

## Содержание

Введение	3
1. Анализ предметной области	4
1.1. Предметная область	4
1.1.1. Описание предметной области	4
1.1.2. Организационная структура	5
1.1.3. Основные бизнес-процессы	Error! Bookmark not defined.
1.2. Схемы бизнес-процессов предприятия	7
1.2.1. Схема исполнения заказа	7
1.2.2. Схема обработки заявки	8
1.2.3. Схема проектирования	9
1.3. Диаграмма Ганта	10
1.4. Анализ существующих аналогов ПО	11
1.5. Определение требований к приложению	12
2 Проектирование приложения	20
2.1 Обоснование выбора средств разработки	20
2.2 Концепция разработки	21
2.3 Структура приложения	23
3 Разработка приложения	28
3.1 Разработка БД	28
3.2 Разработка пользовательского интерфейса	39
4 Инструкции	48

					КП 09.02.07.21И1.09.000 ПЗ							
Изм.	Лист	№ докум.	Подпись	Дата								
Разраб.		Козин Д.							Лит.	Лист	Листов	
Провер.		Трифорова Э.М.							✓			
									КАТК зр. 21И1			
Н. контр.												
Утв.												

КАТК эр. 21И1

4.1	Инструкции программисту .....	<b>Error! Bookmark not defined.</b>
4.2	Инструкции пользователю .....	48
4	Безопасность и надежность функционирования информационной системы.....	50
5.	Охрана труда .....	52
	Заключение .....	55
	Информационные источники .....	56

		Козин Д			<b>КП 09.02.07.21И1.09.000 ПЗ</b>	Лист
Изм.	Лист	№. докум.	Подпись	Дата		2

## Введение

Опыт ведущих предприятий показывает, что ключ к успешной деятельности предприятия – это четкая организация управления бизнес-процессами предприятия.

В современном мире существует множество способов визуализации и управления рабочим процессом. Одним из них является Канбан.

Канбан (Kanban) — часть agile-философии и метод улучшения рабочих процессов путём визуализации и активной работы над незавершёнными задачами. Для иллюстрации хода выполнения проекта используются карточки, которые обозначают задачи, и столбцы, обозначающие этапы рабочего процесса.

Актуальность выбранной темы обусловлена ростом тенденции использования информационных систем на предприятиях с целью повышения их конкурентоспособности и эффективности деятельности.

Целью курсового проекта является проектирование и разработка виртуальной доски Канбан для компании занимающейся разработкой или поддержкой программного продукта, с целью визуализации деятельности работы и упрощения принятия управленческих решений, поиска мест улучшений.

Исходя из вышесказанного, основными задачами курсового проекта являются:

- Анализ предметной области;
- Анализ существующих аналогов ПО;
- Определение требований к приложению;
- Проектирование приложения;
- Разработка приложения;
- Тестирование приложения;
- Создание инструкций по работе с системой.

		Козин Д			<b>КП 09.02.07.21И1.09.000 ПЗ</b>	Лист
Изм.	Лист	№. докум.	Подпись	Дата		3

## 1. Анализ предметной области

### 1.1. Предметная область

#### 1.1.1. Описание предметной области

Компания занимается разработкой и поддержкой программного обеспечения (ПО)

Компания производит программный продукт и его новые версии

Основной целью организации является производство и поддержка произведенного ПО.

Для реализации поставленных задач создаются микро-группы в зависимости от направлений проектов, назначаются руководители временных рабочих групп, в состав каждой группы входят профильные специалисты в зависимости от направления проекта

Компания поделена на 4 подразделения: Разработка, Тестировка, Управление, Дизайн. В офисе работают все подразделения, которые занимаются:

- 1) Разработка – занимается разработкой ПО, расширением функционала ПО;
- 2) Тестировка – занимается проверкой правильности и корректности работы ПО в ходе внедрения нового функционала и его изменения;
- 3) Управление – занимается курированием работы компании;
- 4) Дизайн – занимается созданием моделей, картинок, фонов и т.д. для ПО.

Исходя из этого, основными функциями организации являются:

- Разработка ПО;
- Поддержка ПО;
- Развитие ПО;

		Козин Д			<b>КП 09.02.07.21И1.09.000 ПЗ</b>	Лист
Изм.	Лист	№. докум.	Подпись	Дата		4

### 1.1.2. Организационная структура

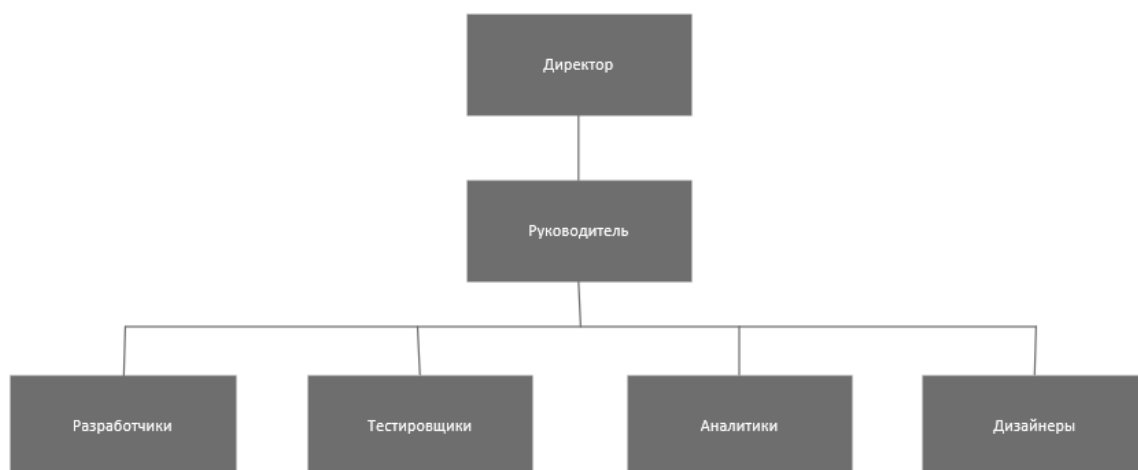


Рисунок 1 Организационная структура

Главой управления предприятием является директор. Его основная функция – это общее руководство организации. Директору подчиняются руководители групп. Основная функция руководителя групп – управление работой команды. Руководителю подчиняются разработчики, тестировщики, дизайнеры и аналитики.

Разработчики Разработчик ПО выполняет функции по созданию авторских продуктов и корректировки, трансформации действующих сервисов. Основные виды работ:

- 1) Разработка новых сервисов под конкретные цели и задачи заказчика;
- 2) Составление технического и творческого задания для дизайнеров, маркетологов и тестировщиков;
- 3) Консультирование руководителей по возможностям автоматизации отдельных бизнес-процессов;
- 4) Разработка математических моделей.

Тестировщики – задача тестировщика заключается в обнаружении ошибок и дефектов в программном обеспечении, а также в проверке его работоспособности и соответствия заявленным требованиям.

Бизнес-аналитик исследует и описывает непосредственно бизнес-процесс, а также фиксирует требования, исходящие от заказчика. Системный аналитик преобразует описание бизнес-процесса в четкий формализованный документ требований к системе и описывает функциональные модули.

Исходя из описания предметной области и организационной структуры, основными бизнес-процессами организации являются:

- Выпуск ПО;
- Разработка новых версий ПО;
- Улучшение ПО.

		Козин Д			<b>КП 09.02.07.21И1.09.000 ПЗ</b>	Лист
						6
Изм.	Лист	№. докум.	Подпись	Дата		

## 1.2. Схемы бизнес-процессов предприятия

### 1.2.1. Схема исполнения заказа

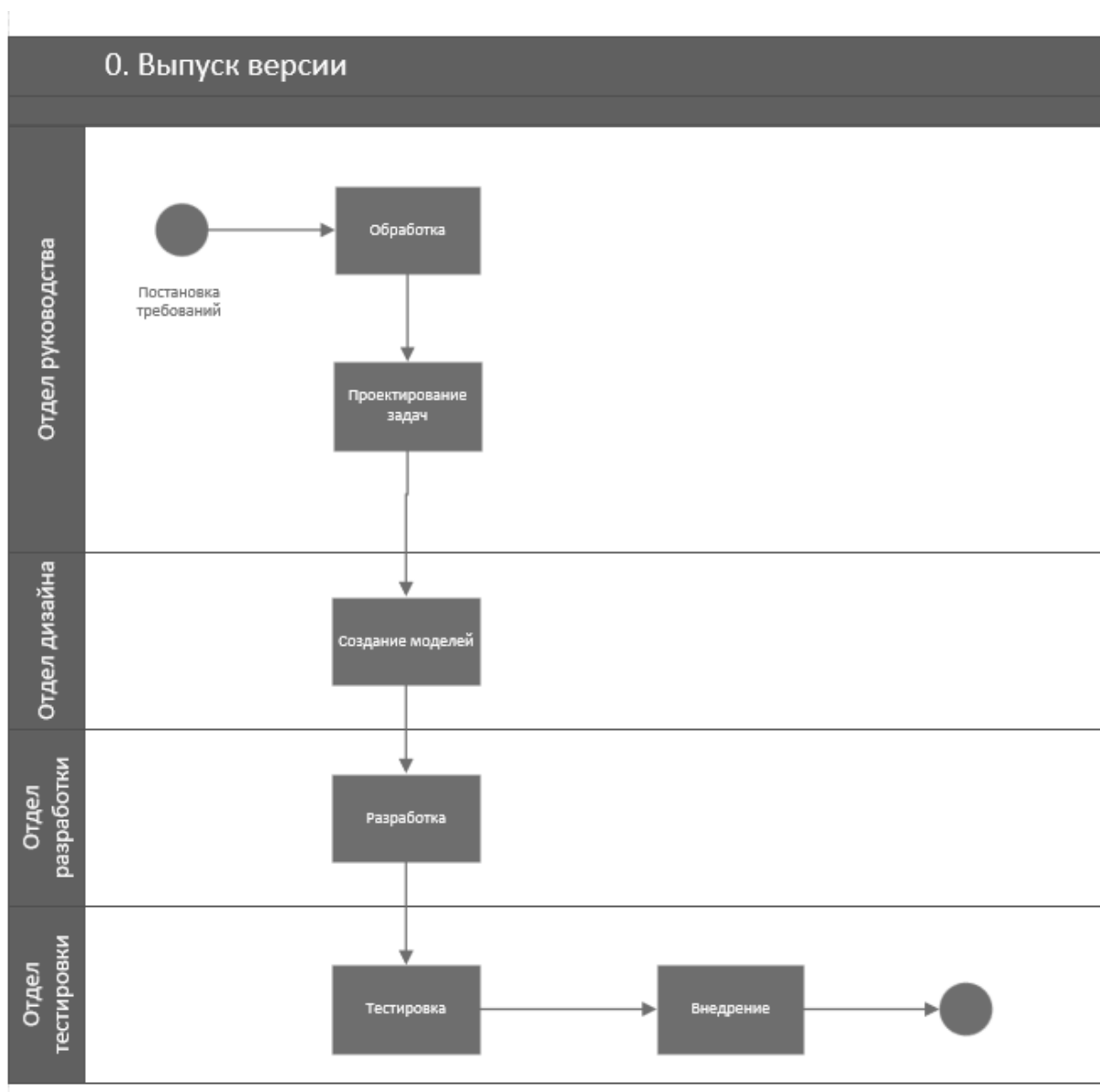


Рисунок 1 Процесс выпуска версии

Процесс «Выпуска версии» является процессом верхнего уровня и описывает работу организации в целом. Схема содержит в себе последовательность выполнения действий (подпроцессов), результатом которой является готовый продукт, полностью удовлетворяющий клиента.

В процессе «Выпуска версии» имеются подпроцессы:

- 1) Обработка (сбор всей первичной информации о пожеланиях клиента и его личных данных)
- 2) Проектирование задач (составления стека задач, для реализации проекта)
- 3) Разработка (разработка задач)
- 4) Производство (тестировка выполненных задач)
- 5) Монтаж (внедрение новой версии в ПО)

### 1.2.2. Схема подпроцесса - разработка

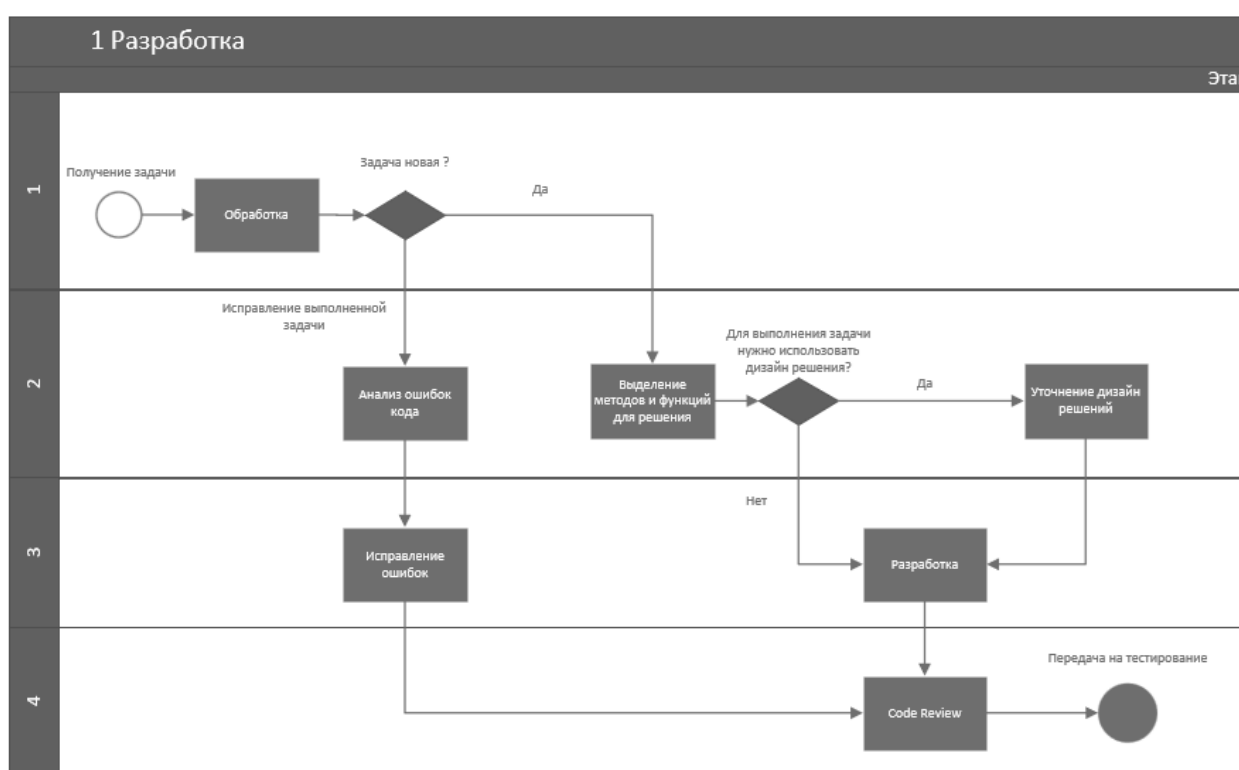


Рисунок 2 Схема подпроцесса - разработка

Подпроцесс – разработка начинается с получения задачи, после определения, новая задача или нет, наступает этап анализа задачи, определения методов решения и получения информации о используемых дизайн решениях. Далее происходит этап решения – разработки. Перед тем как разработчик отдаст задач на проверку, происходит Code Review - процесс проверки и анализа кода задачи разработчиком перед ее релизом.



### 1.2.3. Схема подпроцесса - тестирование

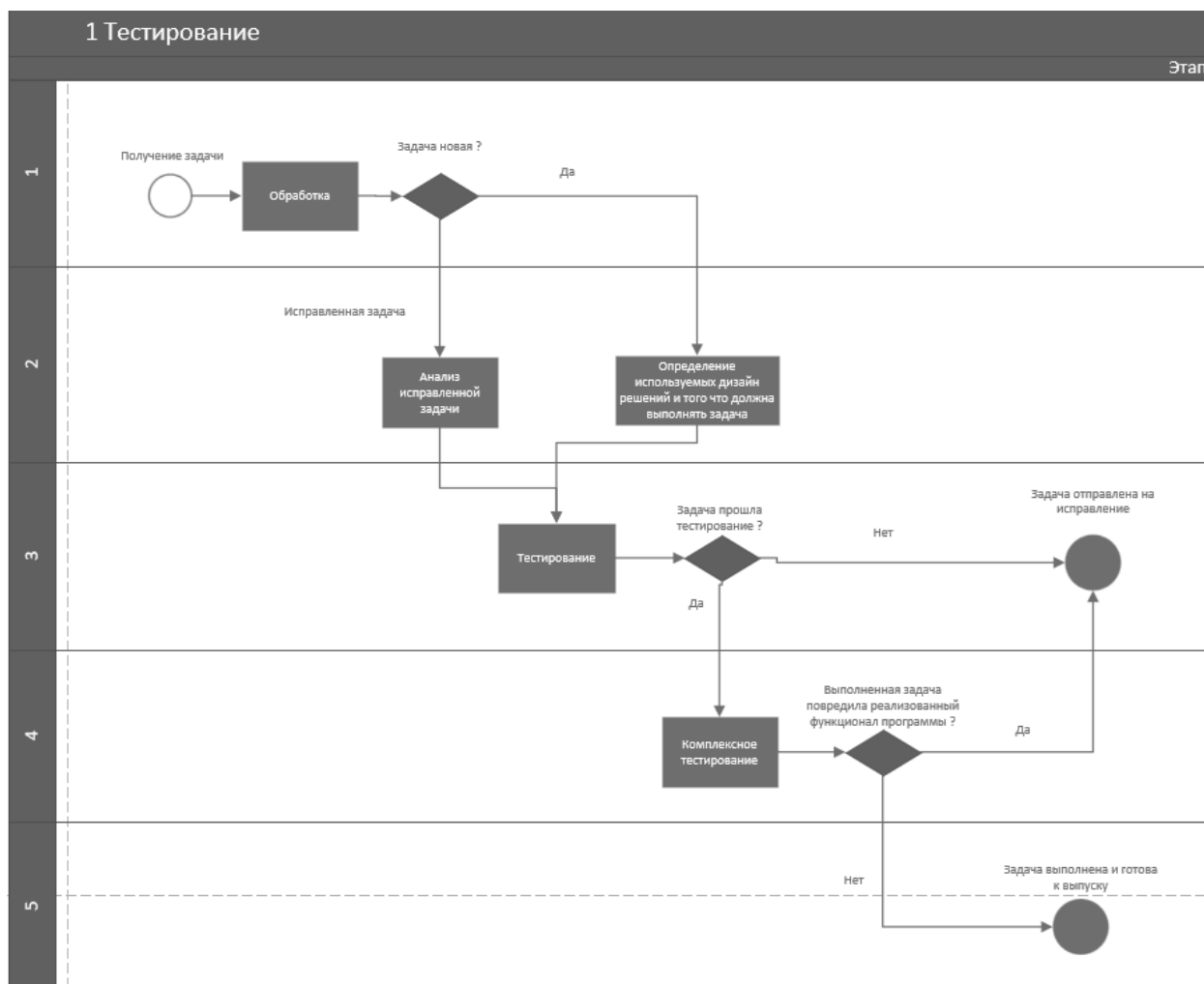


Рисунок 3 Схема подпроцесса - тестирование

Подпроцесс – тестирование начинается с получения задачи с разработки, после определения, новая задача или нет, происходит процесс анализа задачи. Сначала тестировщик тестирует исключительно функционал самой задачи, после чего переходит к комплексному тестированию. Если в процессе тестировки были выявлены ошибки, задача отправляется на исправление, если ошибок обнаружено не было, то задача считается выполненной и готова к релизу.

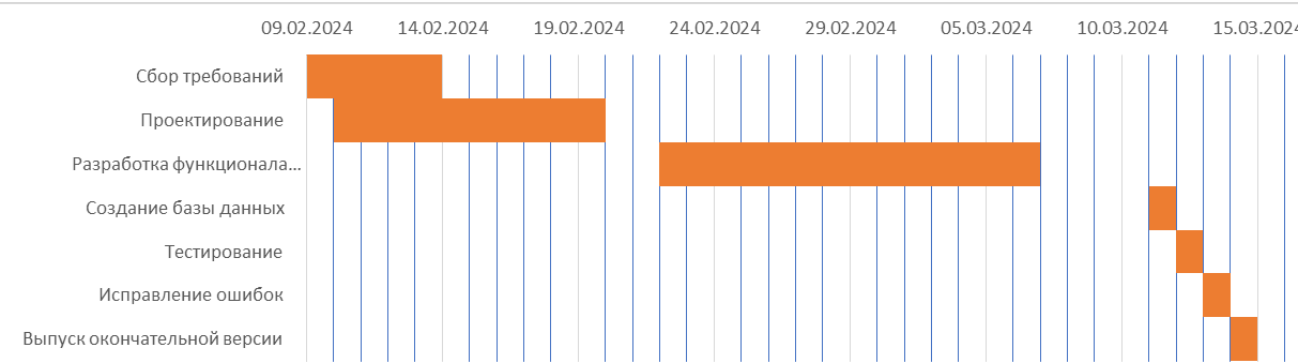
1.3. Диаграмма Ганта

Диаграмма Ганта — это визуальное представление графика работ, построенное согласно плану проекта. На ней отражены задачи и последовательность их выполнения

График работ состоит из ряда отрезков, размещённых вдоль временной оси. Каждый из них соответствует отдельной задаче или подзадаче. Начало и конец отрезка соответствуют моменту начала и завершения работы по задаче. Длина отрезка — продолжительность работ. Диаграмма Ганта нужна, чтобы наглядно представить все этапы работы.

Таблица 2 Сроки разработки проекта

Название задачи	Дата начала	дата завершения	Длительность, дни
Сбор требований	05.02.2024	09.02.2024	5
Проектирование	10.02.2024	21.02.2024	10
Разработка функционала приложения/обновление	22.02.2024	09.03.2024	14
Создание базы данных	11.03.2024	11.03.2024	1
Тестирование	12.03.2024	12.03.2024	1
Исправление ошибок	13.03.2024	13.03.2024	1
Выпуск окончательной версии	14.03.2024	14.03.2024	1



## Рисунок 5 Диаграмма Ганта

### 1.4. Анализ существующих аналогов ПО

Jira — это система управления проектами, которая позволяет закрывать почти все задачи РМ-а в рамках одного инструмента: от планирования до контроля процессов и результатов. Представляет собой комплекс из IT-решений от компании Atlassian, которые объединены воедино в Jira Family Products: Jira WM для работы с бизнес-процессами, Jira SM для построения сервис-деска, Jira Software для проектов по разработке программного обеспечения.

Основные плюсы Jira:

Широкий набор функций.

Большой интеграционный потенциал. Jira дружелюбна к интеграции с различными сервисами: Github, Salesforce, Outlook, Slack, Gmail, Teams.

Возможность расширения функционала, используя различные плагины. Наиболее популярные — Tempo, Script Runner, EazyBI, Big Picture, Structure.

Возможность работы по методологиям Scrum или Kanban «из коробки».

Недостатки Jira : длительный процесс настройки под конкретные рабочие процессы и сложный интерфейс.

GitLab:

- a. Не только система контроля версий, но и инструмент для управления проектами разработки.
- b. Предоставляет возможности для создания задач, код-ревью, управления CI/CD процессами и др.
- c. Хорошо интегрируется с Git для удобной работы с кодом.

		Козин Д			<b>КП 09.02.07.21И1.09.000 ПЗ</b>	Лист
						11
Изм.	Лист	№. докум.	Подпись	Дата		

#### d. Определение требований к приложению

##### 1.5.1. Назначение системы

Доска-Kanban предназначена для визуализации рабочих процессов организации с возможностью повышения эффективности производства и с обеспечением прозрачности получаемых и обрабатываемых данных.

Основным назначением Kanban-доски является подробная визуализация процесса работы с возможностью обработки информации о проектах, сотрудниках.

Таким образом, доска-Kanban будет незаменимым средством управления выполнения проектов в организации.

##### 1.5.2. Цели и задачи создания системы

Доска-Kanban создается с целью:

- Обеспечения сбора и первичной обработки исходной информации, необходимой для оформления проектов, определение исполнителей.
- Обеспечения наглядной прозрачности работ в организации;
- Повышения качества (полноты, точности, достоверности, своевременности, согласованности) информации;

Задачи:

- Формирование единого хранилища данных о проектах/сотрудниках/этапах выполнения проектов;
  - Введение возможности обработки информации о проектах, сотрудниках, этапах выполнения проектов;
  - Введение возможности оповещений о сроках сделок и действиях сотрудников;
  - Формирование многопользовательского доступа к системе;

		Козин Д			<b>КП 09.02.07.21И1.09.000 ПЗ</b>	Лист
						12
Изм.	Лист	№. докум.	Подпись	Дата		

- Автоматический расчет даты завершения проектов;
- Визуализация работы в методологии канбан.

### 1.5.3. Характеристика объектов автоматизации

Пользователями данной ИС на начальной стадии будут:

- Исполнители этапов проекта,
- Администратор системы,
- Руководитель организации.

### 1.5.4 Объект автоматизации

Таблица 2 Автоматизация процессов

Структурное подразделение	Наименование процесса	Возможность автоматизации	Решение об автоматизации в ходе проекта.
Руководство	Уведомления о действиях сотрудников	Возможна	Будет автоматизирована
Разработка	Расчет дат выпуска версий	Возможна	Будет автоматизирована
Руководство	Управление доступом к системе	Возможна	Будет автоматизирован
Руководство	Отслеживание стадий работы	Возможна	Будет автоматизирована

### 1.5.5 Требования к структуре и функционированию системы

В системе предлагается выделить следующие функциональные подсистемы:

– Подсистема сбора, обработки и загрузки данных, которая предназначена для реализации процессов сбора данных из систем источников, приведения указанных данных к виду, необходимому для наполнения подсистемы хранения данных;

– Подсистема хранения данных, которая предназначена для хранения данных в структурах, нацеленных на принятие решений;

– Подсистема формирования и визуализации отчетности.

#### 1.5.6 Требования к численности и квалификации персонала системы

В состав персонала, необходимого для обеспечения эксплуатации системы в рамках соответствующих подразделений, необходимо выделить следующих ответственных лиц:

– Исполнители этапов проекта;

– Руководители проектов;

– Администратор системы.

Данные лица должны выполнять следующие функциональные обязанности:

– Руководитель проекта производит ввод данных о проекта в систему и обеспечивает своевременное передвижение карточек заказов по процессам в соответствии с реальным течением процессов;

– Администратор системы контролирует доступ к системе.

#### 1.5.7 Требования к надежности системы

– Применения технических средств, системного и базового программного обеспечения, соответствующих классу решаемых задач;

– Своевременного выполнения процессов администрирования;

		Козин Д			<b>КП 09.02.07.21И1.09.000 ПЗ</b>	Лист
						14
Изм.	Лист	№. докум.	Подпись	Дата		

– Соблюдения правил эксплуатации и технического обслуживания программно-аппаратных средств.

#### 1.5.8 Требования к защите информации от несанкционированного доступа

##### 1.5.8.1 Требования к антивирусной защите

Средства антивирусной защиты, Dr.Web, Avast, Kaspersky, должны быть установлены на всех рабочих местах пользователей и администраторов. Средства антивирусной защиты рабочих мест пользователей и администраторов должны обеспечивать:

- Централизованное управление сканированием, удалением вирусов и протоколированием вирусной активности на рабочих местах пользователей;
- Централизованную автоматическую установку клиентского ПО на рабочих местах пользователей и администраторов;
- Централизованное автоматическое обновление вирусных сигнатур на рабочих местах пользователей и администраторов;
- Ведение журналов вирусной активности;
- Администрирование всех антивирусных продуктов.

##### 1.5.8.2 Требования по сохранности информации при авариях

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

- Сбой в электроснабжении сервера;
- Сбой в электроснабжении рабочей станции пользователей системы;
- Сбой в электроснабжении обеспечения локальной сети;

		Козин Д			<b>КП 09.02.07.21И1.09.000 ПЗ</b>	Лист
Изм.	Лист	№. докум.	Подпись	Дата		15

- Ошибки, не выявленные при отладке и испытании системы;
- Сбои программного обеспечения сервера.

Необходимо использовать систему резервного копирования с использованием встроенных средств или специального стороннего программного обеспечения, согласно общего регламента администрирования.

#### 1.5.8.3 Требования к составу и содержанию работ по подготовке ввода системы в эксплуатацию

Необходимые требования:

- 1) Заполнение базы данных;
- 2) Обучение персонала работе с системой;
- 3) Обеспечение сотрудников компьютерами с необходимым программным обеспечением.

#### 1.5.8.4 Требования к видам обеспечения

Требования к техническому обеспечению

Минимальные технические требования к рабочему месту пользователя представлены в таблице 5.

Таблица 3 Техническое обеспечение

ICL Моноблок ATX 450Wt P4 Black (G31M-S2L) 2*2048Mb DDR3 PC6400 /800 Kingston-256Gb 7200 SATA-II -512 Mb PCIe-16x DDR3 intel HD Graphics 3000 grap 2xDVI+TV -Out -6ch sound-LAN 10/100/1000	1	Шт
Процессор intel Pentium CPU G2020 (3000/1333/6Mb) S775 (Intel)	1	Шт
Жесткий диск 320 Gb 7200 Seagate baracuda Serial-ATA-II 16 Mb Cache	1	Шт



Память DIMM 2048 Mb DDR3 PC 6400/800 (Kingston)	2	Шт
Видеокарта(встройка) 512Mb DDR3 intel HD Graphics 3000	1	Шт
Материнская плата Gigabyte GA-G31M-S2L S775, IG31, 2xDDR2, PCIe-x16, 1-PCIe-x1,2-PCI,4-SATA II, 1-ATA100,HAD Sound, Giga LAN	1	Шт
Мышь ICL Mouse USB	1	Шт
Клавиатура ICL USB	1	Шт
Дисковод перезаписывающий NEC DVD-writer AD-7203S Black (SATA, LabelFlash,20xDVD+-R, 6xDVD-RW, 8xDVD+RW,12xDVD-RAM,12xDVD+-R9,16xDVD,48xR,48xW,32xRW)OEM	1	Шт

#### Требования к программному обеспечению

##### Системное программное обеспечение:

- ОС Windows 7 и выше;
- Антивирусные программы.

#### 1.5.8.5 Требования к надежности технических средств и программного обеспечения

##### К надежности оборудования предъявляются следующие требования:

- В качестве аппаратных платформ должны использоваться средства с повышенной надежностью;
- Применение технических средств, соответствующих классу решаемых задач;

Надежность программного обеспечения подсистем должна обеспечиваться за счет:

- Надежности общесистемного ПО и ПО, разрабатываемого Разработчиком;

– Проведением комплекса мероприятий отладки, поиска и исключения ошибок.

– Ведением журналов системных сообщений и ошибок по подсистемам для последующего анализа и изменения конфигурации.

#### 1.5.9 Источники разработки

ГОСТ 24.211-82 «Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Требования к содержанию документа «Описание алгоритма»».

ГОСТ Р 50922-2006. «Защита информации. Основные термины и определения».

ГОСТ 24.301-80 «Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Общие требования к выполнению текстовых документов».

ГОСТ 24.204-80. «Комплекс стандартов на автоматизированные системы». «Автоматизированные системы». «Требования к содержанию документа «Описание постановки задачи»».

ГОСТ 24.302-80 «Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Общие требования к выполнению схем».

ГОСТ 24.205-80 «Комплекс стандартов на автоматизированные системы». «Автоматизированные системы». «Требования к содержанию документов по информационному обеспечению»

ГОСТ 24.206-80 «Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Требования к содержанию документов по техническому обеспечению»

ГОСТ 24.209-80 «Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Требования к содержанию документов по организационному обеспечению».

ГОСТ 24.210-82 «Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Требования к содержанию документов по функциональной части».

ГОСТ 24.208-80 «Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Требования «Ввод в эксплуатацию»».

ГОСТ 7.32-2001. «Отчет о научно-исследовательской работе: структура и правила оформления».

ГОСТ 7.1-2003. «Система стандартов по информации, библиотечному и издательскому делу». «Библиографическая запись». «Библиографическое описание». «Общие требования и правила составления».

		Козин Д			<b>КП 09.02.07.21И1.09.000 ПЗ</b>	Лист
						19
Изм.	Лист	№. докум.	Подпись	Дата		

## 2 Проектирование приложения

### 2.1 Обоснование выбора средств разработки

Сегодня управление компанией может осуществляться как через коробочную программу, так и через приложение. Один из самых удобных форматов работы – это приложение, позволяющий использовать широкий функционал и в то же время упростить процесс работы.

Главными преимуществами приложения являются:

- Отсутствие нужды в установке на машину тяжеловесного программного обеспечения. Все, что требуется для полноценной работы это библиотеки, поставляемые вместе с операционной системой, и доступ в Интернет;
- Приложение не требовательно к ресурсам и не предъявляют никаких требований к аппаратной платформе.
- Нет проблем с поддержкой старых версий программ и обратной совместимостью.
- Можно работать в сети, сохранять результаты своей работы на сервере и, в случае необходимости, иметь к ним доступ отовсюду, где есть Интернет;

Следующим шагом является выбор правильного стека технологий. Стек технологий разделяется на разработку двух основных частей – клиентской и серверной.

Основными компонентами стека клиентской части являются:

- Visual Studio: Основная интегрированная среда разработки (IDE) для создания приложений под платформу .NET, включая WPF.
- XAML (Extensible Application Markup Language): Язык разметки для описания пользовательского интерфейса в WPF.

– Entity Framework (для работы с базами данных, если требуется): ORM (объектно-реляционное отображение) для работы с данными в приложениях .NET.

Таким образом, для разработки проекта был выбран следующий стек технологий: Visual Studio, XAML (Extensible Application Markup Language), Entity Framework.

## 2.2 Концепция разработки

В современном мире существует множество способов визуализации и управления рабочим процессом. Одним из них является Канбан.

Методика предполагает обсуждение производительности в режиме реального времени и полную прозрачность рабочих процессов. Рабочие задачи визуально представлены на доске, что позволяет участникам команды видеть состояние каждой задачи в любой момент времени. Основная ценность доски – анализ рабочего процесса «в целом» и поиск точек для улучшения.

Основным назначением Kanban-доски в организации является обработка данных о проектах и сотрудниках с возможностью демонстрации рабочего процесса.

В состав персонала, необходимого для обеспечения эксплуатации системы в рамках соответствующих подразделений, необходимо выделить следующих ответственных лиц:

- Исполнители этапов проекта;
- Руководитель проекта;
- Администратор системы.

Данные лица должны будут выполнять следующие функциональные обязанности:

- Руководитель проекта производит ввод данных о проекте и его исполнителей в систему и обеспечивает своевременное передвижение

		Козин Д			<b>КП 09.02.07.21И1.09.000 ПЗ</b>	Лист
Изм.	Лист	№. докум.	Подпись	Дата		21

карточек заказов по столбцам в соответствии с реальным течением рабочего процесса;

- Администратор системы контролирует доступ к системе.

		Козин Д			<b>КП 09.02.07.21И1.09.000 ПЗ</b>	Лист
Изм.	Лист	№. докум.	Подпись	Дата		22

## 2.3 Структура приложения

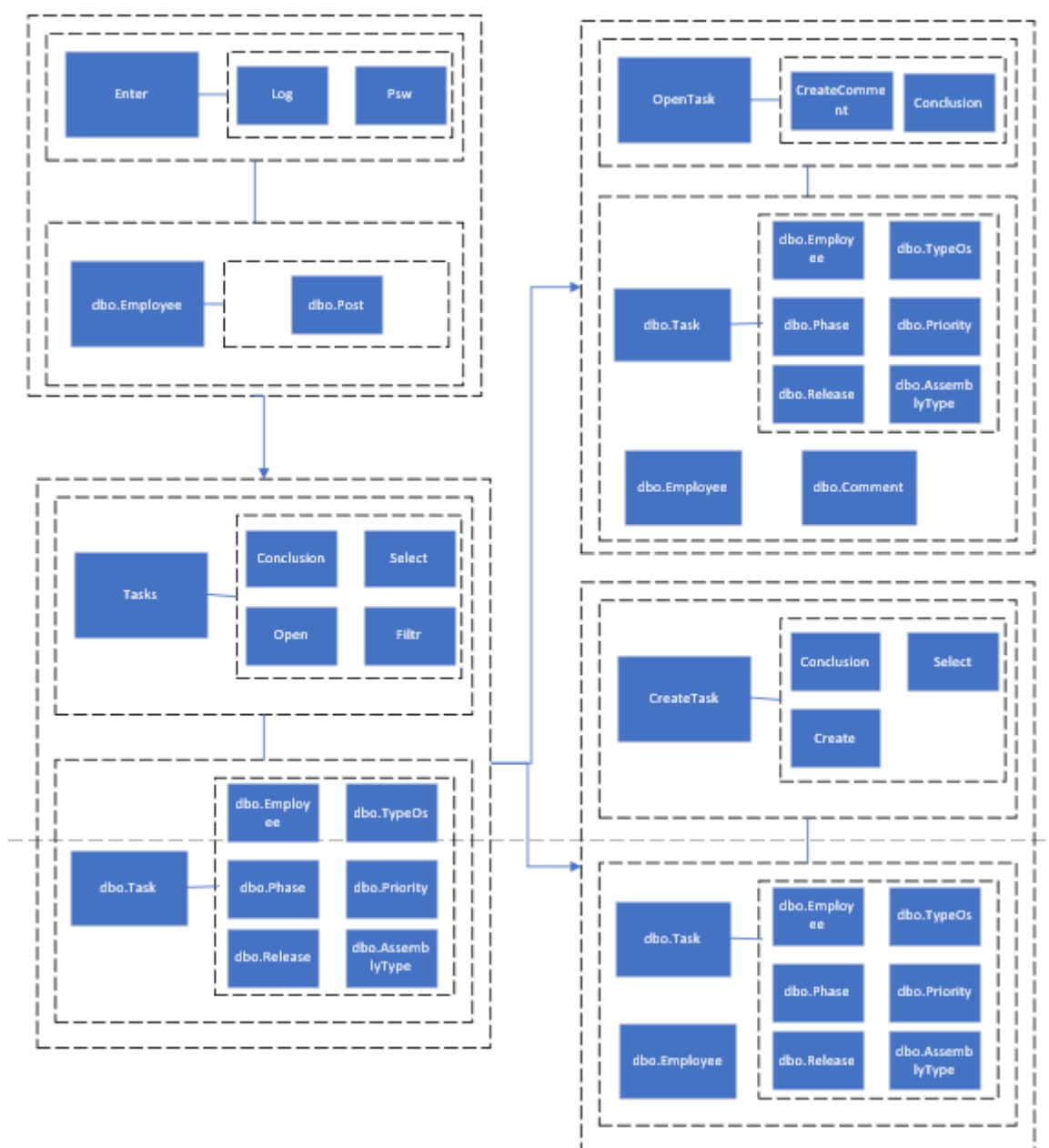


Рисунок 4 Структура приложения

### Описание структуры:

- Модуль Enter связан в таблицей Employee и включает подмодули: Log и Psw. Данный модуль отвечает за вход в систему, путем проверки логина и пароля пользователя, и переход в главное окно
- Модуль Tasks связан с таблицей Task и включает в себя подмодули: Conclusion, Select, Open, Filtr. Данный модуль отвечает за: отображение задач, их фильтрацию, выбор, открытие и переход в создание задачи.

- Модуль OpenTask связан с таблицами: Employee, Task и Comment, и включает подмодули CreateComment и Conclusion. Данный модуль отвечает за отображение открытой задачи и создания комментариев к ней.
- Модуль CreateTask связан с таблицами Task и Employee, и включает подмодули: Conclusion, Create и Select. Данный модуль отвечает за создание задач, путем выбора и прописывания элементов.

## Описание БД

- 1) Таблица «Среда разработки» Отображает среды разработки.

Таблица 4 Среда разработки

Наименование столбцов	Тип данных	Примечание
Код среды	Int	Ключевое поле
Наименование среды	nvarchar(50)	

- 2) Таблица «Стадия работы» отображает стадии работы.

Таблица 5 Стадия работы

Наименование столбцов	Тип данных	Примечание
Код стадии	Int	Ключевое поле
Наименование стадии	nvarchar(50)	

- 3) Таблица «Должность» содержит информацию о должностях пользователей.

Таблица 6 Должность

Наименование столбцов	Тип данных	Примечание
Код должности	Int	Ключевое поле
Наименование должности	nvarchar(35)	



- 4) Таблица «Приоритет» содержит информацию о приоритетах задач.

Таблица 7 Приоритет

Наименование столбцов	Тип данных	Примечание
Код приоритета	Int	Ключевое поле
Наименование приоритета	nvarchar(50)	

- 5) Таблица «Версия» содержит информацию о версиях проекта.

Таблица 8 Версия

Наименование столбцов	Тип данных	Примечание
Код версии	Int	Ключевое поле
Наименование версии	nvarchar(50)	
Дата начала	date	
Дата конца	date	

- 6) Таблица «ОС» содержит информацию об ОС.

Таблица 9 ОС

Наименование столбцов	Тип данных	Примечание
Код ОС	Int	Ключевое поле
Наименование ОС	nvarchar(50)	

- 7) Таблица «Комментарий» содержит информацию о комментариях к задачам.

Таблица 10 Комментарий

Наименование столбцов	Тип данных	Примечание
Код комментария	Int	Ключевое поле
Описание	nvarchar(255)	
Код сотрудника	int	
Код задачи	int	

- 8) Таблица «Зависимость» содержит информацию о зависимостях.

Таблица 11 Зависимость

Наименование столбцов	Тип данных	Примечание
Код зависимости	Int	Ключевое поле
Код зависимой задачи	Зависимость	
Код со зависимой задачи	int	

- 9) Таблица «Сотрудников» содержит данные пользователей

Таблица 12 Проект

Наименование столбцов	Тип данных	Примечание
Код сотрудника	Int	Ключевое поле
Фамилия	nvarchar(50)	
Имя	nvarchar(30)	
Отчество	nvarchar(50)	
Код должности	nvarchar(50)	
Аватар	nvarchar(100)	
Логин	nvarchar(20)	
Пароль	nvarchar(15)	

- 10) Таблица «Задачи» содержит информацию о выполняемых задачах

Таблица 13 Процессы

Наименование столбцов	Тип данных	Примечание
Код задачи	Int	Ключевое поле
Название	Varchar(255)	
Описание	nvarchar(1000)	
Код создателя	int	
Код разработчика	Int	
Код тестировщика	Int	
Код стадии	Int	
Код среды	int	
Код версии	Int	
Код приоритета	Int	
Код ОС	Int	
Доступ	bit	

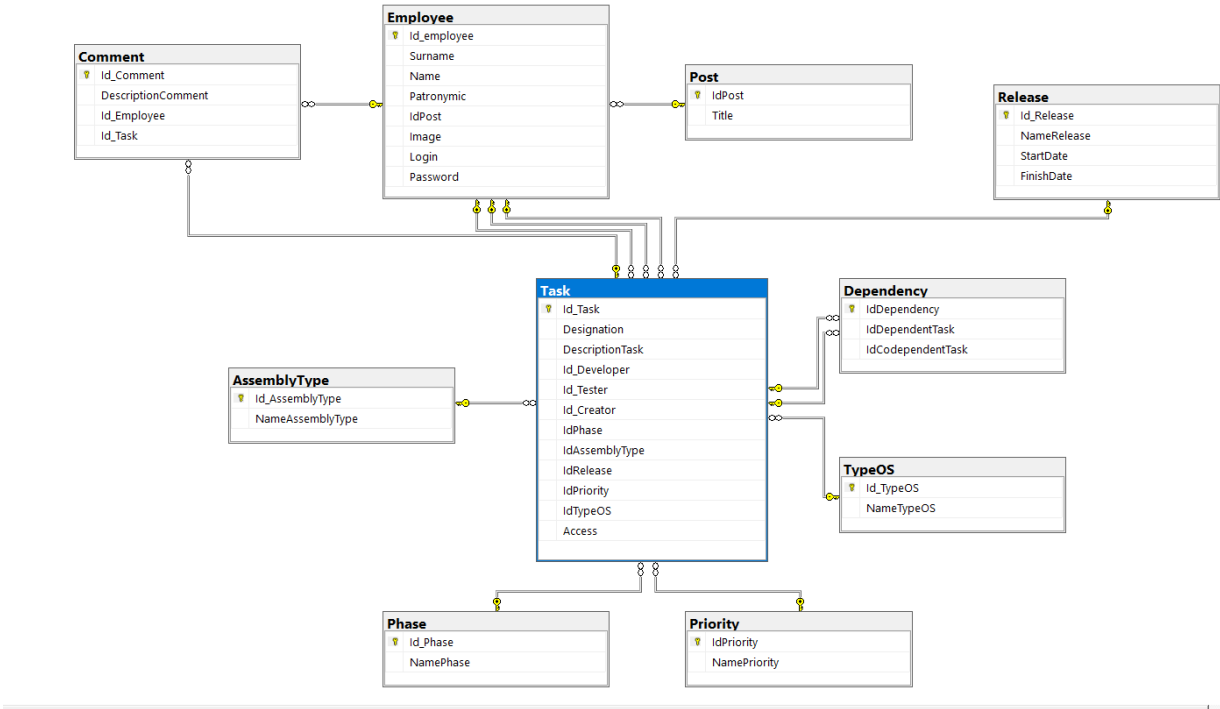


Рисунок 5 Схема базы данных

### 3 Разработка приложения

#### 3.1 Разработка БД

GO

ALTER DATABASE [Development] SET QUERY\_STORE = OFF

GO

USE [Development]

GO

/\*\*\*\*\* Object: Table [dbo].[AssemblyType] Script Date: 09.04.2024 6:28:21

\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[AssemblyType](

[Id\_AssemblyType] [int] IDENTITY(1,1) NOT NULL,

[NameAssemblyType] [nvarchar](50) NOT NULL,

CONSTRAINT [PK\_Type] PRIMARY KEY CLUSTERED

(

[Id\_AssemblyType] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF,

IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON,

ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY =

OFF) ON [PRIMARY]

) ON [PRIMARY]

GO

/\*\*\*\*\* Object: Table [dbo].[Comment] Script Date: 09.04.2024 6:28:22

\*\*\*\*\*/

SET ANSI\_NULLS ON

		Козин Д			<b>КП 09.02.07.21И1.09.000 ПЗ</b>	Лист
						28
Изм.	Лист	№. докум.	Подпись	Дата		

```

GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Comment](
    [Id_Comment] [int] IDENTITY(1,1) NOT NULL,
    [DescriptionComment] [nvarchar](255) NOT NULL,
    [Id_Employee] [int] NOT NULL,
    [Id_Task] [int] NOT NULL,
    CONSTRAINT [PK_Comment] PRIMARY KEY CLUSTERED
(
    [Id_Comment] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,
ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY =
OFF) ON [PRIMARY]
) ON [PRIMARY]
GO
/***** Object: Table [dbo].[Dependency]    Script Date: 09.04.2024 6:28:22
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Dependency](
    [IdDependency] [int] IDENTITY(1,1) NOT NULL,
    [IdDependentTask] [int] NOT NULL,
    [IdCodependentTask] [int] NOT NULL,
    CONSTRAINT [PK_Dependency] PRIMARY KEY CLUSTERED
(

```

		Козин Д			<b>КП 09.02.07.21И1.09.000 ПЗ</b>	Лист
						29
Изм.	Лист	№. докум.	Подпись	Дата		

```

[IdDependency] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,
ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY =
OFF) ON [PRIMARY]
) ON [PRIMARY]
GO

/***** Object: Table [dbo].[Employee]    Script Date: 09.04.2024 6:28:22
*****/

SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Employee](
    [Id_employee] [int] IDENTITY(1,1) NOT NULL,
    [Surname] [nvarchar](50) NOT NULL,
    [Name] [nvarchar](30) NOT NULL,
    [Patronymic] [nvarchar](50) NOT NULL,
    [IdPost] [int] NOT NULL,
    [Image] [nvarchar](100) NULL,
    [Login] [nvarchar](20) NOT NULL,
    [Password] [nvarchar](15) NOT NULL,
    CONSTRAINT [PK_Employee] PRIMARY KEY CLUSTERED
(
    [Id_employee] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,
ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY =
OFF) ON [PRIMARY]

```

		Козин Д			<b>КП 09.02.07.21И1.09.000 ПЗ</b>	Лист
						30
Изм.	Лист	№. докум.	Подпись	Дата		

```

) ON [PRIMARY]
GO

/***** Object: Table [dbo].[Phase]  Script Date: 09.04.2024 6:28:22 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Phase](
    [Id_Phase] [int] IDENTITY(1,1) NOT NULL,
    [NamePhase] [nvarchar](50) NOT NULL,
    CONSTRAINT [PK_Phase] PRIMARY KEY CLUSTERED
(
    [Id_Phase] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,
ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY =
OFF) ON [PRIMARY]
) ON [PRIMARY]
GO

/***** Object: Table [dbo].[Post]  Script Date: 09.04.2024 6:28:22 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Post](
    [IdPost] [int] IDENTITY(1,1) NOT NULL,
    [Title] [nvarchar](35) NOT NULL,
    CONSTRAINT [PK_Post] PRIMARY KEY CLUSTERED
(

```

		Козин Д			<b>КП 09.02.07.21И1.09.000 ПЗ</b>	Лист
						31
Изм.	Лист	№. докум.	Подпись	Дата		

```

[IdPost] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,
ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY =
OFF) ON [PRIMARY]
) ON [PRIMARY]
GO

/***** Object: Table [dbo].[Priority]    Script Date: 09.04.2024 6:28:22
*****/

SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Priority](
    [IdPriority] [int] IDENTITY(1,1) NOT NULL,
    [NamePriority] [nvarchar](50) NOT NULL,
    CONSTRAINT [PK_Priority] PRIMARY KEY CLUSTERED
(
    [IdPriority] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,
ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY =
OFF) ON [PRIMARY]
) ON [PRIMARY]
GO

/***** Object: Table [dbo].[Release]    Script Date: 09.04.2024 6:28:22
*****/

SET ANSI_NULLS ON
GO

```

		Козин Д			<b>КП 09.02.07.21И1.09.000 ПЗ</b>	Лист
						32
Изм.	Лист	№. докум.	Подпись	Дата		



```

SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Release](
    [Id_Release] [int] IDENTITY(1,1) NOT NULL,
    [NameRelease] [nvarchar](50) NOT NULL,
    [StartDate] [date] NULL,
    [FinishDate] [date] NULL,
    CONSTRAINT [PK_Release] PRIMARY KEY CLUSTERED
(
    [Id_Release] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,
ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY =
OFF) ON [PRIMARY]
) ON [PRIMARY]
GO
/***** Object: Table [dbo].[Task]  Script Date: 09.04.2024 6:28:22 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Task](
    [Id_Task] [int] IDENTITY(1,1) NOT NULL,
    [Designation] [nvarchar](255) NOT NULL,
    [DescriptionTask] [nvarchar](1000) NOT NULL,
    [Id_Developer] [int] NOT NULL,
    [Id_Tester] [int] NOT NULL,
    [Id_Creator] [int] NOT NULL,
    [IdPhase] [int] NOT NULL,

```

		Козин Д			<b>КП 09.02.07.21И1.09.000 ПЗ</b>	Лист
						33
Изм.	Лист	№. докум.	Подпись	Дата		

```

[IdAssemblyType] [int] NOT NULL,
[IdRelease] [int] NOT NULL,
[IdPriority] [int] NOT NULL,
[IdTypeOS] [int] NOT NULL,
[Access] [bit] NOT NULL,
CONSTRAINT [PK_Task] PRIMARY KEY CLUSTERED
(
    [Id_Task] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,
ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY =
OFF) ON [PRIMARY]
) ON [PRIMARY]
GO

/***** Object:  Table [dbo].[TypeOS]      Script Date: 09.04.2024 6:28:22
*****/

SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[TypeOS](
    [Id_TypeOS] [int] IDENTITY(1,1) NOT NULL,
    [NameTypeOS] [nvarchar](50) NOT NULL,
    CONSTRAINT [PK_TypeOS] PRIMARY KEY CLUSTERED
(
    [Id_TypeOS] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,

```

		Козин Д			<b>КП 09.02.07.21И1.09.000 ПЗ</b>	Лист
Изм.	Лист	№. докум.	Подпись	Дата		34

```

ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY =
OFF) ON [PRIMARY]
) ON [PRIMARY]
GO
ALTER TABLE [dbo].[Comment] WITH CHECK ADD CONSTRAINT
[FK_Comment_Employee] FOREIGN KEY([Id_Employee])
REFERENCES [dbo].[Employee] ([Id_employee])
GO
ALTER TABLE [dbo].[Comment] CHECK CONSTRAINT
[FK_Comment_Employee]
GO
ALTER TABLE [dbo].[Comment] WITH CHECK ADD CONSTRAINT
[FK_Comment_Task] FOREIGN KEY([Id_Task])
REFERENCES [dbo].[Task] ([Id_Task])
GO
ALTER TABLE [dbo].[Comment] CHECK CONSTRAINT
[FK_Comment_Task]
GO
ALTER TABLE [dbo].[Dependency] WITH CHECK ADD CONSTRAINT
[FK_Dependency_Task] FOREIGN KEY([IdCodependentTask])
REFERENCES [dbo].[Task] ([Id_Task])
GO
ALTER TABLE [dbo].[Dependency] CHECK CONSTRAINT
[FK_Dependency_Task]
GO
ALTER TABLE [dbo].[Dependency] WITH CHECK ADD CONSTRAINT
[FK_Dependency_Task1] FOREIGN KEY([IdDependentTask])
REFERENCES [dbo].[Task] ([Id_Task])
GO

```

```

ALTER TABLE [dbo].[Dependency] CHECK CONSTRAINT
[FK_Dependency_Task1]
GO
ALTER TABLE [dbo].[Employee] WITH CHECK ADD CONSTRAINT
[FK_Employee_Post] FOREIGN KEY([IdPost])
REFERENCES [dbo].[Post] ([IdPost])
GO
ALTER TABLE [dbo].[Employee] CHECK CONSTRAINT
[FK_Employee_Post]
GO
ALTER TABLE [dbo].[Task] WITH CHECK ADD CONSTRAINT
[FK_Task_Employee] FOREIGN KEY([Id_Developer])
REFERENCES [dbo].[Employee] ([Id_employee])
GO
ALTER TABLE [dbo].[Task] CHECK CONSTRAINT [FK_Task_Employee]
GO
ALTER TABLE [dbo].[Task] WITH CHECK ADD CONSTRAINT
[FK_Task_Employee1] FOREIGN KEY([Id_Tester])
REFERENCES [dbo].[Employee] ([Id_employee])
GO
ALTER TABLE [dbo].[Task] CHECK CONSTRAINT [FK_Task_Employee1]
GO
ALTER TABLE [dbo].[Task] WITH CHECK ADD CONSTRAINT
[FK_Task_Employee2] FOREIGN KEY([Id_Creator])
REFERENCES [dbo].[Employee] ([Id_employee])
GO
ALTER TABLE [dbo].[Task] CHECK CONSTRAINT [FK_Task_Employee2]
GO

```

```

ALTER TABLE [dbo].[Task] WITH CHECK ADD CONSTRAINT
[FK_Task_Phase] FOREIGN KEY([IdPhase])
REFERENCES [dbo].[Phase] ([Id_Phase])
GO
ALTER TABLE [dbo].[Task] CHECK CONSTRAINT [FK_Task_Phase]
GO
ALTER TABLE [dbo].[Task] WITH CHECK ADD CONSTRAINT
[FK_Task_Priority] FOREIGN KEY([IdPriority])
REFERENCES [dbo].[Priority] ([IdPriority])
GO
ALTER TABLE [dbo].[Task] CHECK CONSTRAINT [FK_Task_Priority]
GO
ALTER TABLE [dbo].[Task] WITH CHECK ADD CONSTRAINT
[FK_Task_Release] FOREIGN KEY([IdRelease])
REFERENCES [dbo].[Release] ([Id_Release])
GO
ALTER TABLE [dbo].[Task] CHECK CONSTRAINT [FK_Task_Release]
GO
ALTER TABLE [dbo].[Task] WITH CHECK ADD CONSTRAINT
[FK_Task_Type] FOREIGN KEY([IdAssemblyType])
REFERENCES [dbo].[AssemblyType] ([Id_AssemblyType])
GO
ALTER TABLE [dbo].[Task] CHECK CONSTRAINT [FK_Task_Type]
GO
ALTER TABLE [dbo].[Task] WITH CHECK ADD CONSTRAINT
[FK_Task_TypeOS] FOREIGN KEY([IdTypeOS])
REFERENCES [dbo].[TypeOS] ([Id_TypeOS])
GO
ALTER TABLE [dbo].[Task] CHECK CONSTRAINT [FK_Task_TypeOS]

```

```
GO
USE [master]
GO
ALTER DATABASE [Development] SET READ_WRITE
GO
```

## 3.2 Разработка пользовательского интерфейса

### 3.2.1 Разработка стиля приложения.

При разработке данного приложения был выбран определённый набор цветов и использован при создании элементов окна

Примеры:

- Задание цвета окна:

```
<Grid      Background="#024C68"      Height="792"      VerticalAlignment="Top"
HorizontalAlignment="Center" Width="1400">
```

- Задание цвета листа:

```
<Grid      Background="#024C68"      Height="792"      VerticalAlignment="Top"
HorizontalAlignment="Center" Width="1400">
```

- Создание стиля кнопки:

```
<Button      x:Name="CreateTask"      Content="Создать задачу"      FontSize="23"
HorizontalAlignment="Left"      Margin="1052,10,0,739"      Background="#62B1D0"
VerticalAlignment="Center"      Foreground="#024C68"      Width="175"      Height="43"
FontFamily="Segoe UI Semibold" Click="Create_Click"/>
```

При разработке был выбран основной шрифт – «Segoe UI Semibold». А так же учитывалось расположение элементов в окне, размер шрифта, размер объектов – для удобства и практичности пользования приложением.

### 3.2.2 Разработка подсистемы авторизации

```
public partial class MainWindow : Window
{
    int trys = 0;
    public static string fullname;
    public static int idpost;
    public static int idempl;

    DevelopmentEntities db = new DevelopmentEntities();
    Employee NSP = new Employee();

    public MainWindow()
    {
        InitializeComponent();
    }

    private void Enter_Click(object sender, RoutedEventArgs e)
    {
        var users = db.Employee;
        if(users.Select(x=>x.Login).ToList().Contains(Log.Text))
        {
            if(users.Where(x=>x.Login == Log.Text).ToList()[0].Password == Psw.Text)
            {
                fullname = Convert.ToString(users.Where(x => x.Login ==
Log.Text).ToList()[0].FullName);
                idpost = Convert.ToInt32(users.Where(x => x.Login ==
Log.Text).ToList()[0].IdPost);
                idempl = Convert.ToInt32(users.Where(x => x.Login ==
Log.Text).ToList()[0].Id_employee);

                NSP.Name = fullname;
                NSP.IdPost = idpost;
                NSP.Id_employee = idempl;

                User._user.Add(NSP);

                Main main = new Main();
                main.Show();
                this.Close();
            }
            else
            {
                MessageBox.Show("Вы ввели неверный логин или пароль. Пожалуйста
проверьте ещё раз введенные данные");
                trys++;
                if (trys == 3)
                {
                    MessageBox.Show("Вы 3 раза не верно авторизовались, перепроверьте
свои данные и попробуйте снова");
                    this.Close();
                }
            }
        }
    }
}
```

		Козин Д			КП 09.02.07.21И1.09.000 ПЗ	Лист
Изм.	Лист	№. докум.	Подпись	Дата		40



При входе в систему происходит проверка соответствия логина и пароля. При правильном вводе, в созданный класс в Листе User, из базы данных заносятся: полное имя вошедшего пользователя, его Id и Id должности, для дальнейшего разграничения доступа в системе. Так же создан обработчик события для вывода сообщения о неправильности введенных данных и вывод сообщения о том, что пользователь неправильно ввел данные 3 раза, с последующим закрытием программы.

		Козин Д			<b>КП 09.02.07.21И1.09.000 ПЗ</b>	Лист
Изм.	Лист	№. докум.	Подпись	Дата		41

### 3.2.3 Разработка подсистемы приложения

XAML код:

```
Grid Background="#024C68" Margin="0,0,-0.4,0">
    <ListView x:Name="lstw_SelectTask"
        ScrollViewer.HorizontalScrollBarVisibility="Disabled"
        HorizontalContentAlignment="Stretch" Background="#3AA6D0" Margin="27,63,24,27">

        <ListView.ItemTemplate>
            <DataTemplate>
                <Border BorderBrush="Black" BorderThickness="1">
                    <Grid>
                        <Grid.ColumnDefinitions>
                            <ColumnDefinition Width="Auto" />
                            <ColumnDefinition Width="*" />
                            <ColumnDefinition Width="Auto" />
                        </Grid.ColumnDefinitions>

                        <StackPanel x:Name="stc1" Grid.Column="0"
                            Orientation="Vertical" Background="#62B1D0" Margin="5" Width="400">
                            <TextBlock FontSize="16" FontWeight="Bold" Text="{Binding
                                Designation}" Margin="5" TextWrapping="Wrap"/>
                            <TextBlock FontSize="14" Text="{Binding DescriptionTask}"
                                Margin="5" TextWrapping="Wrap"/>

                            <TextBlock FontSize="14" Text="Создатель: "
                                Foreground="Gray" Margin="5,2,5,0"/>
                            <TextBlock x:Name="Creator" FontSize="14" Text="{Binding
                                Id_Creator.FullName}" Margin="5,0,5,5"/>
                            <TextBlock FontSize="14" Text="Разработчик: "
                                Foreground="Gray" Margin="5,2,5,0"/>
                            <TextBlock x:Name="Developer" FontSize="14"
                                Text="{Binding Id_Developer.FullName}" Margin="5,0,5,5"/>
                            <TextBlock FontSize="14" Text="Тестировщик: "
                                Foreground="Gray" Margin="5,2,5,0"/>
                            <TextBlock x:Name="Tester" FontSize="14" Text="{Binding
                                Id_Tester.Name}" Margin="5,0,5,5"/>
                        </StackPanel>
                        <StackPanel Grid.Column="1" Orientation="Vertical"
                            Background="#62B1D0" Margin="5">
                            <TextBlock FontSize="14" Text="Тип ОС: "
                                Foreground="Gray" Margin="5,2,5,0"/>
                            <TextBlock FontSize="14" Text="{Binding
                                TypeOS.NameTypeOS}" Margin="5,0,5,5"/>

                            <TextBlock FontSize="14" Text="Тип сборки: "
                                Foreground="Gray" Margin="5,2,5,0"/>
                            <TextBlock FontSize="14" Text="{Binding
                                AssemblyType.NameAssemblyType}" Margin="5,0,5,5"/>

                            <TextBlock FontSize="14" Text="Версия: "
                                Foreground="Gray" Margin="5,2,5,0"/>
                            <TextBlock FontSize="14" Text="{Binding
                                Release.NameRelease}" Margin="5,0,5,5"/>
                            <TextBlock FontSize="14" Text="Приоритет: "
                                Foreground="Gray" Margin="5,2,5,0"/>
                            <TextBlock FontSize="14" Text="{Binding
                                Priority.NamePriority}" Margin="5,0,5,5"/>
                        </StackPanel>
                    </Grid>
                </Border>
            </DataTemplate>
        </ListView.ItemTemplate>
    </ListView>
</Grid>
```

		Козин Д			КП 09.02.07.21И1.09.000 ПЗ	Лист
						42
Изм.	Лист	№. докум.	Подпись	Дата		

```

        <StackPanel Grid.Column="2" Orientation="Vertical"
Width="200" Background="#62B1D0" Margin="5">
        <TextBlock FontSize="16" FontWeight="Bold"
Text="Комментарии" Margin="0,10,0,5"/>
        <ScrollViewer VerticalScrollBarVisibility="Auto"
Height="150">
            <ItemsControl ItemsSource="{Binding Comments}">
                <ItemsControl.ItemTemplate>
                    <DataTemplate>
                        <StackPanel>
                            <TextBlock FontSize="14"
Text="{Binding Employee.Name}" FontWeight="Bold" Margin="5,0,0,0"/>
                            <TextBlock FontSize="14"
Text="{Binding Comment.DescriptionComment}" TextWrapping="Wrap" Margin="5,0,0,0"/>
                        </StackPanel>
                    </DataTemplate>
                </ItemsControl.ItemTemplate>
            </ItemsControl>
            <ScrollViewer>
                <TextBox x:Name="CommentTextBox" TextWrapping="Wrap"
Height="50" Margin="0,10,0,5" VerticalScrollBarVisibility="Auto"/>
                <Button Content="Добавить комментарий"
Click="AddCommentButton_Click"/>
            </ScrollViewer>
        </StackPanel>
    </Grid>
</Border>
</DataTemplate>
</ListView.ItemTemplate>
</ListView>
<Button x:Name="Enter" Content="Вернуться" FontSize="23"
HorizontalAlignment="Left" Margin="27,11,0,366" Background="#62B1D0"
VerticalAlignment="Center" Foreground="#024C68" Width="142" Height="43" FontFamily="Segoe
UI Semibold" Click="Enter_Click"/>
</Grid>

```

## Main код:

```

public partial class Window1 : Window
{
    DevelopmentEntities db = new DevelopmentEntities();
    Employee NSP = new Employee();

    public Window1()
    {
        InitializeComponent();
        lstw_SelectTask.ItemsSource = SelectTask._select_task;
    }

    private void Enter_Click(object sender, RoutedEventArgs e)
    {
        Main main = new Main();
        main.Show();
        this.Close();
    }

    private void AddCommentButton_Click(object sender, RoutedEventArgs e)
    {
        try
        {

```

		Козин Д			КП 09.02.07.21И1.09.000 ПЗ	Лист
Изм.	Лист	№. докум.	Подпись	Дата		43

```

var selected_task = SelectTask._select_task;
Comment comment = new Comment();

int id_t = 0;
foreach (var h in SelectTask._select_task)
{
    id_t = h.Id_Task;
}

if (selected_task != null)
{
    TextBox commentTextBox = lstw_SelectTask.FindName("CommentTextBox")
as TextBox;

    if (commentTextBox != null)
    {

        string newCommentText = commentTextBox.Text;

        if (!string.IsNullOrEmpty(newCommentText))
        {

            comment.DescriptionComment = commentTextBox.Text;
            comment.Id_Employee = MainWindow.idempl;
            comment.Id_Task = id_t;
            db.Comment.Add(comment);
            db.SaveChanges();
            MessageBox.Show("Комментарий добавлен");
        }
        else
        {
            MessageBox.Show("Поле не заполнено");
        }
    }
}
}
catch
{
    MessageBox.Show("Ошибка!");
}
}
}

```

В XAML коде прописаны элементы, выводимые в окне и в самом листе, а также что в этих элементах будет выводиться. Так же в листе создано поле и кнопка для возможности добавления комментария к задаче

В Main коде описан вызов выбранной задачи, для вывода данных о ней в окне. Прописан код на кнопку для возвращения в главное окно. А так же прописан код для добавления комментария к задаче путем. Из поля считывается введенный текст, присваивается Id пользователя за которого был произведен вход в систему и присваивается Id открытой задачи. После чего эти данные заносятся в базу данных. Так же создан обработчик для информирования

		Козин Д			<b>КП 09.02.07.21И1.09.000 ПЗ</b>	Лист
Изм.	Лист	№. докум.	Подпись	Дата		44

пользователя о том, что поле не заполнено и о том, что произошла ошибка при добавлении.

		Козин Д			<b>КП 09.02.07.21И1.09.000 ПЗ</b>	Лист
Изм.	Лист	№. докум.	Подпись	Дата		45

### 3.2.4 Разработка подсистемы администратора БД.

```
public partial class Window2 : Window
{
    DevelopmentEntities db = new DevelopmentEntities();
    Task T = new Task();
    public static string developer;
    public static string tester;
    public static char version;
    public static string prioritet;
    public static string os;
    public static string assemble;
    public Window2()
    {
        InitializeComponent();

        var typeos = db.TypeOS;
        foreach (var type in typeos)
        {
            OS.Items.Add(type.NameTypeOS);
        }

        var assemtype = db.AssemblyType;
        foreach (var assembl in assemtype)
        {
            Asmb.Items.Add(assembl.NameAssemblyType);
        }

        var release = db.Release;
        foreach (var rel in release)
        {
            Rel.Items.Add(rel.NameRelease);
        }

        var prioritet = db.Priority;
        foreach (var pri in prioritet)
        {
            Prior.Items.Add(pri.NamePriority);
        }

        var devop = db.Employee.Where(x => x.IdPost == 2);
        foreach (var dv in devop)
        {
            Dev.Items.Add(dv.FullName);
        }

        var testor = db.Employee.Where(x => x.IdPost == 3);
        foreach (var tst in testor)
        {
            Test.Items.Add(tst.FullName);
        }
    }

    private void Enter_Click(object sender, RoutedEventArgs e)
    {
        try
        {
            Task task = new Task();
        }
    }
}
```

		Козин Д			КП 09.02.07.21И1.09.000 ПЗ	Лист
Изм.	Лист	№. докум.	Подпись	Дата		46

```

        if (!string.IsNullOrEmpty(Discript.Text) &&
!string.IsNullOrEmpty(Discription.Text) && Dev != null && Test != null && Asmb != null &&
OS != null && Rel != null && Prior != null)
        {
            task.Id_Developer = db.Employee.FirstOrDefault(x =>
Dev.Text.Contains(x.Surname)).Id_employee;
            task.Id_Tester = db.Employee.FirstOrDefault(x =>
Test.Text.Contains(x.Surname)).Id_employee;
            task.IdRelease = db.Release.FirstOrDefault(x => x.NameRelease ==
Rel.Text).Id_Release;
            task.IdAssemblyType = db.AssemblyType.FirstOrDefault(x =>
x.NameAssemblyType == Asmb.Text).Id_AssemblyType;
            task.IdTypeOS = db.TypeOS.FirstOrDefault(x => x.NameTypeOS ==
OS.Text).Id_TypeOS;
            task.IdPriority = db.Priority.FirstOrDefault(x => x.NamePriority ==
Prior.Text).IdPriority;
            task.Designation = Discript.Text;
            task.DescriptionTask = Discription.Text;
            task.Access = false;
            task.IdPhase = 1;
            task.Id_Creator = MainWindow.idempl;
            db.Task.Add(task);
            db.SaveChanges();
            MessageBox.Show("Задача успешно создана!");
        }
        else
        {
            MessageBox.Show("Есть пустые поля!");
        }
    }
    catch
    {
        MessageBox.Show("Ошибка!");
    }
}

private void Enter1_Click(object sender, RoutedEventArgs e)
{
    Main main = new Main();
    main.Show();
    this.Close();
}
}
}

```

Администратор (Руководитель) имеет возможность создавать задачу. В окне созданы элементы для выбора и заполнения данных по задаче, наименование, разработчик, тестировщик и т.д.. При нажатии на кнопку создания со всех элементов собираются значения, группируются и заносятся в базу данных. Так как в приложении руководитель выполняет функцию администратора, то при создании задачи в Id\_Creator вносится Id вошедшего пользователя, так как к данному окну доступ имеет только руководитель. Для создания задачи так же прописан обработчик информирующий о наличии пустых полей и об ошибке создания. Прописан код на кнопку возвращения в главное окно.

4 Инструкции

4.1 Инструкции пользователю

Для доступа к ПО нужно быть зарегистрированным в системе, после этого получить Логин и Пароль для входа. После входа в систему в зависимости от должности пользователь может приступать к работе.



Рисунок 6 Kanban-доска

После входа в систему пользователю будут доступны те или иные возможности, тестировщики и разработчики могут взаимодействовать только со своими задачами и только в определенных полях, в зависимости от должности. Аналитики и Дизайнеры могут только оставлять комментарии к задачам и просматривать все выполняемые задачи в системе. Руководитель может заносить новые задачи, взаимодействовать и перемещать все задачи.



Рисунок 7 Форма создания задачи

Рисунок 8 Вывод полной информации о выбранной задаче

Вывод отдельной задачи в окно предоставляет пользователю всю информацию о задаче, а также дает возможность оставить комментарий в случае необходимости

## 5 Безопасность и надежность функционирования информационной системы

Для реализации надежности и безопасности от несанкционированного доступа системы, вход реализуется с помощью ввода Логина и Пароля, так же при входе в систему формируются токены, по которым будет распределяться функционал системы между пользователями.

Рисунок 26 Вход в систему

Код для проверки логина и пароля, введенного пользователем:

```
private void Enter_Click(object sender, RoutedEventArgs e)
{
    var users = db.Employee;
    if (users.Select(x => x.Login).ToList().Contains(Log.Text))
    {
        if (users.Where(x => x.Login == Log.Text).ToList()[0].Password ==
Psw.Text)
        {
            fullname = Convert.ToString(users.Where(x => x.Login ==
Log.Text).ToList()[0].FullName);
            idpost = Convert.ToInt32(users.Where(x => x.Login ==
Log.Text).ToList()[0].IdPost);
            idempl = Convert.ToInt32(users.Where(x => x.Login ==
Log.Text).ToList()[0].Id_employee);

            NSP.Name = fullname;
            NSP.IdPost = idpost;
            NSP.Id_employee = idempl;

            User._user.Add(NSP);
        }
    }
}
```

		Козин Д			<b>КП 09.02.07.21И1.09.000 ПЗ</b>	Лист
Изм.	Лист	№. докум.	Подпись	Дата		50

```

        Main main = new Main();
        main.Show();
        this.Close();
    }
}
else
{
    MessageBox.Show("Вы ввели неверный логин или пароль. Пожалуйста проверьте
ещё раз введенные данные");
    trys++;
    if (trys == 3)
    {
        MessageBox.Show("Вы 3 раза не верно авторизовались, перепроверьте
свои данные и попробуйте снова");
        this.Close();
    }
}
}
}

```

		Козин Д			<b>КП 09.02.07.21И1.09.000 ПЗ</b>	Лист
						51
Изм.	Лист	№. докум.	Подпись	Дата		

## 6. Охрана труда

### 6.1 Общие требования охраны труда

6.1.1 К работе на персональном компьютере допускаются лица не моложе 18 лет, прошедшие предварительный (при поступлении на работу) медицинский осмотр, выполняющие требования правил по эксплуатации конкретного персонального компьютера (далее ПК), ознакомленные с настоящей инструкцией по пожарной безопасности.

6.1.2 Работник, допустивший нарушение инструкции по охране труда, может быть привлечен к дисциплинарной ответственности. Если нарушение правил охраны труда связано с причинением имущественного ущерба предприятию, работник несет и материальную ответственность в установленном законом порядке.

6.1.3 В помещениях, где проводятся работы на ПК, необходимо создать оптимальные условия зрительной работы. Освещенность рабочего места при смешанном освещении (в горизонтальной плоскости в зоне размещения клавиатуры и рабочих документов) должна быть в пределах от 300 до 500 Лк. Основной поток естественного света должен быть слева, солнечные лучи и блики не должны попадать в поле зрения работающего и на экраны видеомониторов.

6.1.4 Монитор ПК должен находиться на расстоянии 50-70 см от глаз оператора и иметь антибликовое покрытие. Покрытие должно также обеспечивать снятие электростатического заряда с поверхности экрана, исключать искрение и накопление пыли.

6.1.5 Нельзя загораживать заднюю стенку системного блока или ставить ПК вплотную к стене, это приводит к нарушению охлаждения системного блока и его перегреву.

6.1.6 Режим работы и отдыха должен зависеть от характера выполняемой работы. При вводе данных, редактировании программ,

		Козин Д			<b>КП 09.02.07.21И1.09.000 ПЗ</b>	Лист
						52
Изм.	Лист	№. докум.	Подпись	Дата		

считывании информации с экрана непрерывная продолжительность работы с ПК не должна превышать 4 часа за рабочий день при 8-ми часовом рабочем дне. Через каждый час работы необходимо делать перерывы на отдых по 5-10 минут или по 15-20 минут каждые два часа работы.

#### 6.2 Требования охраны труда перед началом работы

6.2.1 Подготовить свое рабочее место к работе, убрать посторонние предметы.

6.2.2 Произвести визуальный осмотр ПК, убедиться в исправности розеток, штепсельных вилок, питающих электрошнуров.

6.2.3 Включить ПК в сеть 220В, при этом штепсельную вилку держать за корпус.

#### 6.3 Требования охраны труда во время работы

6.3.1 Во время работы быть внимательным, не отвлекаться посторонними делами и разговорами.

6.3.2 Рабочее место должно быть оборудовано так, чтобы исключать неудобные позы и длительные статические напряжения тела.

6.3.3 При работе на ПК должна быть исключена возможность одновременного прикосновения к оборудованию и к частям помещения или оборудования, имеющим соединение с землей (радиаторы батарей, металлоконструкции).

6.3.4 Во время работы нельзя класть на монитор бумаги, книги и другие предметы, которые могут закрыть его вентиляционные отверстия.

6.3.5 Запрещается оставлять без присмотра включенное оборудование; вскрывать устройства ПК.

#### 6.4 Требования охраны труда во время аварийных ситуациях

6.4.1 При возникновении неисправности в ПК необходимо отключить ПК от сети. ЗАПРЕЩАЕТСЯ пытаться самостоятельно устранить причину неисправности, об этом необходимо сообщить в соответствующие службы технического обслуживания.

		Козин Д			<b>КП 09.02.07.21И1.09.000 ПЗ</b>	Лист
Изм.	Лист	№. докум.	Подпись	Дата		53

6.4.2 В случае загорания электропровода или ПК немедленно отключить его от сети, сообщить об этом в пожарную часть по телефону 01 (112 с мобильного телефона) и приступить к тушению пожара углекислотным или порошковым огнетушителем.

Запрещается применять пенные огнетушители для тушения электропроводок и оборудования под напряжением, так как пена — хороший проводник электрического тока.

6.4.3 В случае поражения работника электрическим током оказать первую помощь пострадавшему, обратиться в медпункт или вызвать врача.

#### 6.5 Требования охраны труда после окончания работы

6.5.1 Отключить ПК от сети, штепсельную вилку при этом держать за корпус. Запрещается отключать ПК за электропровод. При отключении ПК со съемным шнуром питания сначала необходимо отключить вилку от розетки, а затем отключить питающий шнур от ПК.

6.5.2 Привести в порядок рабочее место.

6.5.3 Чистку ПК от пыли необходимо производить только после отключения ПК от сети.

		Козин Д			<b>КП 09.02.07.21И1.09.000 ПЗ</b>	Лист
Изм.	Лист	№. докум.	Подпись	Дата		54

## Заключение

В результате выполнения курсового проекта была спроектирована и разработана виртуальная Kanban-доска. Была описана часть необходимых функций, которыми должна обладать подсистема, из них были реализованы наиболее актуальные и важные функции:

- Сбор, организация и загрузка данных
- Хранение данных
- Визуализация процесса работы.

Созданием Kanban-доски была достигнута основная цель курсового проекта, повышены навыки использования Visual Studio, XAML (Extensible Application Markup Language) и Entity Framework.

### Достоинства:

- Вся информация о существующих проектах представлена в упрощенном и лаконичном виде;
- Система имеет многопользовательский доступ;
- Не зависит от технических характеристик;
- Не требует установки, не занимает память компьютера.

### Перспективы развития:

- Расширение функционала и возможностей системы;
- Расширение групповой политики.

## Информационные источники

- 1) Должностные инструкции. Электронный ресурс. URL: <https://mirconsalt.ru/dolzhnostnye-instruktsii/> Дата обращения 23.02.2022.
- 2) Инструкция по охране труда. [Электронный ресурс], - Режим доступа: <https://stroii-baza-ru.turbopages.org/> Дата обращения 25.02.2022.
- 3) Подбор цвета для дизайна [Электронный ресурс], - Режим доступа: <https://freelance.codeby.net/> Дата обращения 14.02.2022.
- 4) Проектирование информационных систем. [Электронный ресурс], - Режим доступа: <https://www.sites.google.com/> Дата обращения 24.01.2022.
- 5) ГОСТ 24.211-82 «Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Требования к содержанию документа «Описание алгоритма»».
- 6) ГОСТ Р 50922-2006. «Защита информации. Основные термины и определения».
- 7) ГОСТ 24.301-80 «Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Общие требования к выполнению текстовых документов».
- 8) Проектирование информационных систем. [Электронный ресурс], - Режим доступа: <https://www.sites.google.com/>
- 9) Стратегия формирования единой системы информатизации на мебельном предприятии. [Электронный ресурс], Режим доступа: <https://www.sites.google.com/>
- 10) Понятие информационных технологий [Электронный ресурс], - Режим доступа: <https://www.sites.google.com/>
- 11) Методология Канбан [Электронный ресурс], - Режим доступа: <https://www.atlassian.com/>
- 12) Зачем нужно шифрование [Электронный ресурс], - Режим доступа: <https://xserver.a-real.ru/>

		Козин Д			<b>КП 09.02.07.21И1.09.000 ПЗ</b>	Лист
Изм.	Лист	№. докум.	Подпись	Дата		56



13) Проектирование информационных систем. [Электронный ресурс], -  
Режим доступа: <https://www.sites.google.com/>

		Козин Д			<b>КП 09.02.07.21И1.09.000 ПЗ</b>	Лист
						57
Изм.	Лист	№. докум.	Подпись	Дата		

		Козин Д			<b>КП 09.02.07.21И1.09.000 ПЗ</b>	Лист
Изм.	Лист	№. докум.	Подпись	Дата		58