**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ КОМИ**

**Государственное профессиональное образовательное учреждение**

**«Воркутинский арктический горно-политехнический колледж»**

**ДИПЛОМНЫЙ ПРОЕКТ**

Разработка и тестирование информационной системы

**Тема** «Строительная компания»

Выполнил студент гр. ИСП-21 /\_\_\_\_\_\_\_\_\_\_/Поспелов Дмитрий Сергеевич/

(подпись) (Ф.И.О.)

**ОЦЕНКА: \_\_\_\_\_\_\_\_\_\_\_\_\_**

Дата: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**ПРОВЕРИЛ**

Руководитель\_\_\_\_\_\_\_\_\_\_\_\_\_/Егоров Данил Павлович /

(подпись) (Ф.И.О.)

Воркута

2025

СОДЕРЖАНИЕ

**ВВЕДЕНИЕ** с 3 стр.

**ГЛАВА 1. ПРЕДПРОЕКТНАЯ ЧАСТЬ ТРЕБОВАНИЙ К ИНФОРМАЦИОННОЙ СИСТЕМЕ**

**1.1 Анализ предметной области**

**1.2 Разработка технического задания**

**1.3 Моделирование бизнес-процессов (IDEF0 is / to be)**

**ГЛАВА 2. ВЫБОР ИНСТРУМЕНТАРИЯ**

**2.1 Платформа .NET**

**2.2 Язык программирования С#**

**2.3 Windows Presentation Foundation (WPF)**

**2.4 СУБД SQL server**

**2.5 Microsoft SQL Server Management Studio**

**2.6 Entity Framework**

**ГЛАВА 3. РАЗРАБОТКА БАЗЫ ДАННЫХ**

**3.1 Разработка базы данных**

**3.2 Программирование информационной системы**

**3.2.1 Разработка модуля «Авторизация»**

**3.2.2 Разработка модуля «Окна работы вашей системы»**

**3.2.3 Разработка модуля «Окна работы вашей системы»**

**3.2.4 Разработка модуля «Окна работы вашей системы»**

**3.2.5 Разработка модуля «Отчет»**

**3.3 Тестирование информационной системы**

**ЗАКЛЮЧЕНИЕ**

**СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ**

# ВВЕДЕНИЕ

Строительная отрасль занимает ключевое место в экономике любой страны, поскольку обеспечивает создание инфраструктуры, необходимой для функционирования общества. В условиях современного рынка строительные компании сталкиваются с множеством вызовов: от изменения запросов потребителей до колебаний цен на строительные материалы и необходимость внедрения новых технологий. Однако при этом существует значительный потенциал для роста и оптимизации процессов внутри компаний.

Одним из ключевых аспектов в управлении строительным бизнесом является автоматизация процессов, включая проектирование, планирование, учет затрат и взаимодействие с клиентами. Информационная система может стать надежным инструментом, который объединит все эти функции, обеспечивая доступ к актуальной информации и повышая общую производительность компании.

Гипотеза – исследования заключается в том, что внедрение современных технологий управления проектами и автоматизация бизнес-процессов в строительной компании существенно повышает её конкурентоспособность и способствует снижению сроков реализации проектов при одновременном увеличении качества выполняемых работ. Разработка и тестирование информационной системы для строительной компании существенно улучшат ее внутренние процессы, повысят эффективность управления проектами, а также улучшат взаимодействие с клиентами.

Последние годы показали, что строительный сектор требует гибкости и способности быстро реагировать на изменения. Кроме того, с учетом возрастающего спроса на качественные и своевременные услуги, компании, которые инвестируют в современные информационные технологии, имеют значительно больше шансов на успешное развитие и устойчивые позиции на рынке.

**Предмет исследования** – сфера разработки и тестирования информационной системы для автоматизации бизнес-процессов строительной компании.

**Объект исследования** – информационная система «Строительная компания».

Цель проекта: разработать информационную систему для строительной компании.

Задачи проекта:

* проанализировать предметную область;
* выбор инструментария для разработки информационной системы;
* разработать базу данных для компании;
* программирование информационной системы.

**ГЛАВА 1. ПРЕДПРОЕКТНАЯ ЧАСТЬ ТРЕБОВАНИЙ К ИНФОРМАЦИОННОЙ СИСТЕМЕ**

**1.1 Анализ предметной области**

Строительная отрасль представляет собой сложный и многоуровневый процесс, требующий слаженной работы множества подразделений и специалистов. Успех строительного бизнеса во многом зависит от эффективного управления проектами, рационального распределения ресурсов, качественного взаимодействия с клиентами и партнерами, а также способности адаптироваться к изменениям на рынке.

Основные процессы в строительной компании. В рамках строительной компании можно выделить следующие основные процессы:

Планирование и проектирование: на этом этапе происходит разработка концепции проекта, выбор технологий и методов строительства, а также составление смет и графиков выполнения работ.

Управление строительством: включает в себя организацию работ на объектах, распределение задач между сотрудниками и контроль за выполнением сроков и стандартов качества.

Закупка и логистика: обеспечение необходимыми материалами и оборудованием, управление запасами и поставками.

Учет и отчетность: ведение документации по всем производственным и финансовым аспектам, формирование отчетов для аналитики и контроля.

Компания занимается строительством различных типов объектов:

- жилье;

- дороги;

- больницы и прочее.

С компаниями-заказчиками заключаются договора.

Каждый тип объекта имеет свои уникальные характеристики, которые могут быть только у него, либо еще у нескольких схожих типов. Например, жилье имеет следующие характеристики:

- этажность;

- тип строительного материала;

- количество квартир.

Строительная компания состоит из строительных управлений и каждое такое управление работает на одном или нескольких объектах. Во главе таких управлений находятся начальники участков, которые управляют прорабами, мастерами и техниками.

Вообще всех работников строительных управлений можно подразделить на две категории:

- инженерно-технический персонал (инженеры, технологи и прочие);

- рабочие (каменщики, шоферы, слесари и прочие).

Каждая такая категория имеет уникальные атрибуты. Бригадиры руководят рабочими, которые могут объединяться в бригады. Бригадиры – это те же самые рабочие, которые выбираются путем голосования. Из числа инженерно-технического персонала назначаются мастера, прорабы и начальники участков и управлений.

Каждый участок может содержать один или несколько объектов, а каждый объект может объединять в себе несколько бригад. Когда работа у бригады подходит к концу, они «переезжают» на другой объект, который в свою очередь может находиться на том же самом участке, либо на каком-либо другом.

Для различных объектов необходимы различные типы выполняемых работ. К примеру, для жилья – возведение фундамента, прокладка и прочее. Каждой бригаде предполагается один тип выполняемых работ. Для того, чтобы объекты возводились в указанные сроки, составляется план выполнения каждого вида работ. Поскольку строительство предполагает затраты в виде денежных средств, или, например, строительных материалов, составляется смета, в которой указано какие предполагается использовать строительные материалы и в каком количестве. По завершению работ составляется отчет, где указан срок выполнения работ и сколько было затрачено строительных материалов.

Пользователи системы являются:

Администратор (следит за информационной системой).

Менеджер по проектам (планирует и контролирует реализацию строительных проектов).

Бригада (исполняет физические работы на стройплощадке).

Прораб (отвечает за ведущиеся работы на объекте).

Бухгалтер (подготавливает отчёты).

**1.2 Разработка технического задания**

**1. Наименование работы**: Разработка и тестирование информационной системы «Строительная компания».

**2. Назначение разработки.**

Создания информационной системы для «Строительная компания» заключается в создании системы для хранения, управления и обработки данных, связанных с данными строительной компании, а именно управления данными о заказах, а также о строительных бригадах и выполненных ими работах.

В данной ИС можно получить следующую информацию:

1) список строительных управлений и/или строительные участки и их руководство;

2) перечень работников инженерно-технического персонала заданного участка или строительного управления, так же должны быть указаны их должности;

3) список объектов, строящиеся заданным строительным управлением и/или участком, со сроками их возведения;

4) список участников бригады, которые работают (работали) на строительстве заданного объекта;

5) просмотреть сроки и смету заданного объекта;

6) отчет по строительству заданного объекта;

7) список строительных работ, исполненных бригадой в течении конкретного периода времени.

В информационной системе должны быть четко определены роли пользователей:

• **Администратор:** имеет полный доступ ко всем функциям, включая изменение, добавление и редактирование данных.

• **Менеджер по проектам:** контролирует за сроками реализации.

• **Бригада:** просмотр заданных работ.

• **Прораб:** назначает ведущие работы.

• **Бухгалтер:** загружает смету и редактирует отчёты компании.

**3. Требования к программе**

**3.1. Требования к функциональным характеристикам**

Информационная система «Строительная компания» предназначена для управления данными о недвижимостях, материалов, а также о строительных бригадах и выполненных работах. Система должна обеспечивать выполнение следующих функций:

• Получение информации**:** Система должна обеспечивать своевременный доступ к данным о недвижимостях и статусе

• Управление данными: Пользователи должны иметь возможность изменять, добавлять и удалять данные.

• Ввод, хранение, поиск и обработка информации по проектам, заказам и выполненным работам.

• Ведение журнала регистрации недвижимости и выполненных работ.

Для учета выполненных работ в системе используются следующие первичные документы:

Выходными данными являются следующие виды отчетов:

• Список строительных бригад: данные о бригадах и их членах с указанием должностей.

• Выполненные работы по каждой бригаде за заданный интервал времени: отчет о работах, выполненных каждой бригадой за определенный период.

• В базе данных необходимо предусмотреть возможность корректировки настроек системы; резервное сохранение данных; возможность изменения пароля входа в систему; быстрый поиск необходимой информации

**Шифрование данных**

Авторизация: при входе в систему пароль должны шифроваться и сравниваться с зашифрованными данными.

Регистрация: пароль при регистрации также должны шифроваться.

**Авторизация**

Разрабатываемая система будет иметь модульную структуру, доступ к модулям будет зависеть от роли пользователя. Доступ возможен только для авторизованных пользователей.

В связи с этим при запуске системы первым окном будет окно входа.

Реализуйте окно авторизации для всех типов пользователей.

Для реализации авторизации создайте базу данных с необходимыми таблицами и заполните их тестовыми данными.

Доступ к системе имеют только сотрудники. У каждого сотрудника есть логин, по которому и разграничиваются права доступа.

Алгоритм авторизации:

• сотрудник вводит логин и пароль;

• при вводе пароля сотрудником и нажатии клавиши Enter

• получает доступ к необходимому функционалу.

При открытии окна активны только поле для ввода логина сотрудника и кнопка “Отмена”.

При вводе логина сотрудника и нажатию Enter происходит проверка логина сотрудника. Если логин сотрудника есть в базе данных, то поле для ввода пароля становится активным и в нем установлен курсор. Если логин сотрудника в базе отсутствует, появляется сообщение об ошибке.

При вводе неправильного пароля система сообщает пользователю о неверном пароле.

У пользователя должна быть возможность очистить все поля ввода нажатием на кнопку “Отмена”.

После успешной авторизации сотруднику должно быть выведено сообщение с названием его роли.

**3.2. Требования к надежности**

Разрабатываемое программное обеспечение должно иметь:

• разграничение пользовательских прав;

• парольную защиту при запуске программы;

• возможность восстановления информационной системы;

• данные в должны иметь шифрование;

• контроль вводимой информации

• блокировку некорректных действий пользователя с рабочей системой.

**3.3. Требования к составу и параметрам технических средств**

Системные требования для работы программного продукта должны быть следующими: тактовая частота процессора -2 500 Гц; объем оперативной памяти 4 Гб; объем свободного дискового пространства 500 Мб; разрешение монитора 1920 х 1080; наличие устройства чтения USB-носители

**3.4. Требования к информационной и программной совместимости**

Программа должна работать в операционных системах Windows 10/11. Все формируемые отчеты должны иметь возможность экспортирования в редактор электронных таблиц MS Office Excel 2016/2024.

**3.5. Требования к транспортированию и хранению**

Программа поставляется на USB носителе информации. Программная документация поставляется в электронном и печатном виде.

**3.6. Специальные требования**

Программное обеспечение должно иметь дружественный интерфейс, рассчитанный на пользователя средней квалификации (с точки зрения компьютерной грамотности).

Ввиду объемности проекта задачи предполагается решать поэтапно. При этом модули программного обеспечения (ПО), созданные в разное время, должны предполагать возможность наращивания системы и быть совместимы друг с другом; поэтому документация на принятое эксплуатационное ПО должна содержать полную информацию, необходимую для работы с ним программистов. Язык программирования определяется выбором исполнителя, при этом он должен обеспечивать возможность интеграции программного обеспечения с пакетом MS Office 2016/2024.

**4. Требования к программной документации**

Разрабатываемая система должна включать справочную информацию о работе системы и подсказки пользователю. В состав сопровождающей документации должны входить: расчетно-пояснительная записка, содержащая описание системы; руководство пользователя; руководство системного программиста.

В ходе разработки программы должны быть подготовлены: текст программы, описание программы, программа и методика испытаний, руководство пользователя, технико-экономическое обоснование.

При использовании ручных методов регистрации информации о подрядчиках и проектов требуется значительное время. Например, для получения сведений о конкретном проекта может потребоваться 8—10 минут. С внедрением автоматизированной системы это время сокращается до 1—2 минут. Это позволяет сотрудникам сосредоточиться на более важных задачах, таких как анализ данных и взаимодействие с ними.

В конце каждого месяца сотрудники компании вручную заполняют данные о выполненных работах и строительных бригад. Этот процесс может занимать от 1 до 2 дней (6—12 часов). Разработка информационной системы позволяет получить необходимые данные за 3—4 минуты. Кроме того, ИС предоставляет возможность получить данные строительной компании за любой период времени, (информацию о бригадах, недвижимостях, подрядчиков, должность бригад и их членов, выполненные работы) что упрощает анализ деятельности компании. Ручной ввод данных всегда сопряжен с риском ошибок, которые могут привести к неправильным расчетам и недоразумениям. База данных минимизирует вероятность таких ошибок

Таким образом, автоматизация процессов в компании не только оптимизирует текущие операции, но и создает предпосылки для дальнейшего роста и развития бизнеса.

|  |  |  |  |
| --- | --- | --- | --- |
| **№** | **Этап** | **Дата начала** | **Дата окончания** |
|  | Выдача технического задания |  |  |
|  | Выбор инструментария |  |  |
|  | Проектирование базы данных |  |  |
|  | Разработка базы данных |  |  |
|  | Разработка ИС |  |  |
|  | Защита дипломного проекта |  |  |

**1.3 Моделирование бизнес-процессов (IDEF0 is / to be)**

**IDEF0** — [методология](https://ru.wikipedia.org/wiki/%D0%9C%D0%B5%D1%82%D0%BE%D0%B4%D0%BE%D0%BB%D0%BE%D0%B3%D0%B8%D1%8F) функционального моделирования ([англ.](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) *function modeling*) и графическая нотация, предназначенная для формализации и описания [бизнес-процессов](https://ru.wikipedia.org/wiki/%D0%91%D0%B8%D0%B7%D0%BD%D0%B5%D1%81-%D0%BF%D1%80%D0%BE%D1%86%D0%B5%D1%81%D1%81). Отличительной особенностью IDEF0 является её акцент на соподчинённость объектов. В IDEF0 рассматриваются логические отношения между работами, а не их временная последовательность ([поток работ](https://ru.wikipedia.org/wiki/%D0%9F%D0%BE%D1%82%D0%BE%D0%BA_%D1%80%D0%B0%D0%B1%D0%BE%D1%82)).

Стандарт IDEF0 представляет организацию как набор модулей, здесь существует правило — наиболее важная функция находится в верхнем левом углу, кроме того, существуют правила сторон:

* стрелка входа всегда приходит в левую кромку активности,
* стрелка управления — в верхнюю кромку,
* стрелка механизма — нижняя кромка,
* стрелка выхода — правая кромка.

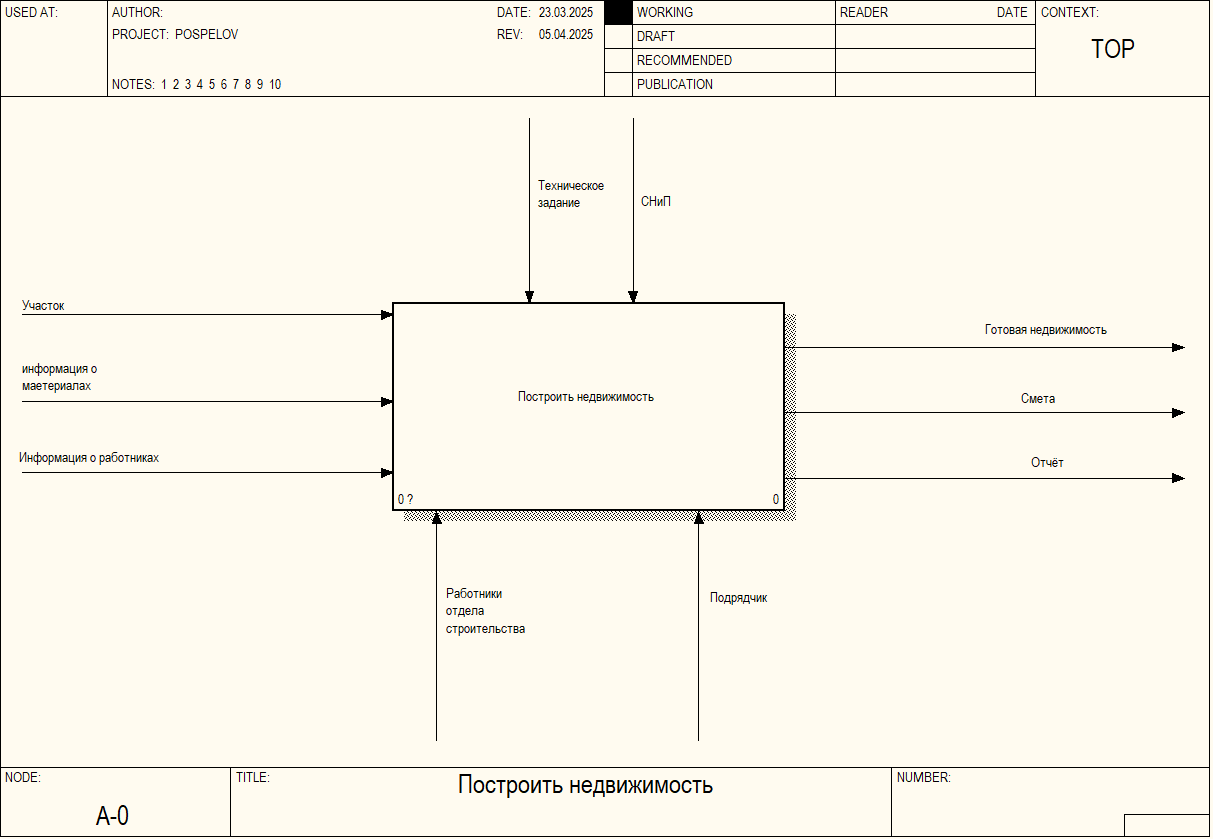
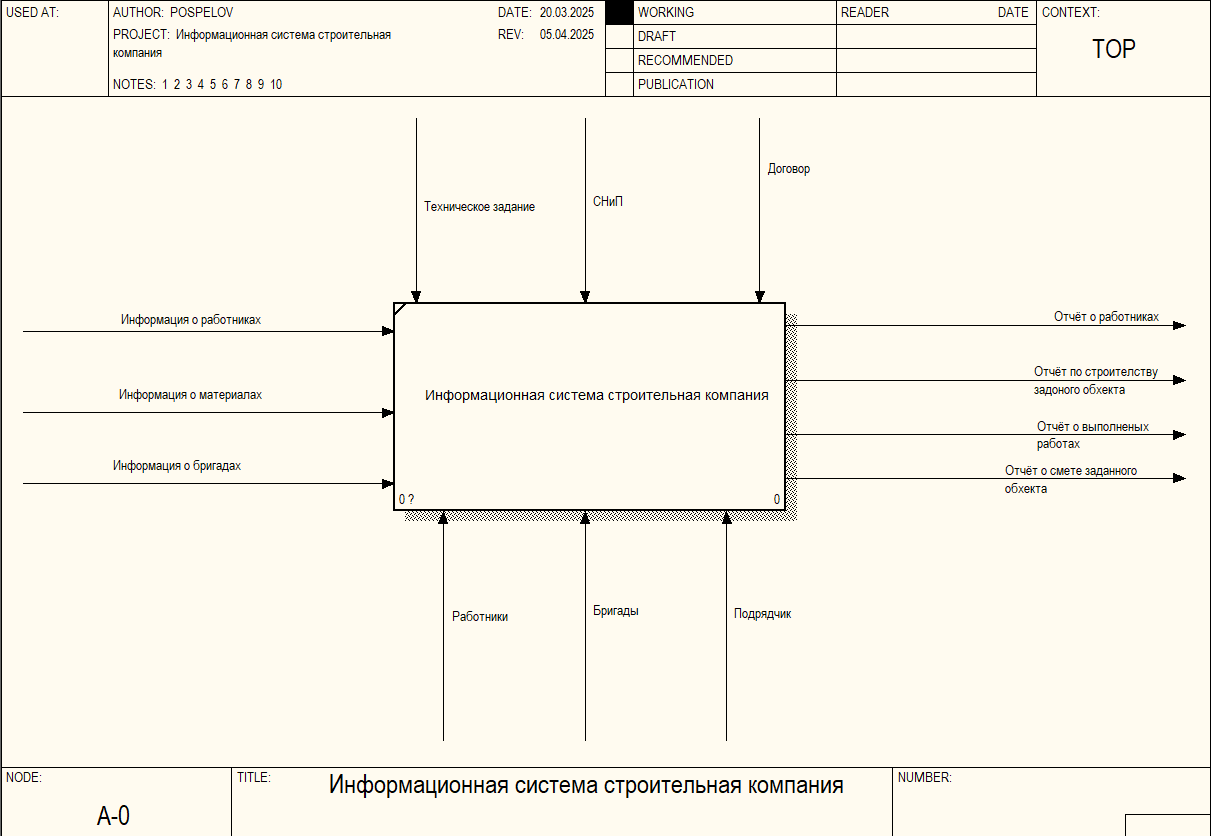
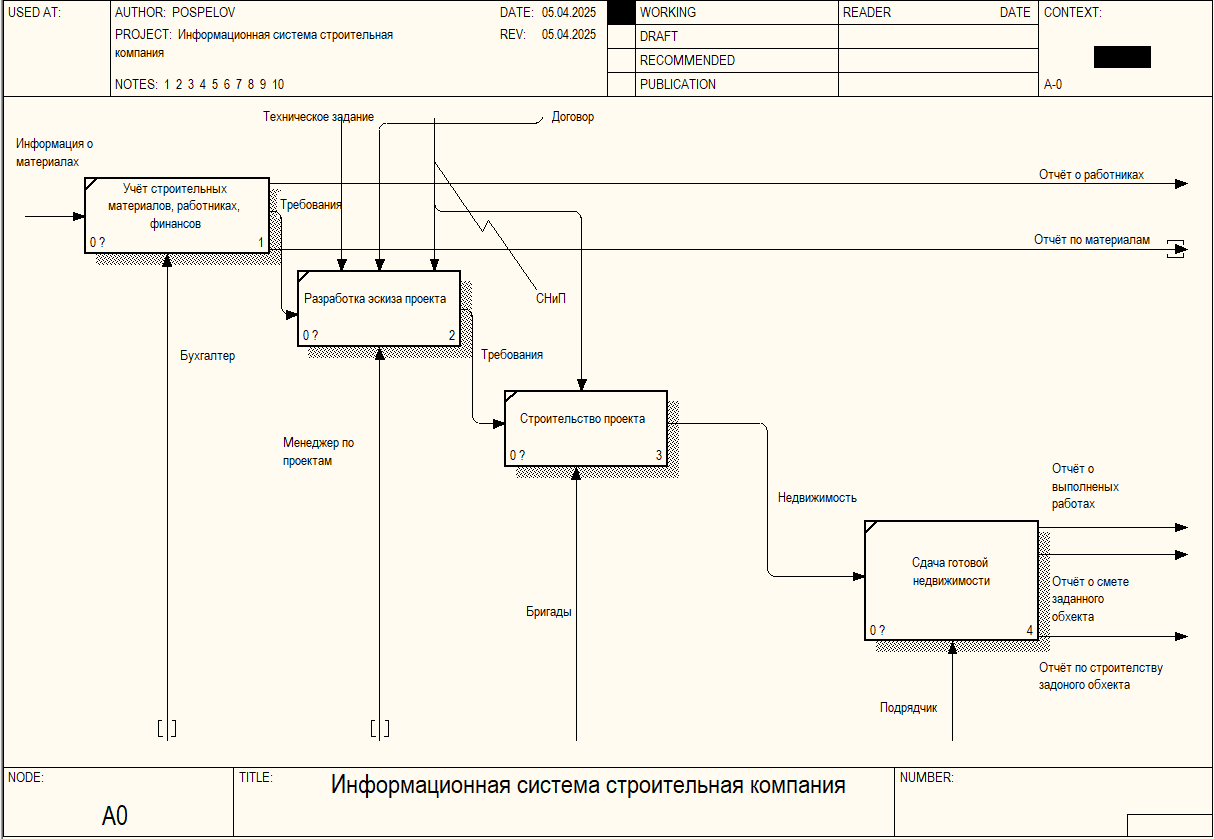
На начальных этапах создания ИС необходимо понять, как работает организация, которую собираются автоматизировать. Никто в организации не знает, как она работает в той мере подробности, которая необходима для создания ИС. Руководитель хорошо знает работу в целом, но не в состоянии вникнуть в детали работы каждого рядового сотрудника. Рядовой сотрудник хорошо знает, что творится на его рабочем месте, но плохо знает, как работают коллеги. Поэтому для описания работы предприятия необходимо построить модель. Такая модель должна быть адекватна предметной области, следовательно, она должна содержать в себе знания всех участников бизнес-процессов организации. Наиболее удобным языком моделирования бизнес-процессов является IDEF0.

Рис. 1.1 Диаграмма Бизнес-процессов (idef0 as is)

Рис. 1.2 Диаграмма Бизнес-процессов (idef0 to be)

Контекстная диаграмма является вершиной древовидной структуры диаграмм и представляет собой самое общее описание системы и ее взаимодействия с внешней средой. После описания системы в целом проводится разбиение ее на крупные фрагменты. Этот процесс называется функциональной декомпозицией, а диаграммы, которые описывают каждый фрагмент и взаимодействие фрагментов, называются диаграммами декомпозиции. После декомпозиции контекстной диаграммы проводится декомпозиция каждого большого фрагмента системы на более мелкие и так далее, до достижения нужного уровня подробности описания. После каждого сеанса декомпозиции проводятся сеансы экспертизы - эксперты предметной области указывают на соответствие реальных бизнес-процессов созданным диаграммам.

*Рис. 1.3 Диаграмма Бизнес-процессов (Decomposition as is)*

функциональная декомпозиция — метод системного проектирования и анализа, который предполагает разбиение сложной системы на более мелкие и управляемые компоненты.

1. Определение основных функций системы.

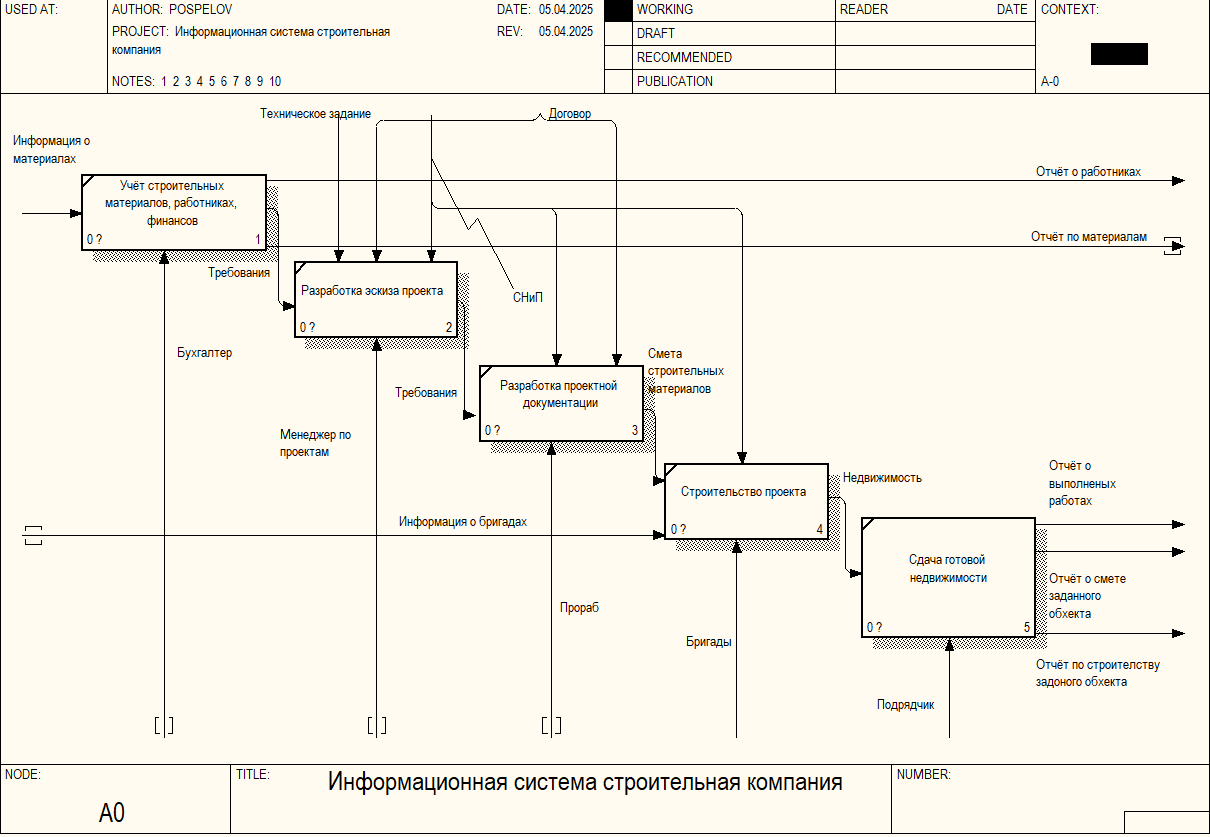
2. Разделение основных функций на подфункции.

3. Определение взаимосвязей между подфункциями.

4. Визуальное представление взаимосвязей.

5. Уточнение функциональной декомпозиции.

Декомпозиция имеет широкий спектр приложений в различных областях, включая разработку программного обеспечения, системную инженерию, совершенствование бизнес-процессов и робототехнику.



*Рис. 1.4 Диаграмма Бизнес-процессов (Decomposition to be)*

# ГЛАВА 2. ВЫБОР ИНСТРУМЕНТАРИЯ

**2.1 Платформа .NET**

Платформа .NET Framework — это технология, которая поддерживает создание и выполнение веб-служб и приложений Windows. При разработке платформы .NET Framework учитывались следующие цели.

1. Обеспечение согласованной объектно-ориентированной среды программирования для локального сохранения и выполнения объектного кода, для локального выполнения кода, распределенного в Интернете, либо для удаленного выполнения.

Предоставление среды выполнения кода, в которой:

* сведена к минимуму вероятность конфликтов в процессе развертывания программного обеспечения и управления его версиями;
* гарантируется безопасное выполнение кода, включая код, созданный неизвестным или не полностью доверенным сторонним изготовителем;
* исключаются проблемы с производительностью сред выполнения скриптов или интерпретируемого кода;

1. Обеспечиваются единые принципы разработки для разных типов приложений, таких как приложения Windows и веб-приложения;
2. Обеспечивается взаимодействие на основе промышленных стандартов, которое гарантирует интеграцию кода платформы .NET Framework с любым другим кодом.

Платформа .NET Framework может размещаться неуправляемыми компонентами, которые загружают среду CLR в собственные процессы и запускают выполнение управляемого кода, создавая таким образом программную среду, позволяющую использовать средства как управляемого, так и неуправляемого выполнения. Платформа .NET Framework не только предоставляет несколько базовых сред выполнения, но также поддерживает разработку базовых сред выполнения независимыми производителями.

**2.2 Язык программирования С#**

C# — современный объектно-ориентированный и типобезопасный язык программирования. C# позволяет разработчикам создавать разные типы безопасных и надежных приложений, выполняющихся в .NET. C# относится к широко известному семейству языков C, и покажется хорошо знакомым любому, кто работал с C, C++, Java или JavaScript. Здесь представлен обзор основных компонентов языка C# 8 и более ранних версий. Если вы хотите изучить язык с помощью интерактивных примеров, рекомендуем поработать с вводными руководствами по C#.

C# — объектно-ориентированный, **ориентированныйна компоненты** язык программирования. C# предоставляет языковые конструкции для непосредственной поддержки такой концепции работы. С момента создания язык C# обогатился функциями для поддержки новых рабочих нагрузок и современными рекомендациями по разработке ПО. В основном C# — **объектно-ориентированный** язык. Вы определяете типы и их поведение.

Вот лишь несколько функций языка C#, которые позволяют создавать надежные и устойчивые приложения. **Сборка мусора** автоматически освобождает память, занятую недостижимыми неиспользуемыми объектами. **Типы, допускающие значение null*,*** обеспечивают защиту от переменных, которые не ссылаются на выделенные объекты. **Обработка исключений** предоставляет структурированный и расширяемый подход к обнаружению ошибок и восстановлению после них. **Лямбда-выражения** поддерживают приемы функционального программирования. **Синтаксис LINQ** создает общий шаблон для работы с данными из любого источника. Поддержка языков для **асинхронныхопераций** предоставляет синтаксис для создания распределенных систем. В C# имеется **Единая система типов**. Все типы C#, включая типы-примитивы, такие как int и double, наследуют от одного корневого типа object.

**2.3 Windows Presentation Foundation (WPF)**

Технология WPF (Windows Presentation Foundation) является часть экосистемы платформы .NET и представляет собой подсистему для построения графических интерфейсов.

Если при создании традиционных приложений на основе WinForms за отрисовку элементов управления и графики отвечали такие части ОС Windows, как User32 и GDI+, то приложения WPF основаны на DirectX. В этом состоит ключевая особенность рендеринга графики в WPF: используя WPF, значительная часть работы по отрисовке графики, как простейших кнопочек, так и сложных 3D-моделей, ложиться на графический процессор на видеокарте, что также позволяет воспользоваться аппаратным ускорением графики.

Одной из важных особенностей является использование языка декларативной разметки интерфейса XAML, основанного на XML: вы можете создавать насыщенный графический интерфейс, используя или декларативное объявление интерфейса, или код на управляемых языках C#, VB.NET и F#, либо совмещать и то, и другое.

Что вам, как разработчику, предлагает WPF?

* Использование традиционных языков .NET-платформы - C#, F# и VB.NET для создания логики приложения
* Возможность **декларативного определения** графического интерфейса с помощью специального языка разметки XAML, основанном на xml и представляющем альтернативу программному созданию графики и элементов управления, а также возможность комбинировать XAML и C#/VB.NET
* **Независимость от разрешения экрана**: поскольку в WPF все элементы измеряются в независимых от устройства единицах, приложения на WPF легко масштабируются под разные экраны с разным разрешением.
* Новые возможности, которых сложно было достичь в WinForms, например, создание трехмерных моделей, привязка данных, использование таких элементов, как стили, шаблоны, темы и др.
* Хорошее **взаимодействие с WinForms**, благодаря чему, например, в приложениях WPF можно использовать традиционные элементы управления из WinForms.
* **Богатые возможности** по созданию различных приложений: это и мультимедиа, и двухмерная и трехмерная графика, и богатый набор встроенных элементов управления, а также возможность самим создавать новые элементы, создание анимаций, привязка данных, стили, шаблоны, темы и многое другое
* **Аппаратное ускорение графики** - вне зависимости от того, работаете ли вы с 2D или 3D, графикой или текстом, все компоненты приложения транслируются в объекты, понятные Direct3D, и затем визуализируются с помощью процессора на видеокарте, что повышает производительность, делает графику более плавной.
* Создание приложений под множество ОС семейства Windows

В тоже время WPF имеет определенные ограничения. Несмотря на поддержку трехмерной визуализации, для создания приложений с большим количеством трехмерных изображений, прежде всего игр, лучше использовать другие средства - DirectX или специальные фреймворки, такие как Monogame или Unity.

Кроме того, несмотря на то что WPF работает поверх кроссплатформенной среды .NET, но в силу природы WPF и зависимости от компонентов Windows, на данный момент создавать приложения на WPF можно только под ОС Windows.

## 2.4 СУБД SQL server

Microsoft SQL Server - система управления реляционными базами данных (РСУБД), разработанная корпорацией Microsoft. Основной используемый язык запросов - Transact-SQL, создан совместно Microsoft и Sybase. Transact-SQL является реализацией стандарта ANЅІЛЅО структурированному языку запросов (SQL) с расширениями. Используется для работы с базами данных размером от персональных до крупных баз данных масштаба предприятия; конкурирует с другими СУБД в этом сегменте рынка.

Особенности функционирования SQL SERVER:

СУБД SQL server используются для создания, размещения, хранения и управления реляционными (табличными) базами данных на специальных серверах или в облаке. Они работают через настольные приложения и web- сайты. К основным преимуществам их функционирования относятся:

* высокоскоростной доступ к данным, обеспечиваемый надежной клиент-серверной архитектурой СУБД;
* простота работы и администрирования, обусловленные понятной структурой языка программирования SQL;
* безопасность хранения информации в БД – благодаря возможности шифрования данных и резервного копирования.
* Специфика работы сервера базы данных SQL server заключается в транзакционной обработке данных. Это означает, что по каждому запросу от СУБД обрабатывается и сохраняется небольшое количество информации.

Применение SQL server позволяет автоматизировать решение различных бизнес-задач, поддерживать проведение аналитики данных в режиме онлайн, отслеживать направление ресурсов СУБД, управлять транзакциями (операциями по обработке данных).

Виды SQL-server:

Для реализации функций СУБД на сегодняшний день чаще всего используются следующие SQL-серверы:

MS SQL server - многопользовательский программный продукт, разработанный компаний Microsoft, обладающий высокой производительностью и отказоустойчивостью, тесно интегрированный с ОС Windows. Этот сервер поддерживает удаленные подключения, работает с многими популярными типами данных, дает возможность создавать триггеры и хранимые данные, имеет практичные и удобные утилиты для настройки;

Oracle Database server СУБД, предназначенная для создания, консолидации и управления базами данных в облачной среде. Используя этот сервер, можно как автоматизировать обычные бизнес-операции, так и выполнять динамический многомерный анализ данных (OLAP), проводить операции с документами xml-формата и управлять разделенной и локальной информацией;

ІВМ DB2 - семейство СУБД для работы с реляционными базами данных, признанное самым производительным, имеющим высокие технические показатели и возможности масштабирования. SQL-серверы этой группы характеризуются мультиплатформенностью, способностью к мгновенному созданию резервных копий и восстановлению БД, реорганизации таблиц в онлайн-режиме, разбиению баз данных, определению пользователями новых типов данных;

MySQL - СУБД, разработанная и поддерживаемая компанией Oracle. В основном она используется локальными или удаленными клиентами, позволяя им работать с таблицами разных типов, поддерживающих полнотекстовый поиск или выполняющих транзакции на уровне отдельных записей;

PostgreSQL – СУБД с открытым исходным кодом, работающая с объектно-реляционными базами данных. Также PostgreSQL предназначена для создания, хранения и извлечения сложных структур данных. Она поддерживает самые различные типы данных.

**2.5 Microsoft SQL Server Management Studio**

SQL Server Management Studio (SSMS) утилита из Microsoft SQL Server 2005 и более поздних версий для конфигурирования, управления и администрирования всех компонентов Microsoft SQL Server. Утилита включает скриптовый редактор и графическую программу, которая работает с объектами и настройками сервера.

Главным инструментом SQL Server Management Studio является Object Explorer, который позволяет пользователю просматривать, извлекать объекты сервера, а также полностью ими управлять.

Также есть SQL Server Management Studio Express для Express версии сервера, которая является бесплатной. Однако в ней нет поддержки ряда компонентов (Analysis Services, Integration Services, Notification Services, Reporting Services) и SQL Server 2005 Mobile Edition.

Среда SQL Server Management Studio предназначена для доступа к службам Службы Analysis Services, Integration Services и Reporting Services, a также для их настройки, администрирования и управления ими. Хотя все три технологии бизнес-аналитики полагаются на среду SQL Server Management Studio, административные задачи, связанные с каждой из этих технологий, несколько отличаются.

ЅЅМЅ работает только в Windows (AMD или Intel). Если вам требуется от Windows, средство, которое работает на платформах, отличных — это кроссплатформенное рассмотрите Azure Data Studio. Azure Data Studio средство для macOS, Linux и Windows.

Среда SQL Server Management Studio позволяет включать компоненты служб Reporting Services, администрировать серверы и базы данных, управлять ролями и заданиями.

**2.6 Entity Framework**

Entity Framework это набор технологий в ADO.NET, которые поддерживают разработку программных приложений, ориентированных на данные. Архитекторам и разработчикам приложений, ориентированных на обработку данных, приходится учитывать необходимость достижения двух совершенно различных целей. Они должны моделировать сущности, связи и логику решаемых бизнес-задач, а также работать с ядрами СУБД, используемыми для сохранения и получения данных. Данные могут распределяться по нескольким системам хранения данных, в каждой из которых применяются свои протоколы, но даже в приложениях, работающих с одной системой хранения данных, необходимо поддерживать баланс между и требованиями системы хранения данных и требованиями написания эффективного и удобного для обслуживания кода приложения.

Платформа Entity Framework позволяет работать с данными в форме специфических для домена объектов и свойств (например, с клиентами и их адресами) без необходимости учитывать формат базовых таблиц и столбцов базы данных, где хранятся эти данные. Entity Framework дает разработчикам возможность работать с данными на более высоком уровне абстракции, создавать и сопровождать приложения, ориентированные на работу с данными, одновременно с этим сокращая объем кода по сравнению с традиционными приложениями. Поскольку Entity Framework компонентом платформа .NET Framework, Entity Framework приложения могут работать на любом компьютере, на котором установлена платформа .NET Framework с пакетом обновления 1 (SP1) версии 3,5.

Entity Framework дает жизнь модели, позволяя разработчикам запрашивать сущности и связи в модели предметной области (называемой концептуальной моделью в Entity Framework), в то же время полагаться на Entity Framework преобразования этих операций в команды, относящиеся к источнику данных. Это позволяет отказаться от применения в приложениях жестко заданных зависимостей от конкретного источника данных.

## ГЛАВА 3. РАЗРАБОТКА БАЗЫ ДАННЫХ

**3.1 Разработка базы данных**

# 3.2 Программирование информационной системы

# ЗАКЛЮЧЕНИЕ

# СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕАРТУРЫ