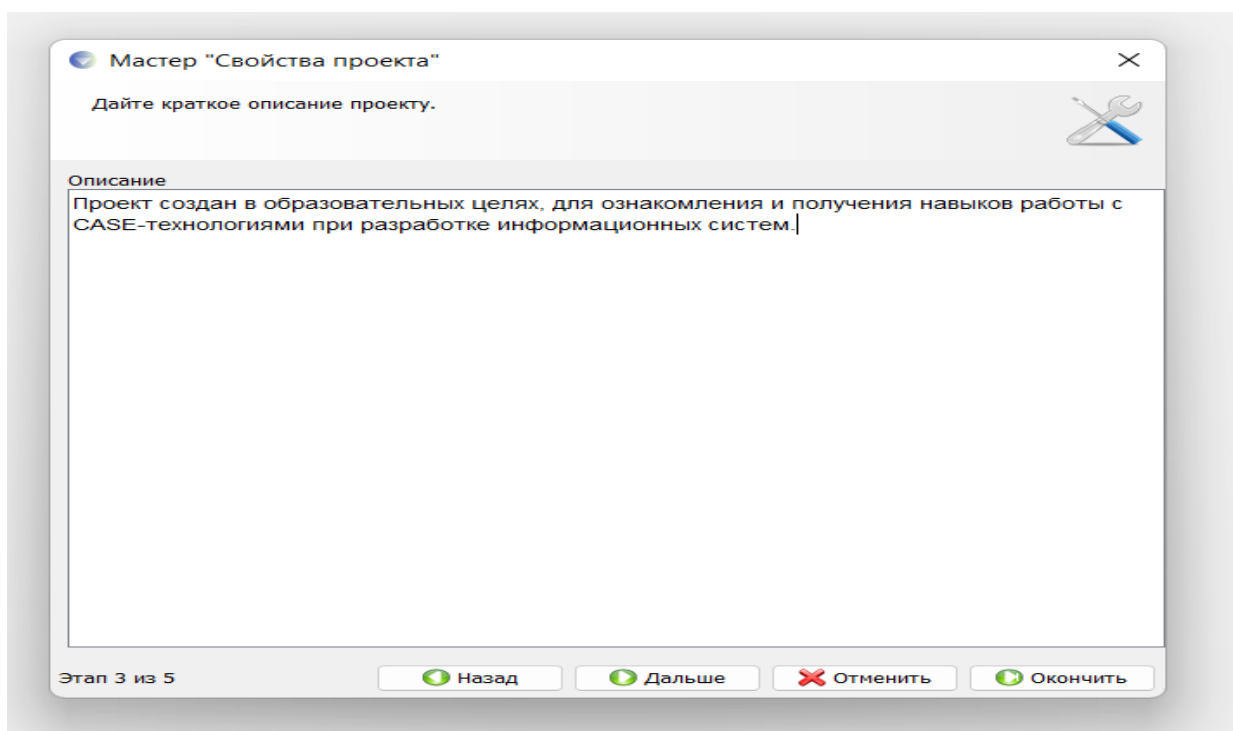



Рисунок 8 – Окно описания проекта

5. Следующие два этапа Свойств проекта предполагают классификаторов и их собственников. Поскольку в нашем проек-

те эта информации не нужна, данные этапы можно пропустить и нажать кнопку **Окончить**.

В результате создания нового файла откроется рабочее пространство Ramus.

Построение контекстной диаграммы IDEF0

1. Построение диаграммы начинается с добавления **функционального блока** с помощью соответствующей кнопки на панели инструментов .

В открывшемся окне **Свойства функционального блока** необходимо ввести наименование нашей функции. Помимо этого, в данном окне также можно изменить шрифт текста, цвет текста и фона, выбрать собственника или изменить тип функционального блока (Рисунок 9).

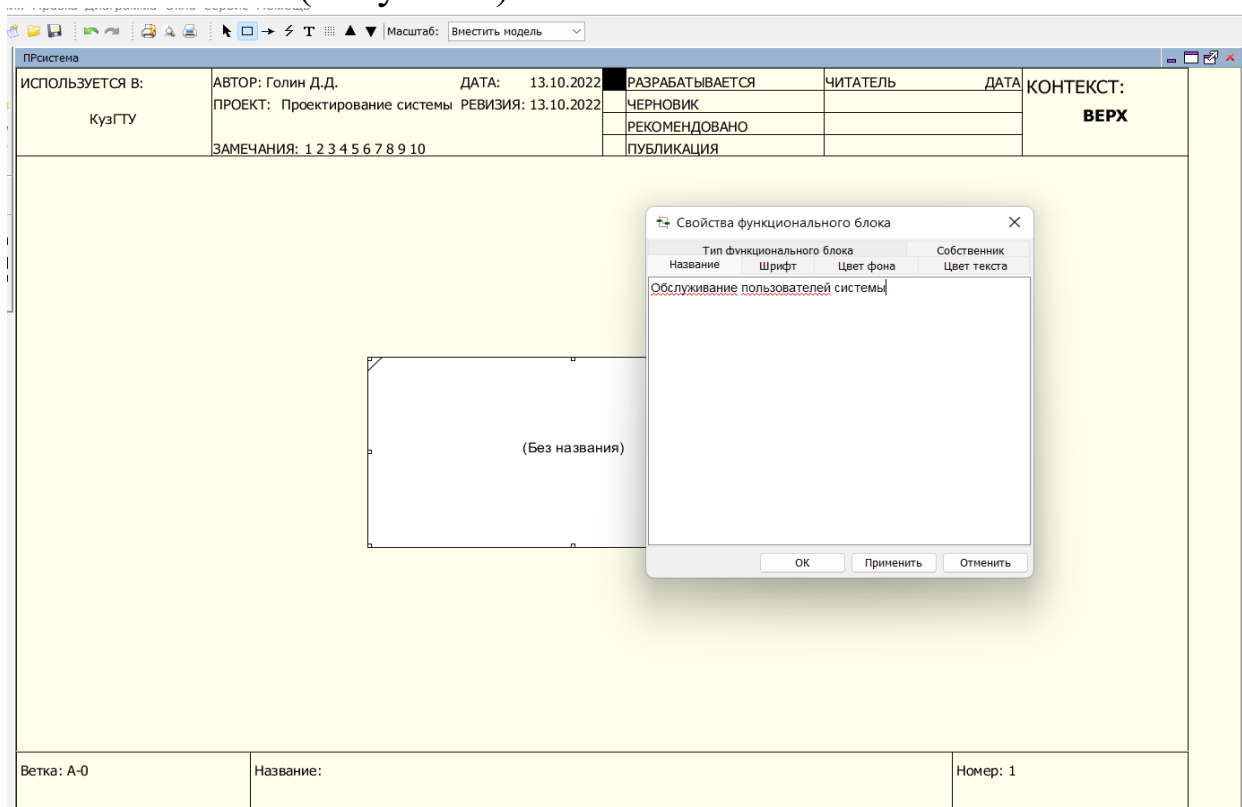


Рисунок 9 – Окно «Свойства функционального блока»

Так как ранее уже было определено, что основной функцией нашей системы является **обслуживание пользователей системы**, то именно так и определим единственный процесс нашей диаграммы.


2. Теперь необходимо добавить остальные элементы диаграммы.

В качестве входных данных нашей системы будут задействованы: *имя пользователя, данные авторизации пользователя, исходная база данных, запрос пользователя.*

В качестве управляющей стрелки будут *параметры доступа.*

В виде конечных результатов мы хотим получить *отчётную документацию и изменение базы данных.*

Механизмами нашей модели выступят *системный администратор* и *мониторинг системы.*

Для того, чтобы добавить все вышеперечисленные стрелки необходимо нажать кнопку  **Режим работы со стрелками.** Затем необходимо навести курсор на точку начала стрелки (которая должна быть расположена в левой, правой, верхней или нижней грани области построения диаграммы) и направить её к грани нашего функционального блока.

Задать имя стрелки можно с помощью двойного клика по ней или нажав правой кнопки мыши и найти нужную строчку в открывшемся контекстном меню.

В результате контекстная диаграмма нашей системы будет иметь следующий вид:

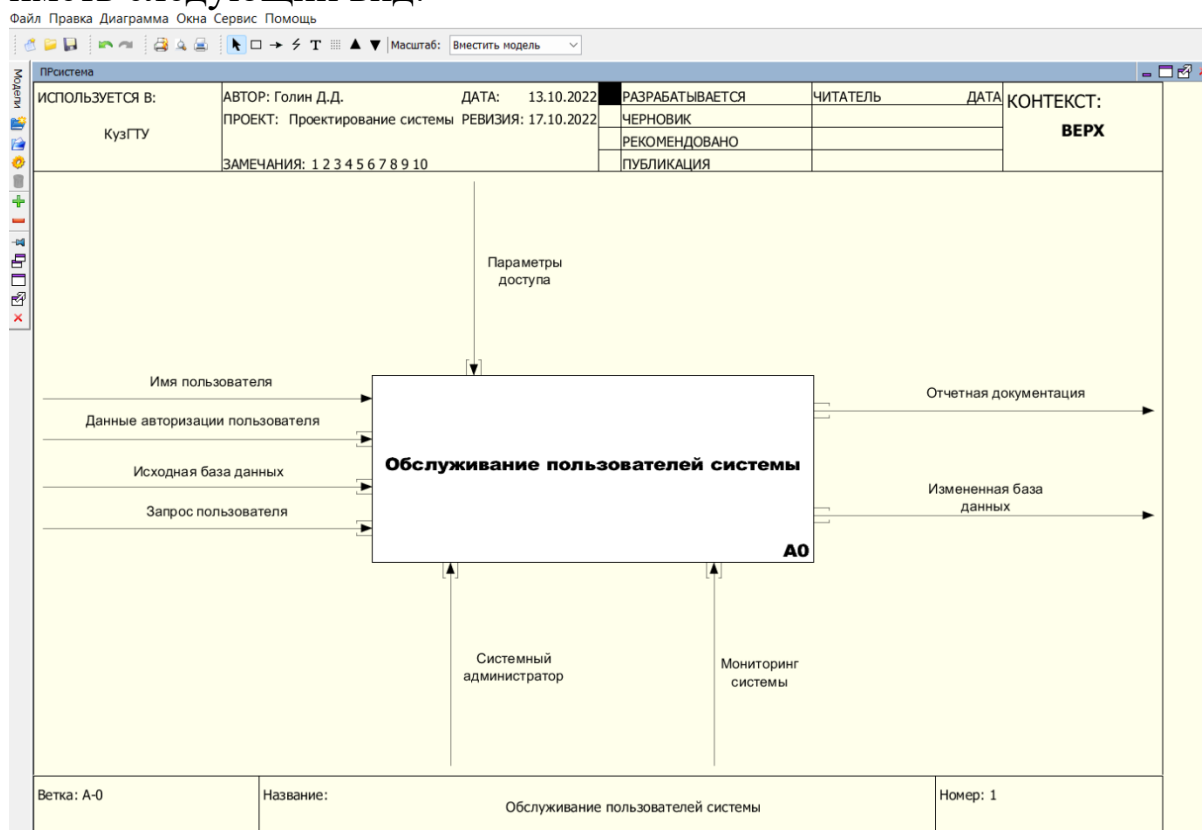



Рисунок 10 – Контекстная диаграмма пользовательской системы

Создание *декомпозиции* контекстной диаграммы

1. Декомпозиция контекстной диаграммы заключается в описании последовательности обслуживания пользователей. Для того, чтобы начать декомпозицию необходимо воспользоваться кнопкой  **Переход к дочерним диаграммам**. В появившемся окне укажем число операций, в нашем случае их будет **4**, а также определим *классическую* структуру диаграмма и выберем тип *IDEF0* (Рисунок 11).

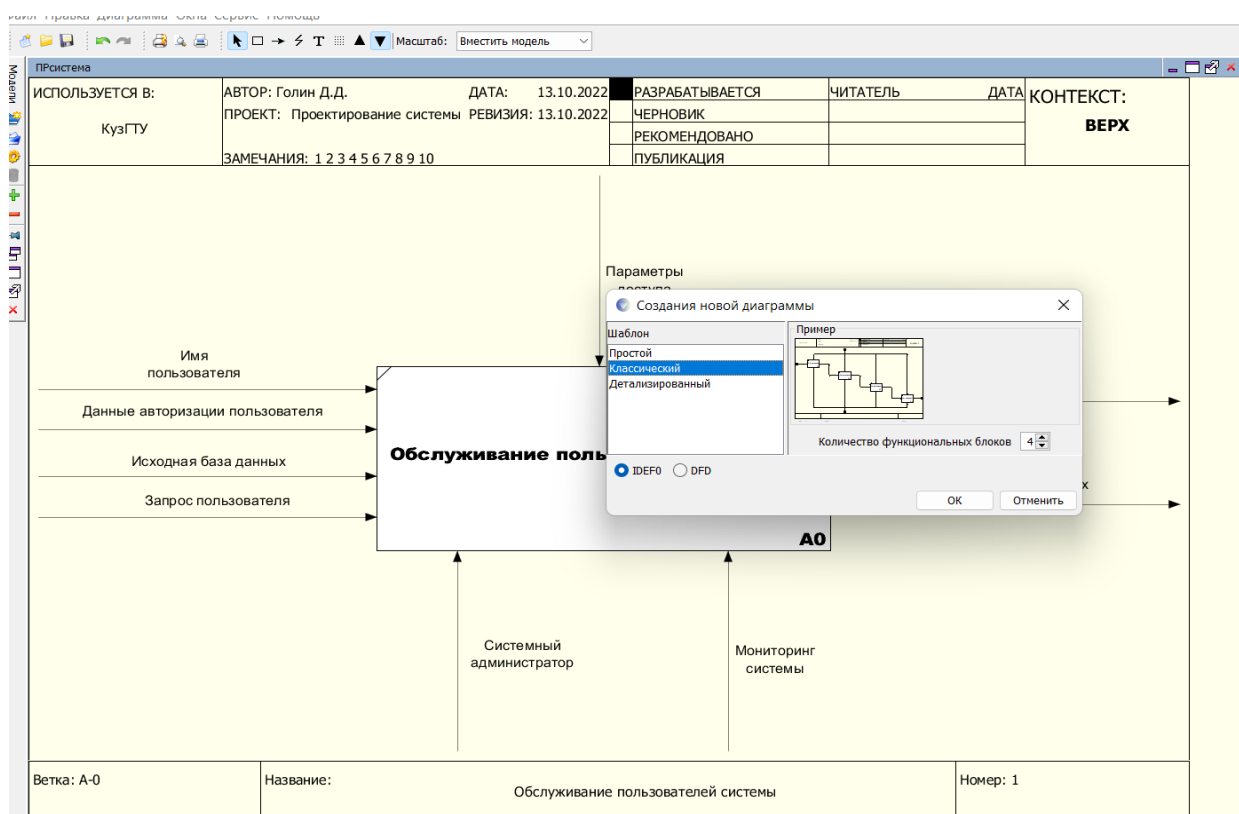



Рисунок 11 – Окно создания дочерней диаграммы

2. Теперь необходимо выбрать функциональные блоки и добавить их наименование, аналогично предыдущему разделу. Операции будут описывать процесс обслуживания пользователя системой и должны располагаться в следующем порядке:

- Определение параметров доступа в систему;
- Обращение к системе;
- Изменение базы данных (если это будет необходимо);
- Обработка запроса пользователя.

Для удобства восприятия блоки можно самостоятельно перемещать по полю диаграммы, а также изменять размеры.

3. Из родительской (структурной) диаграммы автоматически добавятся стрелочки. Эти стрелочки (граничные) нужно связать с функциональными блоками диаграммы при помощи  **Режим работы со стрелками**. С помощью той же кнопки необходимо будет добавить и внутренние стрелки, которые будут связывать и описывать взаимодействия операций между собой.

В результате будет построена следующая диаграмма декомпозиции:

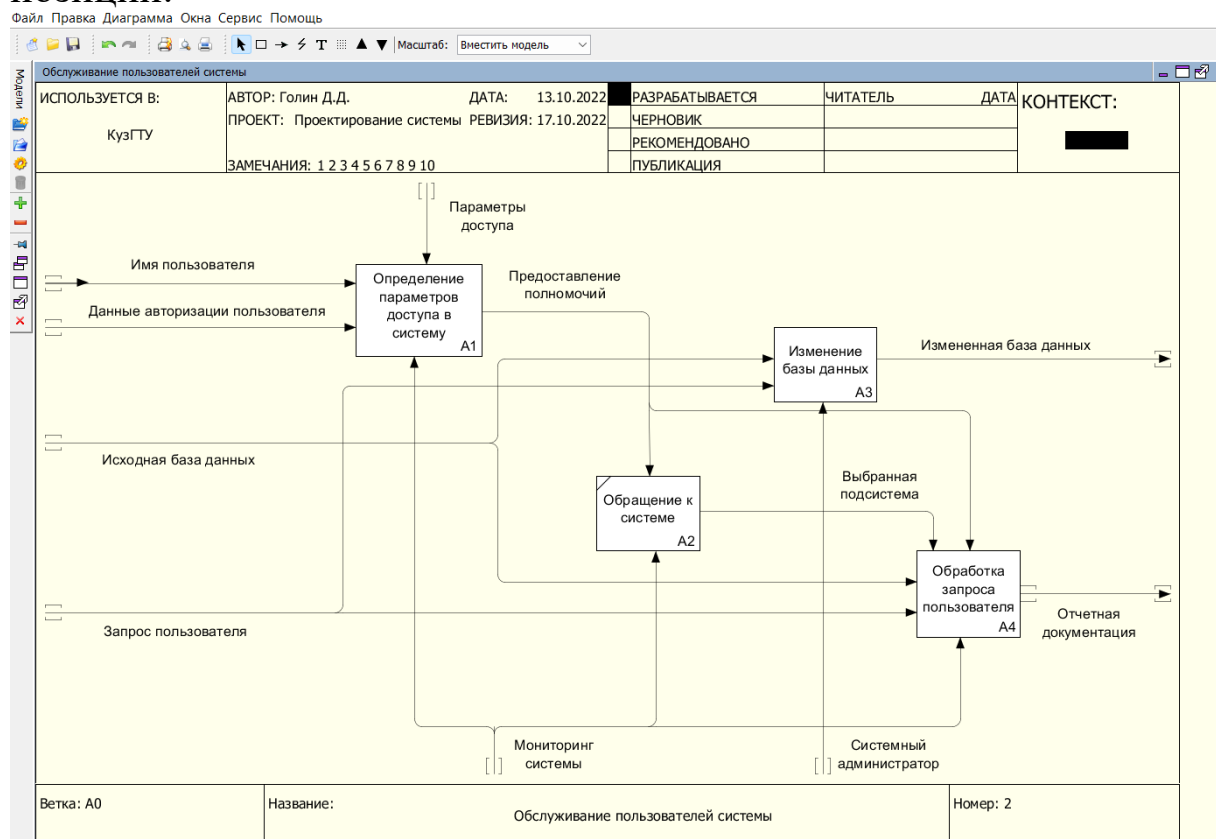


Рисунок 12 – Диаграмма декомпозиции контекстной диаграммы

Создание диаграмм декомпозиции следующих уровней

Последовательно произведём декомпозицию каждой операции, аналогично предыдущему разделу

1. Для этого выберем необходимый процесс и спроектируем для него диаграмму. Для того, чтобы описать **Определение параметров доступа в систему** нужно будет добавить следующие функциональные операции:

- Определение категории пользователей;

- Определение полномочий;
- Открытие доступа в базу данных системы.

Созданные операции свяжем с помощью внутренних стрелок и получим следующую диаграмму декомпозиции:

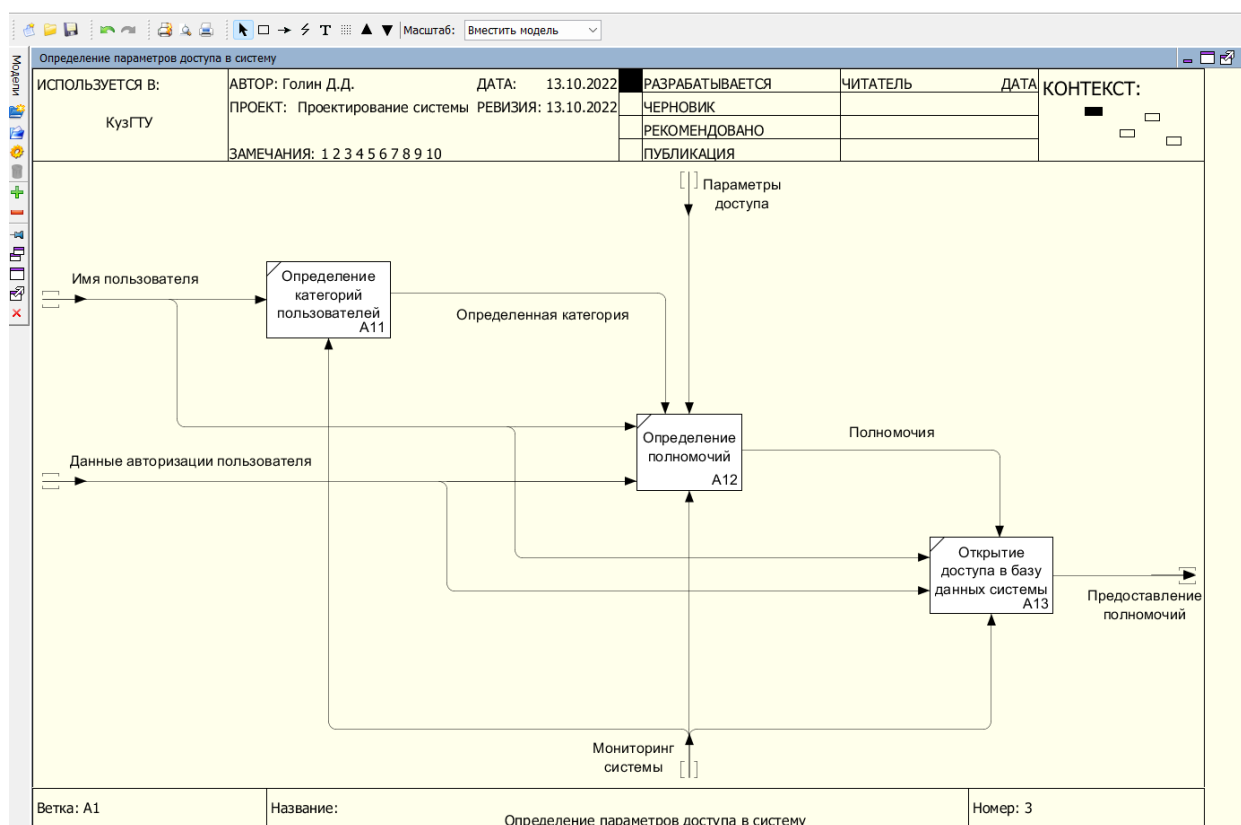


Рисунок 13 – Диаграмма декомпозиции
«Определение параметров доступа в систему»

2. В рамках установленной цели нашей модели системы необходимо также произвести декомпозицию **Обработка запроса пользователя. Изменение базы данных и обращение к подсистеме** не нуждаются в декомпозиции, поскольку нас не интересуют детали алгоритмов и протекающих процессов внутри самой системы.

При декомпозиции данного процесса нужно учитывать, что перед тем, как начнёт поиск ответа на запрос пользователя, необходимо установить соединение с базой данных и сообщить об этом системе. После этих процедур система сможет осуществить запрос пользователя и сгенерировать отчётную документацию.

Ещё одним важным моментом для данной диаграммы является то, что **Выполнение запроса**, включает в себя работу задей-

ствовавших подсистем. На основании этого необходимо определить, как именно будет производиться генерация отчётной документации, добавить правила, то есть создать шаблон. Поэтому добавим шаблоны отчётной документации, которые будут соответствовать запросам пользователей. Пользователю также даётся возможность выбора формы отчётной документации.

В результате получим следующую диаграмму:

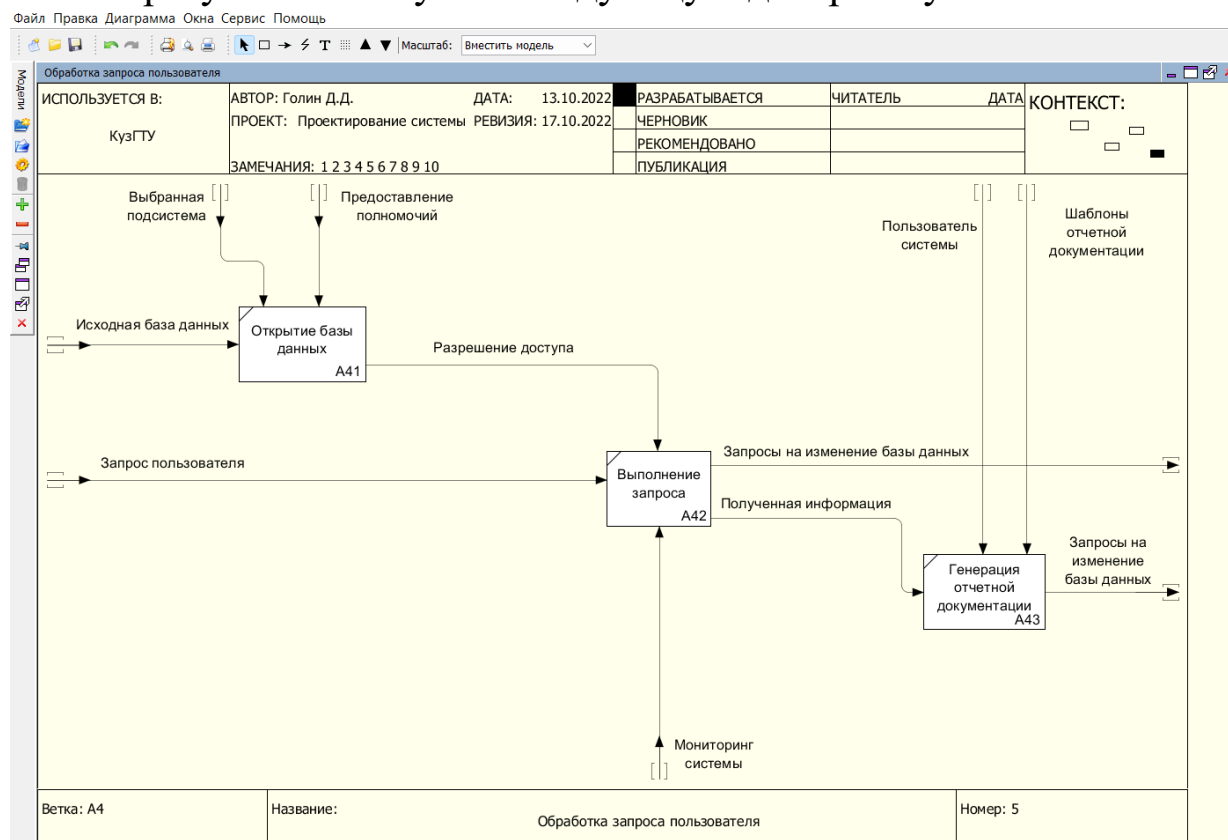


Рисунок 14 – Диаграмма декомпозиции «Обработка запроса пользователя»

Корректировка родительских диаграмм

Поскольку в результате построения дочерних диаграмм были добавлены новые операции и определены новые связи управления, необходимо внести корректировки в диаграммы родительского уровня. Поэтому последовательно внесем изменения в диаграммы уровнем выше.

1. Используем кнопку  **Переход к родительской диаграмме**, чтобы перейти к диаграмме выше уровнем.

В диаграмму «Обслуживание пользователей системы» добавим стрелку управления **Шаблоны отчётной документации**, а

также создадим стрелки для внутренних связей между, которые были выявлены в дочерней диаграмме.

В итоге диаграмма примет следующий вид:

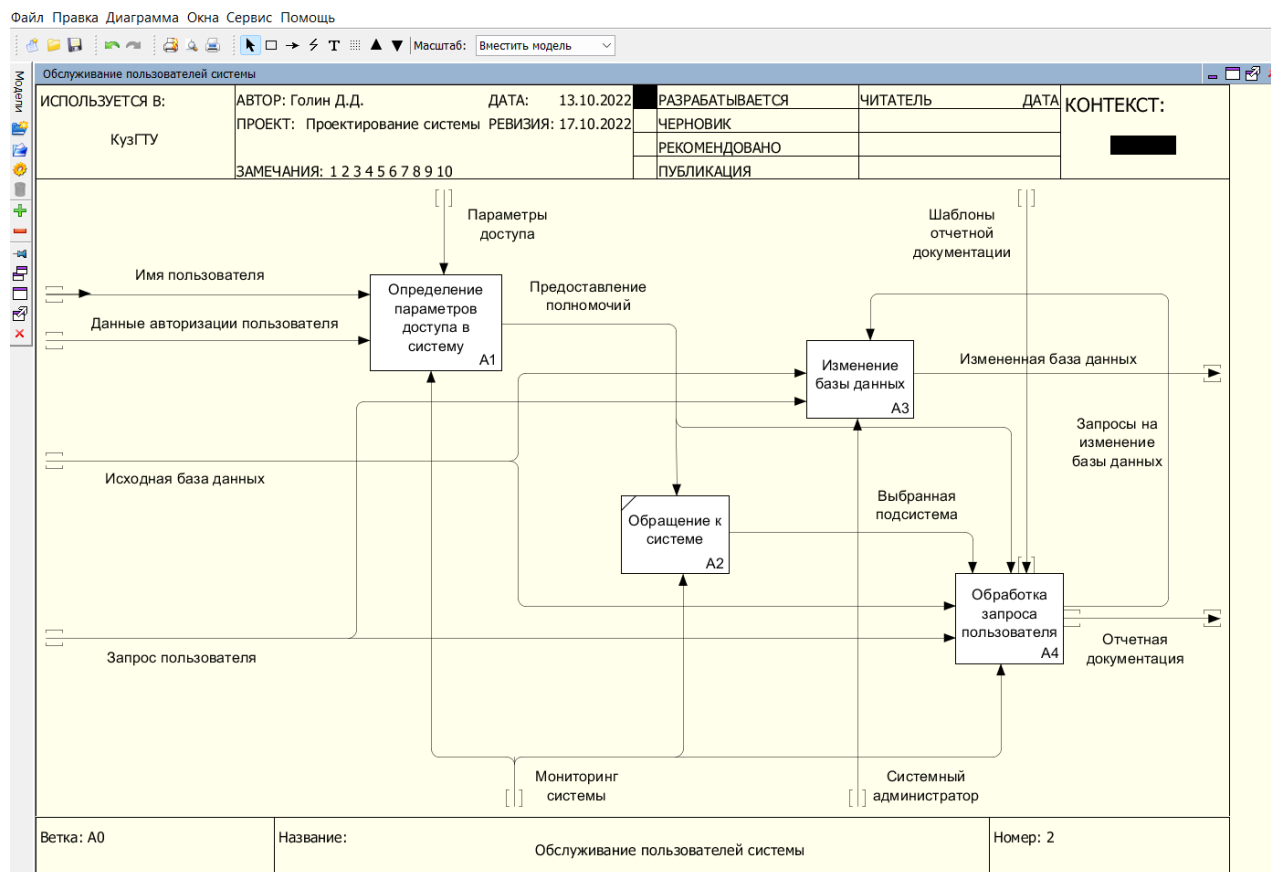


Рисунок 15 – Итоговая диаграмма декомпозиции «Обслуживание пользователей системы»

2. Также нужно внести изменения и в структурную диаграмму. С помощью той же кнопки **▲ Переход к родительской диаграмме** вернёмся к нашей первой диаграмме.

Здесь необходимо добавить лишь новую стрелку управления – **Шаблоны отчётной документации**.

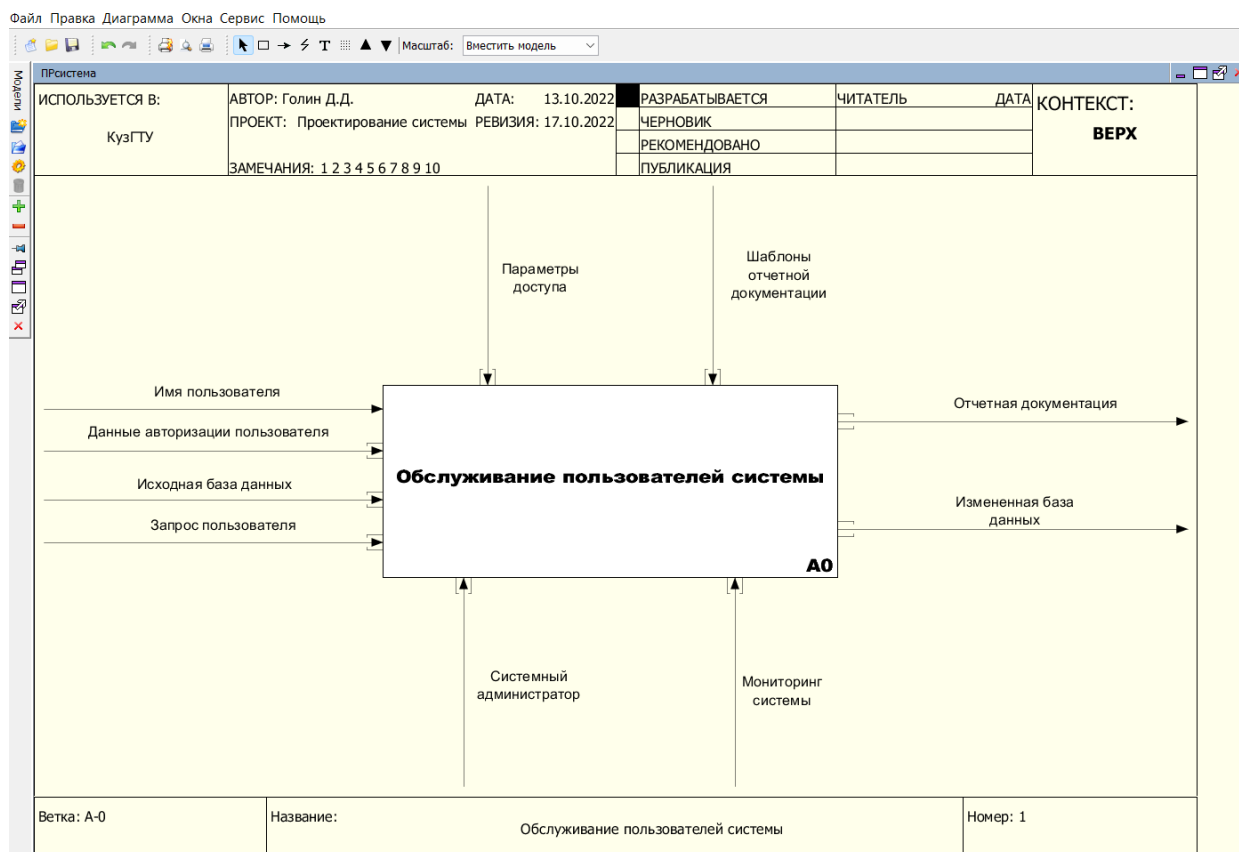


Рисунок 16 – Итоговая структурная диаграмма «Обслуживание пользователей системы»

Экспорт модели в формат IDL

1. Для того, чтобы экспортировать разработанную модель в формат IDL необходимо выбрать в меню Диаграмма и нажать соответствующую команду **Экспортировать в IDL** (Рисунок 17).

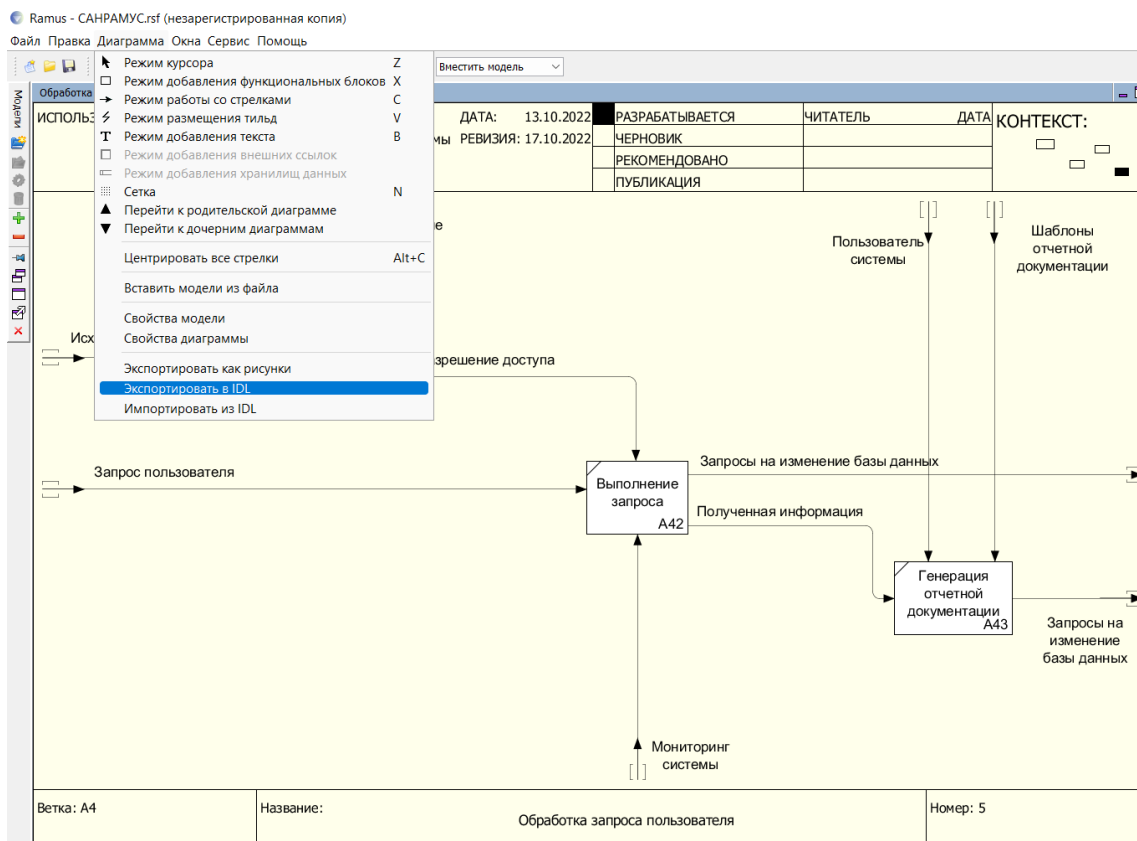


Рисунок 17 – Экспорт модели в формате IDL

2. В появившемся диалоге окне необходимо выбрать нужную нам модель, при необходимости переименовать файл и нажать кнопку **Сохранить** (Рисунок 18).

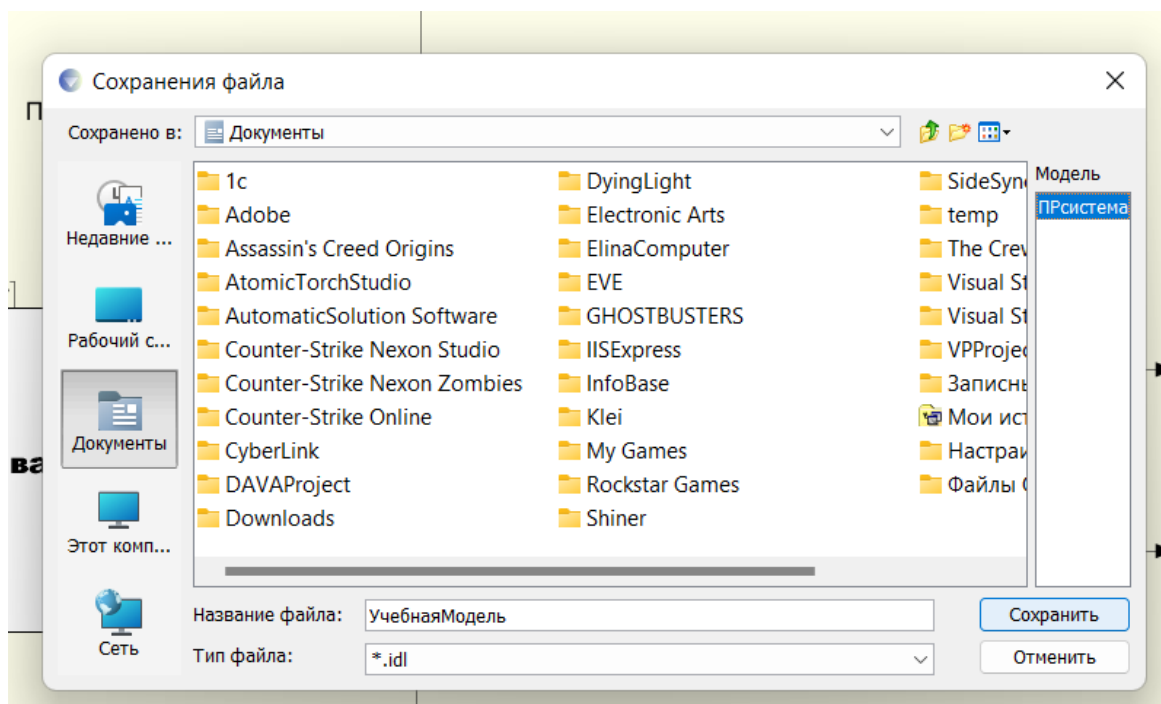


Рисунок 18 – Сохранение файла тип *.idl

3. Теперь полученный файл можно открыть с помощью соответствующих программных комплексов, которые поддерживают формат IDL, среди которых одним из самых популярных является *Microsoft Visual Studio*. Запустим файл.

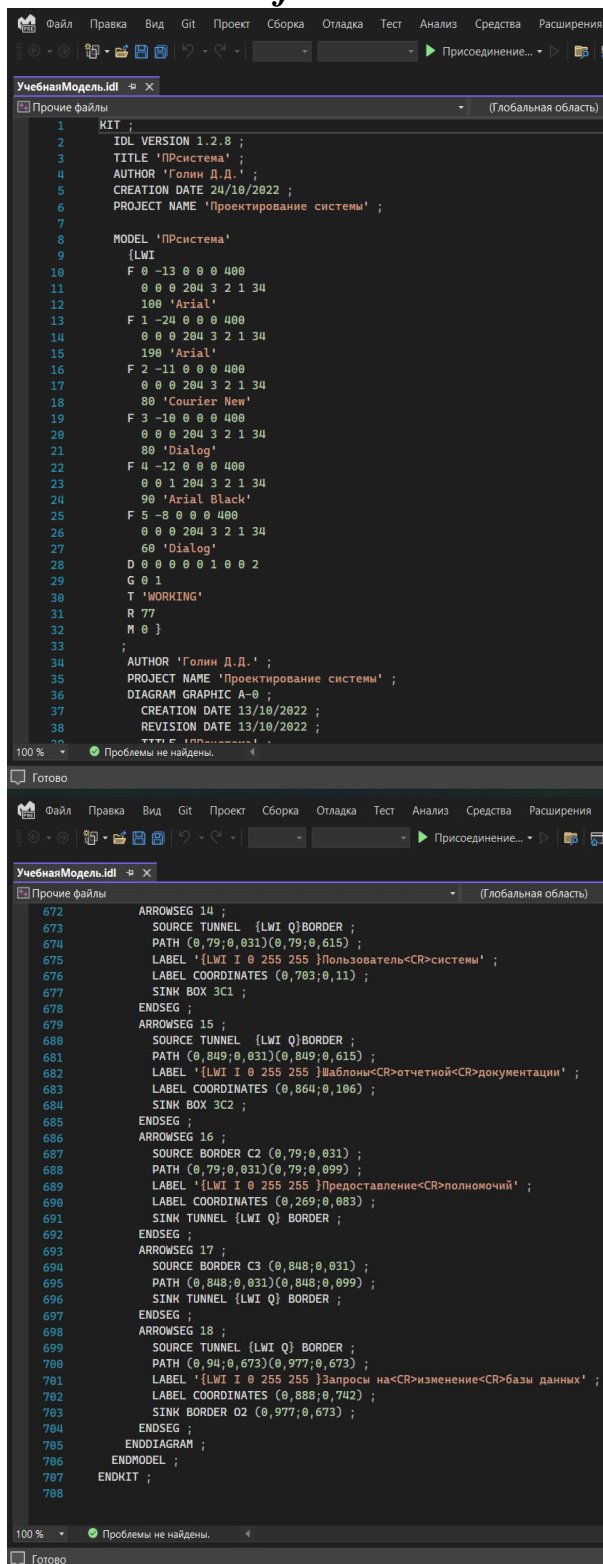


Рисунок 19 – Учебная модель в виде файла IDF

По результатам проделанной работы получим в программном модуле итоговый файл модели ИС «Пользовательская система», разработанный с помощью программы Ramus Educational. Данная модель выступает в качестве основы проектируемой системы, которую можно использовать для реализации ИС или внесения возможных изменений и корректировок, а также дальнейшего изучения и анализа поведения системы.

4. ПРАКТИЧЕСКОЕ ЗАДАНИЕ

1. Выбрать и согласовать с научным руководителем информационную систему, моделирование которой будет произведено с помощью CASE – технологии.

2. Перечислить требования выбранной системы, её функциональные возможности и цель, сформировать цели моделирования.

3. Спроектировать выбранную ИС в Ramus Educational или в любом другом аналоговом программном продукте. Построить контекстную диаграмму IDEF0, произвести декомпозицию для описания процессов функционирования системы.

4. Экспортировать разработанную модель в формат IDL. Запустить полученный файл в соответствующем ПО, где можно будет реализовать спроектированную систему.

5. Сформировать отчёт о проделанной работе.

5. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Основные этапы проектирования ИС?

2. Что такое CASE-технологии, в чём заключаются их основные особенности?

3. Перечислите известные вам программные продукты, которые можно отнести к CASE-средствам?

4. Чем является Ramus Educational? Перечислите его функциональные возможности.

5. Алгоритм создания нового проекта в Ramus Educational?

6. Какие диаграммы можно создать в Ramus Educational?

7. Алгоритм декомпозиции процесса в Ramus Educational?

8. Как происходит переход между родительскими и дочерними диаграммами?

9. Как можно импортировать/экспортировать IDL файл в Ramus Educational?

СПИСОК РЕКОМЕНДУЕМОЙ ЛИТЕРАТУРЫ

1. Гвоздева, Т. В. Проектирование информационных систем : учеб. пособие для студентов вузов, обучающихся по специальности "Прикладная информатика" / Т. В. Гвоздева, Б. А. Баллод. – Ростов н/Д: Феникс, 2009. – 508 с. 4 81057 Ипатова, Э. Р. Методологии и технологии системного проектирования информационных систем [Электронный ресурс]: учебник / Э. Р. Ипатова, Ю. В. Ипатов. – Москва.: Флинта, 2008. – 256 с.

2. Кравец, О. Я. Практикум по проектированию информационных систем: учебн.-метод. пособие для вузов / О. Я. Кравец, С. А. Олейникова. – Воронеж: Научная книга, 2006. – 208 с.

3. Объектно-ориентированный анализ и проектирование с примерами приложений : / Г. Буч [и др.; пер. с англ. и ред. Д. А. Ключина]. – Москва.: Вильямс, 2010. - 720 с.

4. Вендров, А. М. Case-технологии : современные методы и средства проектирования информационных систем. – Москва.: Финансы и статистика, 1998. – 176 с.

ЛАБОРАТОРНАЯ РАБОТА №5. СПЕЦИАЛИЗИРОВАННЫЕ ПРОГРАММА УПРАВЛЕНИЯ ЖИЗНЕННЫМ ЦИКЛОМ: MATHCAD

1. ЦЕЛЬ РАБОТЫ

Целью данной работы является получение общих теоретических знаний об управлении жизненным циклом и практических навыков их реализации в специализированной системе MathCAD.

2. ТЕОРЕТИЧЕСКОЕ ПОЛОЖЕНИЕ

2.1 Основы работы в MathCAD

MathCAD является универсальным средством решения математических задач в разных науках и технике, которое позволяет описывать решения с помощью обычных математических формул и знаков.

В MathCAD входят несколько объединенных компонентов: мощный текстовый редактор, вычислительный процессор, символьный процессор, который является системой искусственного интеллекта, и хранилищем огромного объема справочной информации, содержащей математическую и инженерную информацию в виде электронных книг.

MathCAD также содержит множество встроенных функций, которые позволяют решать задачи и системы выполнять внедрение, дифференцирование, линейное программирование и другие операции.

2.2 Установка и первый запуск MathCAD

Переходим на официальный сайт <https://www.mathcad.com> и переходим в раздел **TRY AND BUY**:

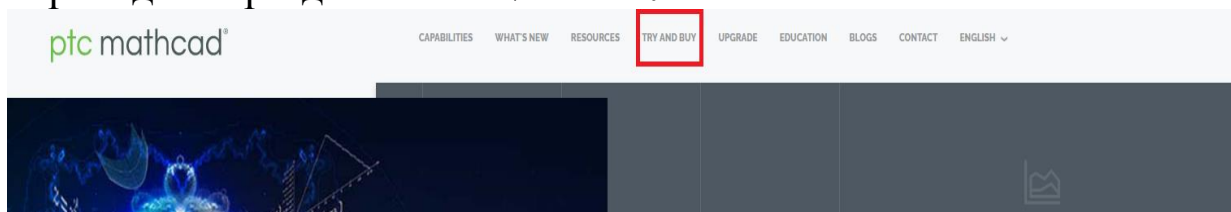


Рисунок 1 – Главная страница mathcad.com

Далее переходим в раздел **Download Mathcad Express** для получения пробного периода на 30 дней.

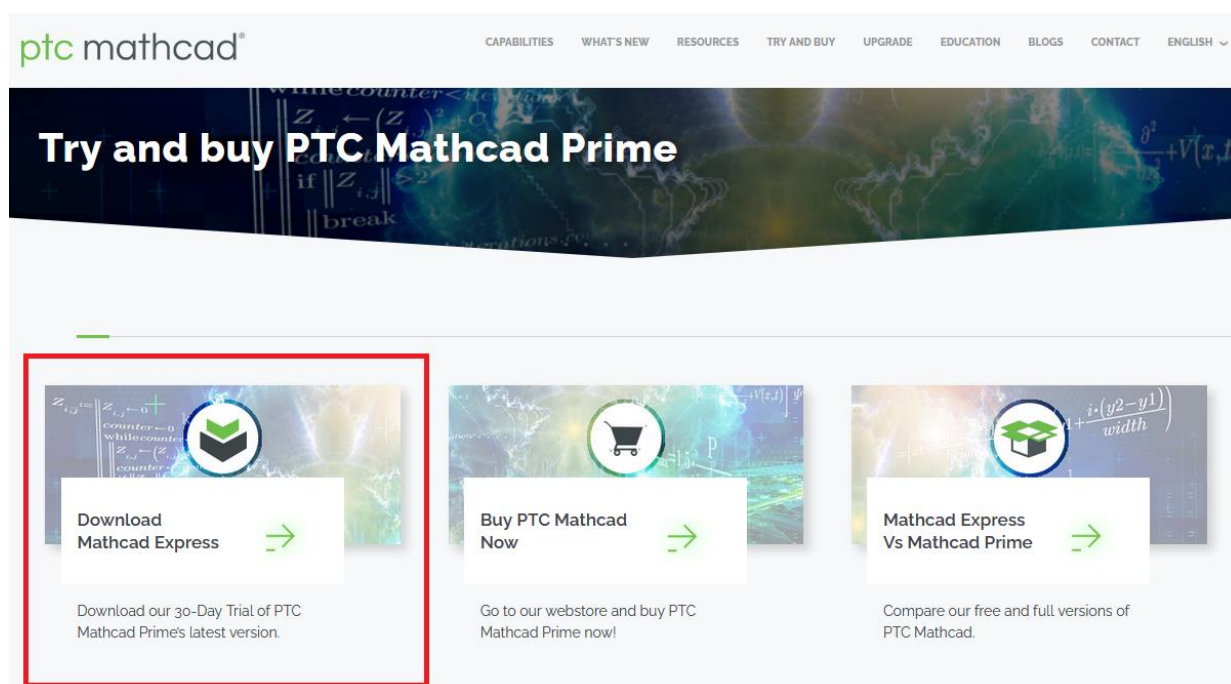


Рисунок 2 – Страница приобретения лицензии MathCAD

Проходим процедуру регистрации для последующей загрузки дистрибутива.

Рисунок 3 – Окно регистрации в MathCAD

После успешной регистрации нас отправят на страницу с уведомлением о успешном завершении регистрации.

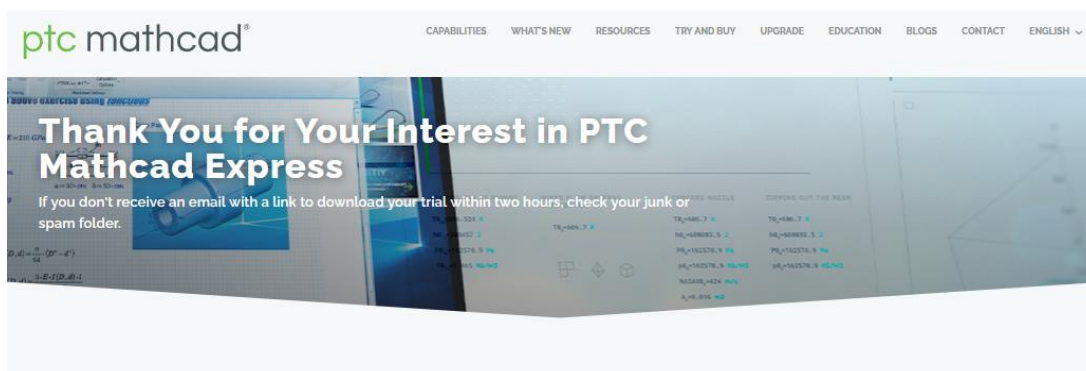


Рисунок 4 – Сообщение об успешной регистрации.

Далее переходим на электронную почту и скачиваем дистрибутив MathCAD:

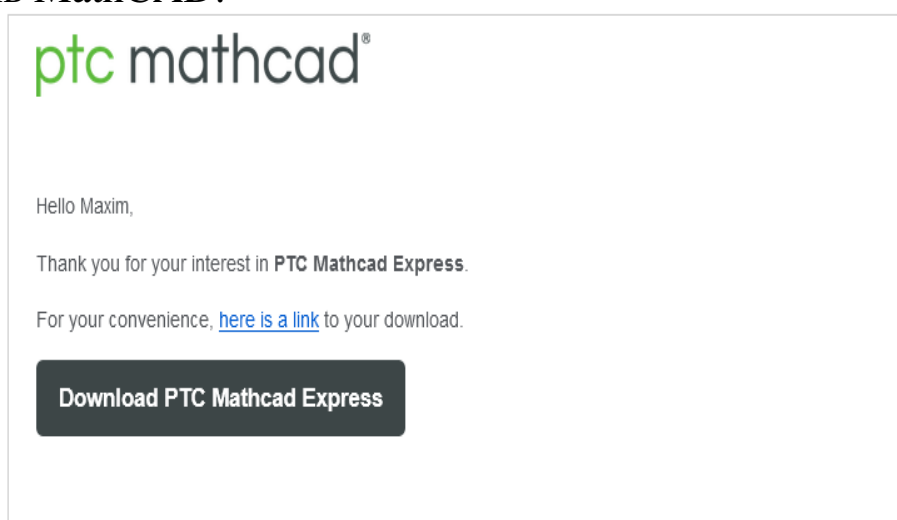


Рисунок 5 – Уведомление для скачивания дистрибутива

После скачивания файла запускаем процесс установки MathCAD:

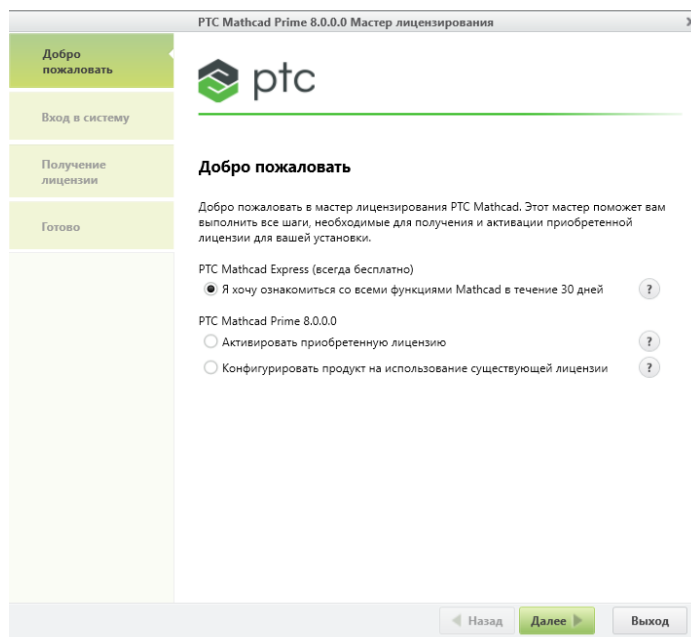


Рисунок 6 – Диалоговое окно установки MatchCAD

После завершения установки запускаем MathCAD и можем приступить решению поставленных задач.

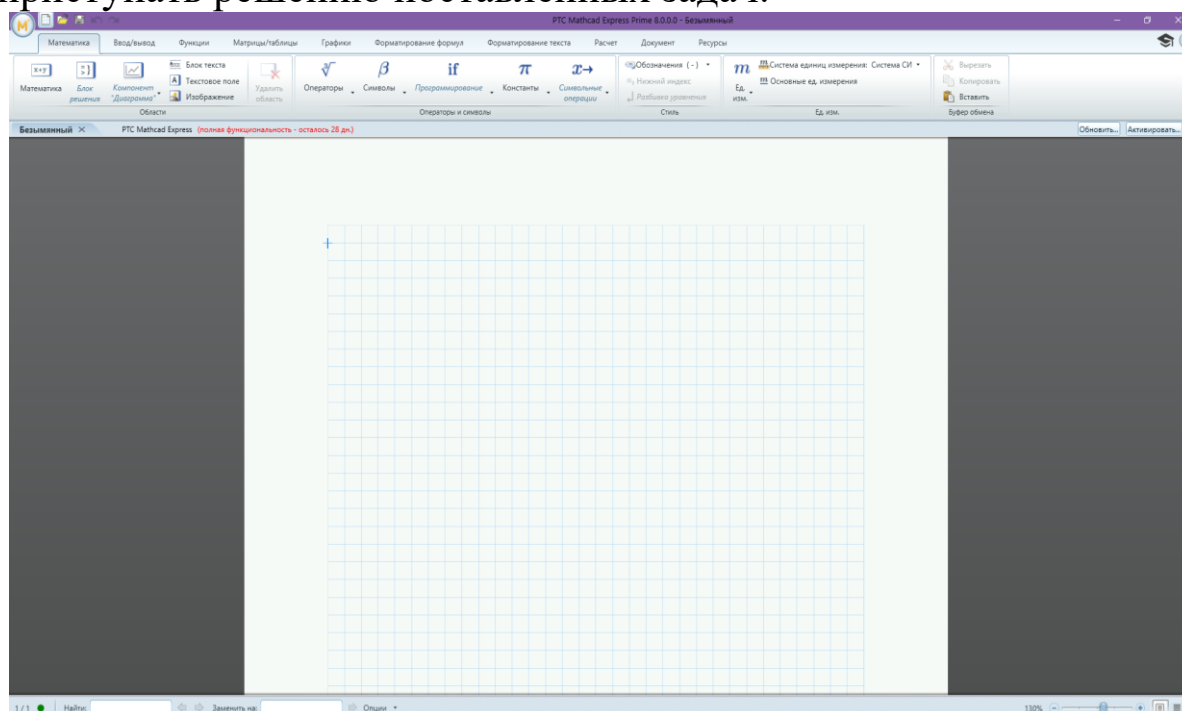


Рисунок 7 – Рабочее пространство MathCAD

2.3 Использование MathCAD для моделирования технологических процессов

Используя программу MathCAD смоделируем решение для следующей задачи: существует реакция $SO_2 + 0.5O_2 \rightleftharpoons SO_3$ ко-

торая протекает в реакторе при температурах 500-700°C. Газ, поступающий в реактор, будет содержать SO_2 , O_2 , N_2 . Содержание SO_2 – **a** мольной долей, O_2 – **b** мольной долей, остальное N_2 .

$a = 0.075$ содержание SO_2 в исходной смеси, мольные доли;

$b = 0.115$ – содержание O_2 в исходной смеси, мольные доли;

$P = 1$ содержание N_2 ;

$E = 87800$ Дж/моль – энергия активации прямой реакции;

$R = 8.314$ Дж(моль*К) – универсальная газовая постоянная;

$k_0 = 0.0302 * 10^7$ – предэкспоненциальный множитель, 1/с.

Математическое описание процесса в ректоре окисления SO_2 будет выглядеть следующим образом:

Константа равновесия рассчитывается по следующей формуле:

$$Kp(T) = 10^{\left(\frac{4905.5}{T} - 4.6455\right)}, \#(1)$$

Для расчета константы скорости реакции используем следующую формулу:

$$kl(T) = k_0 * e^{\frac{-E}{R*T}}, \#(2)$$

Уравнение скорости реакции выглядит следующим образом:

$$r(x, T) = 1 + \frac{kl(T) * P}{a} * \frac{1 - x}{1 - 0.2x} * \left[\frac{b - 0.5a * x}{1 - 0.5a * x} - \frac{x^2}{P * (Kp(T))^2 * (1 - x)^2} \right],$$

где x – степень превращения SO_4

3. ПРАКТИЧЕСКАЯ ЧАСТЬ

Каждый студент выполняет работу самостоятельно на персональном компьютере. Исходные данные представлены в Таблице 1. Вариант выбирается по номеру студента в списке группы.

Таблица 1 – Варианты исходных данных

Варианты	Значение параметров		
	Исходные данные		
	a, %	b, %	E Дж/моль
1	11	10	89000
2	5	14	88800

3	11	9	88600
4	7	11	88400
5	8	10	88200
6	9	8	88000
7	8	13	87800
8	7,5	12	89400
9	7	10	89200
10	6	13	89000
11	10	6,7	88800
12	6	12	88600
13	8	9	88400
14	11	8	88000
15	8	11	88200
Пример	7,5	11,5	87800

Задания:

1. Изучите влияние температуры в интервале 400-660 °С на значение константы равновесия.

2. Постройте график зависимости равновесной степени превращения от температуры.

3. Постройте график зависимости скорости реакции при заданном значении температуры от степени превращения в интервале 0,01 – 0,95.

4. Сформируйте отчёт о проделанной работе. В отчете должны содержаться: тема лабораторной работы, фамилия и группа студента, исходные данные, результат проделанной работы, выводы.

Ход выполнения задания будет продемонстрирован на примере первого задания.

Чтобы изучить влияние температуры на значение константы равновесия перейдем в программу MatchCAD и введем исходные данные.

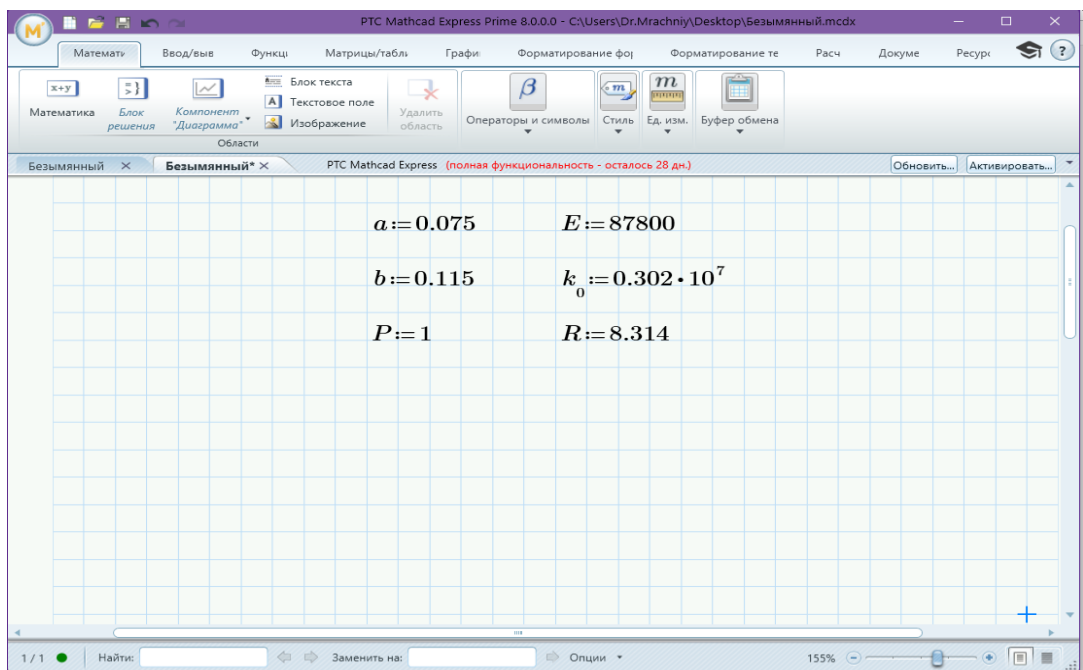


Рисунок 8 – Ввод исходных данных

Для изучения влияния температуры на значения константы равновесия будем использовать формулу (1), а также зададим температуре T нужные значения.

Решение должно принять следующий вид:

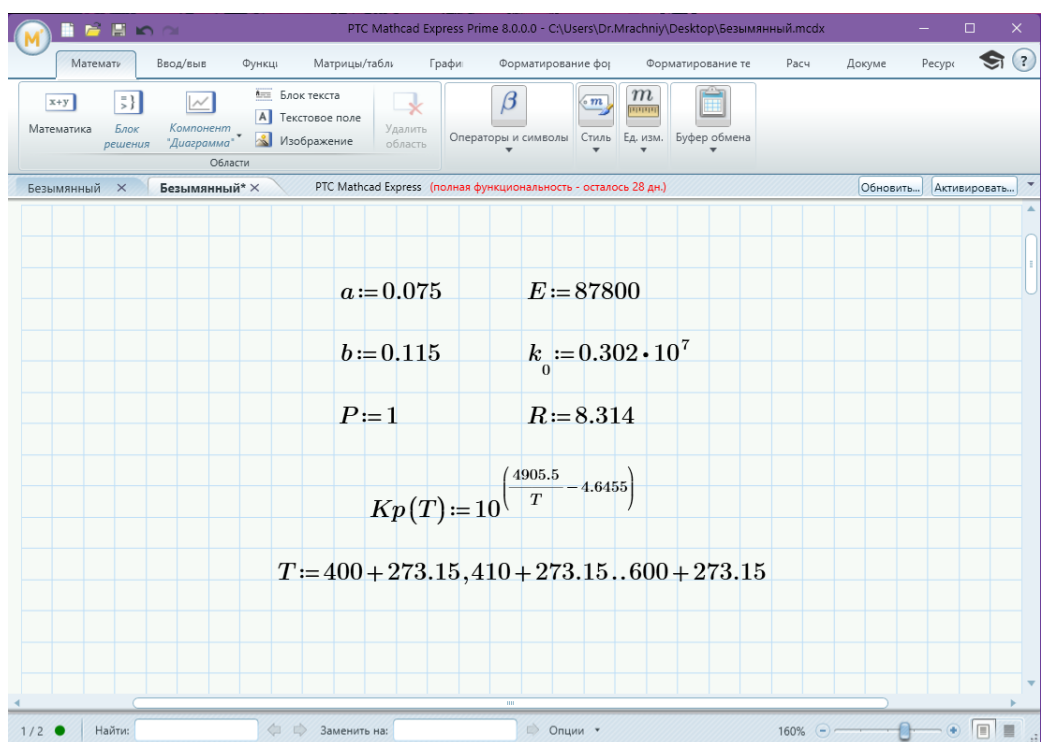


Рисунок 9 – Ввод формул для расчетов

Для наглядного отображения влияния температуры добавим график и присвоим ему значения координат. Для этого перейдем в раздел **График** → **Вставить график** → **График XY** или сочетанием горячих клавиш "**Ctrl**" + "**2**".

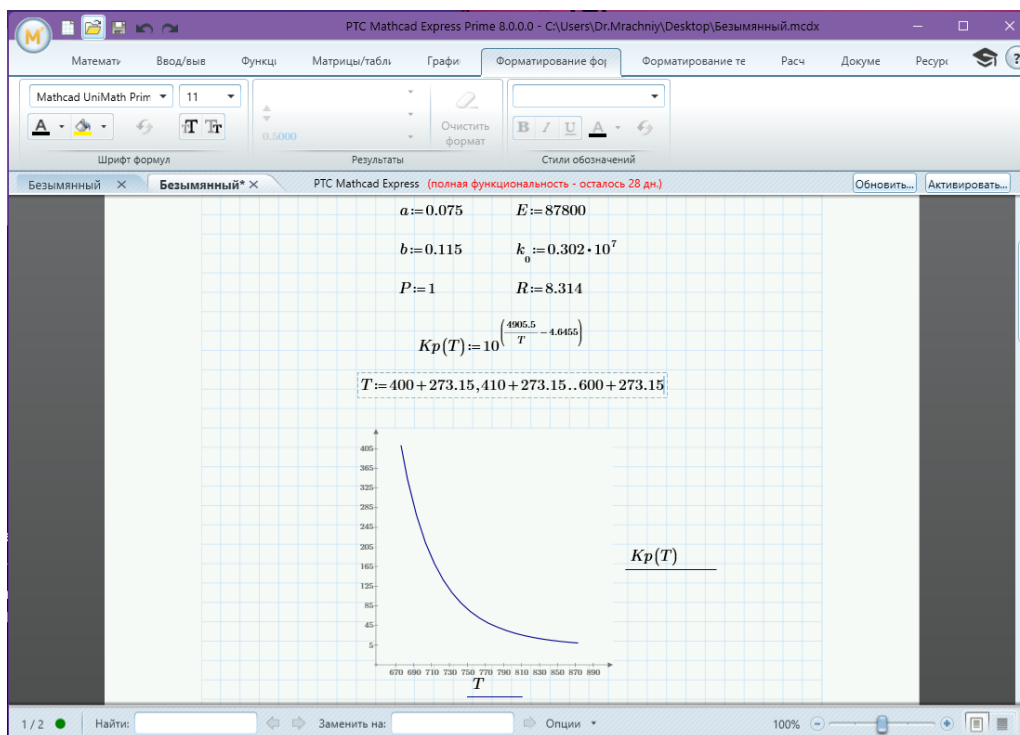


Рисунок 10 – Решение и график влияния температуры на константу равновесия

Вывод: при повышении температуры от 400 до 660 °С константа равновесия уменьшается почти в 100 раз.

4. ВОПРОСЫ ДЛЯ ЗАЩИТЫ РАБОТЫ

1. С помощью какого оператора можно вычислить выражение?
2. Как вставить текстовую область в документ MathCad?
3. Какие виды встроенных функций в MathCad Вам известны?
4. Как вставить встроенную функцию в документ MathCad?
5. Как построить график в среде MathCad?
6. Что представляет собой модель технологического объекта?
7. Перечислите методы составления математического описания объекта.
8. Какие группы уравнений входят в состав математического описания технологического объекта?

9. Какие уравнения применяют в математическом описании технологического объекта?
10. Назовите основные стадии построения математической модели технологического объекта.

СПИСОК РЕКОМЕНДУЕМОЙ ЛИТЕРАТУРЫ

1. MathCad // Санкт-Петербургский государственный электротехнический университет «ЛЭТИ» им. В. И. Ульянова (Ленина) – сайт. [Электронный ресурс]. – URL: <https://etu.ru/ru/obrazovatel'naya-deyatelnost/centr-kompetenci-y-ptc/mathcad>
2. В. В. Фриск. Mathcad. Расчеты и моделирование цепей на ПК. Москва: Солон-Пресс, 2006. ISBN: 5-98003-242-8
3. Система MathCAD в инженерной практике: Лаб. работы / Сост.: А. Ю. Сенкевич, А. А. Чуриков: – Тамбов: Изд-во Тамб. гос. техн. ун-та, 2003. – 28 с.
4. Васильев, Алексей Mathcad 13 на примерах / Алексей Васильев. – Москва.: БХВ-Петербург, 2006. – 228 с