**СОДЕРЖАНИЕ**

[ВВЕДЕНИЕ 3](#_Toc44095151)

[1.ОПИСАНИЕ СТРУКТУРЫ ПРЕДПРИЯТИЯ. 4](#_Toc44095152)

[2.ВЫБОР НАПРАВЛЕНИЯ АВТОМАТИЗИРУЕМОЙ ОБЛАСТИ ДЕЯТЕЛЬНОСТИ 5](#_Toc44095153)

[3.ФОРМИРОВАНИЕ ПОСТАНОВКИ ЗАДАЧИ 6](#_Toc44095154)

[4.ФОРМИРОВАНИЕ ТРЕБОВАНИЙ К ИНФОРМАЦИОННОЙ СИСТЕМЕ 7](#_Toc44095155)

[5.ВЫБОР ТРЕБУЕМОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ДЛЯ РЕШЕНИЯ ЗАДАЧИ 8](#_Toc44095156)

[6.РАЗРАБОТКА И ОФОРМЛЕНИЕ ТЕХНИЧЕСКОГО ЗАДАНИЯ НА ИНФОРМАЦИОННУЮ СИСТЕМУ 10](#_Toc44095157)

[7.РАЗРАБОТКА ИНФОРМАЦИОННО-ЛОГИЧЕСКОЙ МОДЕЛИ ПРЕДМЕТНОЙ ОБЛАСТИ 13](#_Toc44095158)

[8.РАЗРАБОТКА ТРЕБОВАНИЙ БЕЗОПАСНОСТИ ИНФОРМАЦИОННОЙ СИСТЕМЫ 16](#_Toc44095159)

[9.ПРОЕКТИРОВАНИЕ И РАЗРАБОТКА БАЗ ДАННЫХ 17](#_Toc44095160)

[10.ПРОЕКТИРОВАНИЕ И РАЗРАБОТКА ИНТЕРФЕЙСА ИНФОРМАЦИОННОЙ СИСТЕМЫ 19](#_Toc44095161)

[11.РАЗРАБОТКА АЛГОРИТМОВ И ПРОГРАММ ОТДЕЛЬНЫХ МОДУЛЕЙ ИНФОРМАЦИОННОЙ СИСТЕМЫ 21](#_Toc44095162)

[12.РАЗРАБОТКА ТЕСТОВ ДЛЯ КОНТРОЛЯ ПРАВИЛЬНОСТИ РАБОТЫ 26](#_Toc44095163)

[13.ПРОВЕДЕНИЕ ТЕСТИРОВАНИЯ И ОТЛАДКИ РАЗРАБАТЫВАЕМЫХ ПРИЛОЖЕНИЙ 28](#_Toc44095164)

[14.РАЗРАБОТКА РУКОВОДСТВА ПО ИНСТАЛЛЯЦИИ ИНФОРМАЦИОННОЙ СИСТЕМЫ 29](#_Toc44095165)

[15.РАЗРАБОТКА РУКОВОДСТВА ПОЛЬЗОВАТЕЛЯ ИНФОРМАЦИОННОЙ СИСТЕМЫ 34](#_Toc44095166)

[16.ПРОВЕДЕНИЕ ОЦЕНКИ КАЧЕСТВА И ЭКОНОМИЧЕСКОЙ ЭФФЕКТИВНОСТИ ИНФОРМАЦИОННОЙ СИСТЕМЫ 35](#_Toc44095167)

[ЗАКЛЮЧЕНИЕ 43](#_Toc44095168)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 44](#_Toc44095169)

# ВВЕДЕНИЕ

Целью практики является закрепление первоначального практического опыта, полученного мною при изучении ПМ.05 Проектирование и разработка информационных систем.

Задачами практики являются:

* описание структуры предприятия;
* выбор направления автоматизируемой области деятельности подразделения;
* формирование постановки задачи
* формирование требований к информационной системе (ИС);
* выбор требуемого программного обеспечения для решения задачи;
* разработка и оформление технического задания на ИС;
* разработка информационно-логической модели предметной области;
* разработка требований безопасности информационной системы;
* проектирование и разработка баз данных;
* проектирование и разработка интерфейса ИС;
* разработка алгоритмов и программ отдельных модулей информационной системы;
* разработка тестов для контроля правильности работы;
* проведение тестирования и отладки разрабатываемых приложений;
* разработка руководства по инсталляции ИС;
* разработка руководства пользователя ИС;
* проведение оценки качества и экономической эффективности информационной системы.

# Описание структуры предприятия.

Компания ООО «Синаптик» располагается на территории Кировской области в городе Кирове. Организация работает в сфере деятельности разработки компьютерного программного обеспечения, интернет-маркетинга разработка и поддержка web-проектов.

# ВЫБОР НАПРАВЛЕНИЯ АВТОМАТИЗИРУЕМОЙ ОБЛАСТИ ДЕЯТЕЛЬНОСТИ

Компания «Synaptik» занимается созданием сайтов на заказ. При большом количестве заказов появляется проблема учета бизнес-процессов.

Информационная система служит для облегчения и уменьшения времени учета проектов. Данные бизнес-процессы можно облегчить с помощью:

1. Ведение базы данных сотрудников.
2. Ведение базы данных проектов.
3. Ведение базы данных типа проекта.
4. Ведение базы данных клиентов.
5. Ведение базы данных цен.

Исходя из предложенных бизнес-процессов было выбрано использование автоматизированной системы управления.

Автоматизированная Система Управления (АСУ) –комплекс аппаратных и программных средств, а также персонала, предназначенный для управления различными процессами в рамках технологического процесса, производства, предприятия.

Главной задачей АСУ является повышение эффективности управления объектом на основе роста производительности труда и совершенствования методов планирования процесса управления.

# Формирование постановки задачи

Основной задачей производственной практики является создание информационной системы для контроля бизнес-процессов. Данная ИС упростит и уменьшит затраченное время учета проектов. Система создана для сотрудников компании «Синаптик».

# **Формирование требований к информационной системе**

Требования к системе:

1. Форма авторизации

Должна давать возможность входа в систему при правильном вводе логина и пароля

1. Форма учета проектов

Должна давать возможность вывода проекта в одну из таблиц статусов проекта.

1. Форма добавления проектов

Должна дать возможность добавить имя проекта в базу данных.

1. Форма добавления сотрудников

Должна дать возможность добавить, изменить или удалить информацию о сотруднике.

1. Форма Price List

Должна дать возможность ввести данные о ценовой политике компании в базу данных.

1. Форма добавления типов проекта

Должна дать возможность добавление типа проекта в базу данных.

1. Форма добавления клиентов

Должна дать возможность добавить, изменить или удалить данные о клиенте в базе данных.

# **Выбор требуемого программного обеспечения для решения задачи**

В качестве IDE для программирования на C# была выбрана среда разработки Visual Studio, т.к. язык и среда разработки созданы Microsoft и более удобного IDE для создания приложения на платформе .NET не найти.

В качестве примера могу предложить JetBrains.

Минусы данной IDE:

1. Молодость. Часть функциональности еще в разработке, не все баги исправлены.
2. Стоимость. Самая дешевая версия Project Rider стоит 139 долларов за первый год использования.

В качестве еще одного пример могу предложить Eclipse.

Плюсы:

1. Множество плагинов.
2. Кастомизация.
3. Бесплатность.
4. Высокая функциональность.

Минусы:

1. Сложность. Для новичка может показаться слишком сложным.
2. Нет гарантии надежности. Так как плагины создаются сообществом, за их качество отвечает только разработчик.

Сред для разработки кода на C# не мало и у каждой есть свои плюсы и минусы. В качестве IDE ,ыло выбрано Visual Studio потому что в данной программе присутствует много удобных инструментов для создания приложений.

В качестве программы для создания баз данных была выбрана SQL Server Management Studio.

SQL Server Management Studio (SSMS) – утилита для создания, управления и администрирования всех компонентов Microsoft SQL Server.

Плюсы данной СУБД:

1. Надежность.
2. Безопасность.
3. Бесплатность.

Также существует 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.

# Разработка и оформление технического задания на информационную систему

1. Общие сведения.

1.1. Наименование системы.

1.1.1. Полное наименование системы.

Система управления бизнес-процессов «Синаптик».

1.1.2. Краткое наименование системы.

СУБП «Синаптик».

1.2. Основания для проведения работ.

Работа выполняется на основании договора № 0001 от 11.06.2020 между Токаревым Павлом Михайловичем и ООО «Синаптик».

1.3. Наименование организаций – Заказчика и Разработчика.

1.3.1. Заказчик.

Заказчик: ОOО «Синаптик».  
Адрес фактический: г. Киров, ул. Володарского, д. 108, кв 11, 610002.  
Телефон / Факс: 8 (922) 977-84-24.

1.3.2. Разработчик.

Разработчик: Токарев Павел Михайлович.  
Адрес фактический: г. Киров, ул. Ленина, 184/3.  
Телефон / Факс: +7 (958) 3935357.

1.4. Плановые сроки начала и окончания работы.

Начало работы: 11.06.2020 Конец работы: 01.07.2020.

2. Назначение и цели создания системы

2.1. Назначение системы.

Вид деятельности: учет проектов.

Перечень автоматизированных процессов:

* учет клиентов;
* учет цен;
* учет проектов;
* учет сотрудников
* уменьшение бумажного документ оборота;
* оптимизация работы.

2.2. Цели создания системы

СУБП «Синаптик» создается с целью:

* упрощения учета проектов;
* уменьшения времени учета проектов;
* оптимизации работы.

3. Характеристика объектов автоматизации

Область деятельности компании:

Компания «Синаптик» занимается разработкой компьютерного программного обеспечения.

Рисунок 1 – Структура компании

4. Требования к системе в целом.

* ИС должна соответствовать требованиям ТЗ;
* сократить время и упростить процесс учета проектов.

Таблица 1 – Задачи автоматизации

|  |  |
| --- | --- |
| Функция | Задача |
| Редактирование списка сотрудников | Добавление нового сотрудника |
| Редактирование информации о сотруднике |
| Удаление сотрудника |
| Редактирование списка проектов | Добавление нового проекта |
| Изменение информации о проекте |
| Удаление проекта |
| Редактирование списка клиентов | Добавление новой клиента |
| Редактирование информации о клиенте |
| Удаление информации о клиенте |

# Разработка информационно-логической модели предметной области

1. Разработка концептуальной модели

Концептуальная модель – это модель, представленная множеством понятий и связей между ними, определяющих смысловую структуру рассматриваемой предметной области или её конкретного объекта.

Концептуальная модель отражает такие бизнес процессы в компании Синаптик как:

* ведение базы данных сотрудников;
* ведение базы данных проектов;
* ведение базы данных клиентов.

Бизнес-процессы компании (рисунок 1):

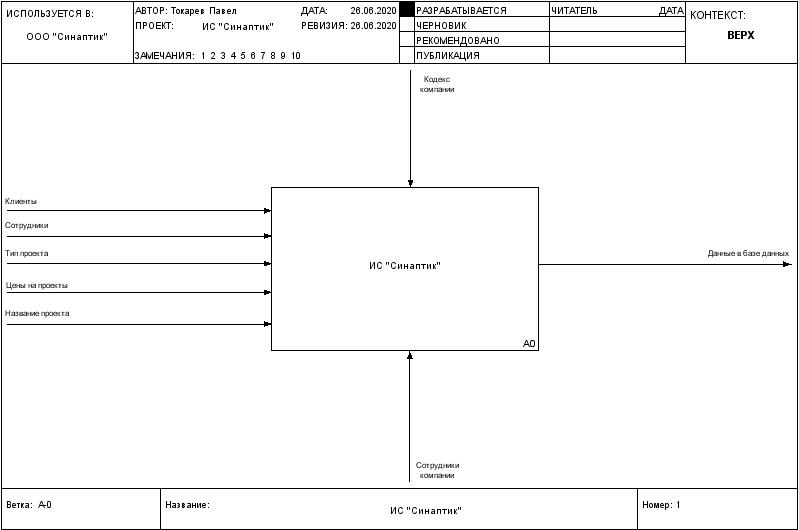


Рисунок 1 – Нулевая модель IDEF0

Входящие данные в ИС являются: информация о клиенте, информация о сотруднике, информация о типе проекта, информация о цене, название проекта.

Нормативно-правовыми документами являются: кодекс компании.

Механизмом управления являются: сотрудники компании.

Выходными данными является: таблица в базе данными с проектами.

Декомпозиция бизнес-процессов (рисунок 2) – это научный метод, использующий структуру задачи и позволяющий заменить решение одной большой задачи решением серии меньших задач.

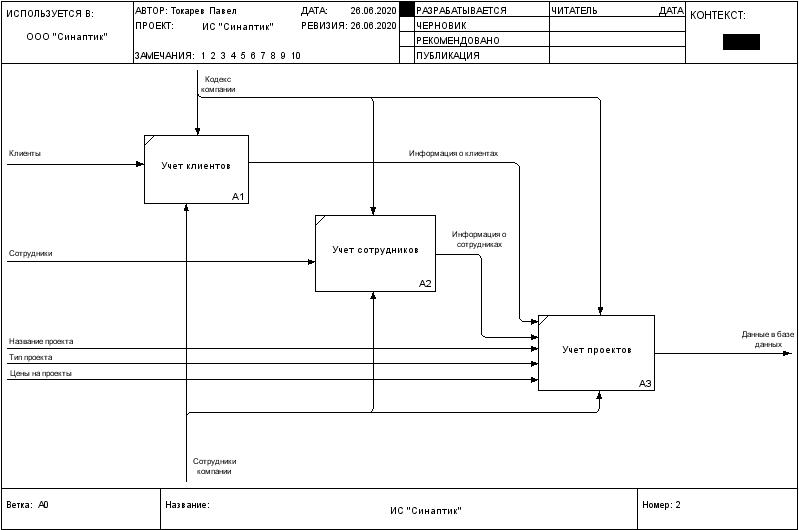


Рисунок 2 – Декомпозиция модели IDEF0

1. Разработка логической модели

Логическая схема – модель базы данных, выраженная в понятиях модели данных. Логическая модель содержим все объекты рассматриваемой предметной области и связи между ними. Для графического представления логической модели будет использовать нотацию IDЕF1X (рисунок 3).

IDEF1 применяется для построения информационной модели, которая представляет структуру информации, необходимой для поддержки функций производственной системы или среды.

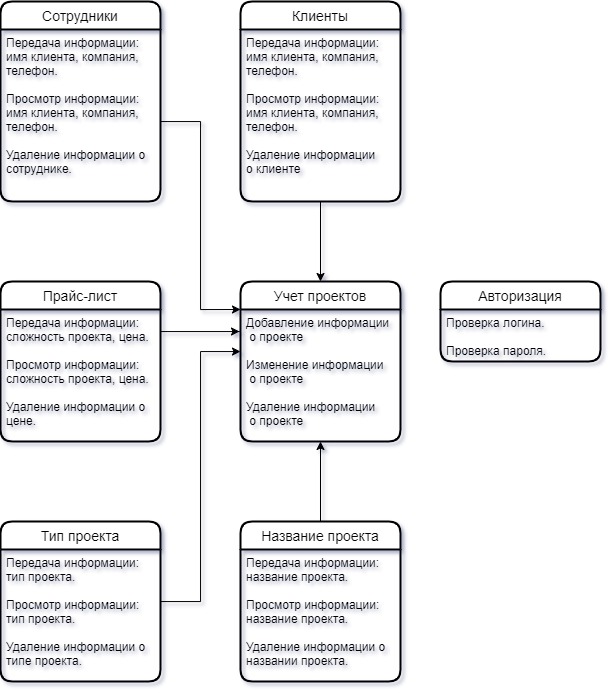


Рисунок 3 – Логическая модель

# Разработка требований безопасности информационной системы

Для использования информационной системы нужно войти под учетной записью пользователя. В случае если пользователь введет неправильный логин или пароль система не даст таблицами.

# Проектирование и разработка баз данных

Физическая модель – физическое представление системы, объекта или процесса с целью их исследования, то есть это представление с помощью другого физического, реального объекта, имеющего в том или ином аспекте схожую динамику поведения. (Рисунок 4)

Для работы с базой данных использовалась SQL Server Management Studio

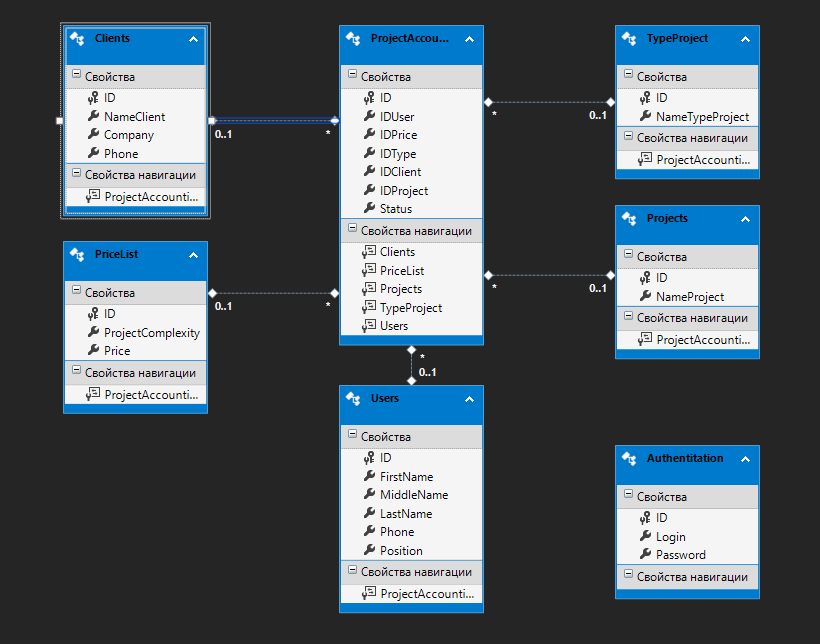


Рисунок 4 – физическая модель

Таблица 3 – Описание таблиц баз данных

|  |  |  |  |
| --- | --- | --- | --- |
| Таблица | Атрибут | Тип данных | Размер |
| Authentitation | ID(PK) | int | 4 байта |
| Login | nvarchar | (MAX) |
| Password | nvarchar | (MAX) |
| Clients | ID(PK) | int | 4 байта |
| NameClient | nvarchar | (MAX) |
| Company | nvarchar | (MAX) |
| Phone | nvarchar | (MAX) |
| PriceList | ID(PK) | int | 4 байта |
| ProjectСomplexity | nvarchar | (MAX) |
| Price | nvarchar | (MAX) |
| ProjectAccounting | ID(PK) | int | 4 байта |
| IDUser (FK к Users столб. ID) | int | 4 байта |
| IDPrice (FK к Price столб.) | int | 4 байта |
| IDType (FK к TypeProject столб. ID) | int | 4 байта |
| IDClient (FK к Clients столб. ID) | int | 4 байта |
| IDProject (FK к Projects столб. ID) | int | 4 байта |
| Status | nvarchar | (MAX) |
| Projects | ID(PK) | int | 4 байта |
| NameProject | nvarchar | (MAX) |
| TypeProject | ID(PK) | int | 4 байта |
| NameTypeProject | nvarchar | (MAX) |
| Users | ID(PK) | int | 4 байта |
| FirstName | nvarchar | (MAX) |
| MiddleName | nvarchar | (MAX) |
| LastName | nvarchar | (MAX) |
| Phone | nvarchar | (MAX) |
| Position | nvarchar | (MAX) |

# Проектирование и разработка интерфейса информационной системы

Компания имеет логотип (Рисунок 5)



Рисунок 5 – Логотип

В качестве иконки использовалась картинка (Рисунок 6)



Рисунок 6 – Иконка

Стиль форм (Рисунок 7):

* цветовая гамма была выбрана белая
* прозрачный задний фон (90%)
* кнопки без теней
* шрифт был выбран Microsoft Sans Serif 8,25pt
* высота элементов подобрана вручную.

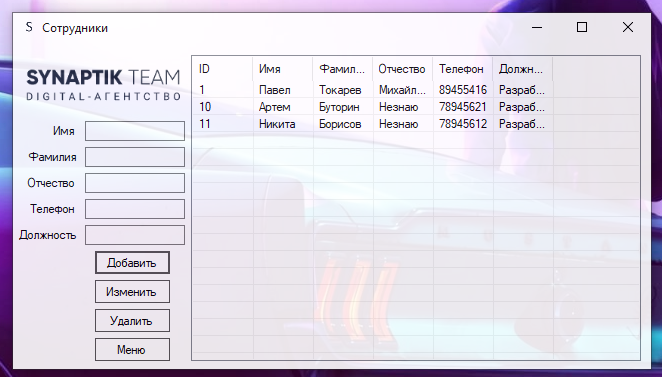


Рисунок 7 – Стиль форм

Схема взаимодействия компонентов представлена на рисунке 8.

Рисунок 8 – Взаимодействие компонентов

# Разработка алгоритмов и программ отдельных модулей информационной системы

Схема включает в себя модули: авторизация, меню, учет проектов, добавление проектов, сотрудники, Price List, тип проектов, клиенты.

В модуле «авторизация» для входа в систему сотруднику нужно указать данные учетной записи (рисунок 9).

Код информационной системы прикреплен в приложении А

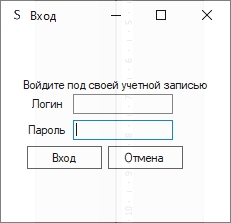


Рисунок 9 – Авторизация

В модуле «Меню» сотруднику предоставляется выбор дальнейших действий (Рисунок 10)

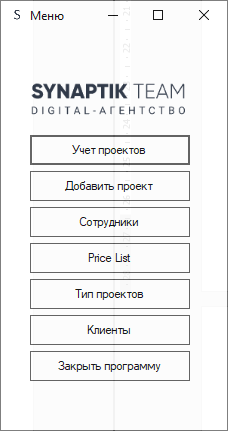


Рисунок 10 – Меню

В модуле «Учет проектов» сотрудник может внести данные о проекте (Рисунок 11)

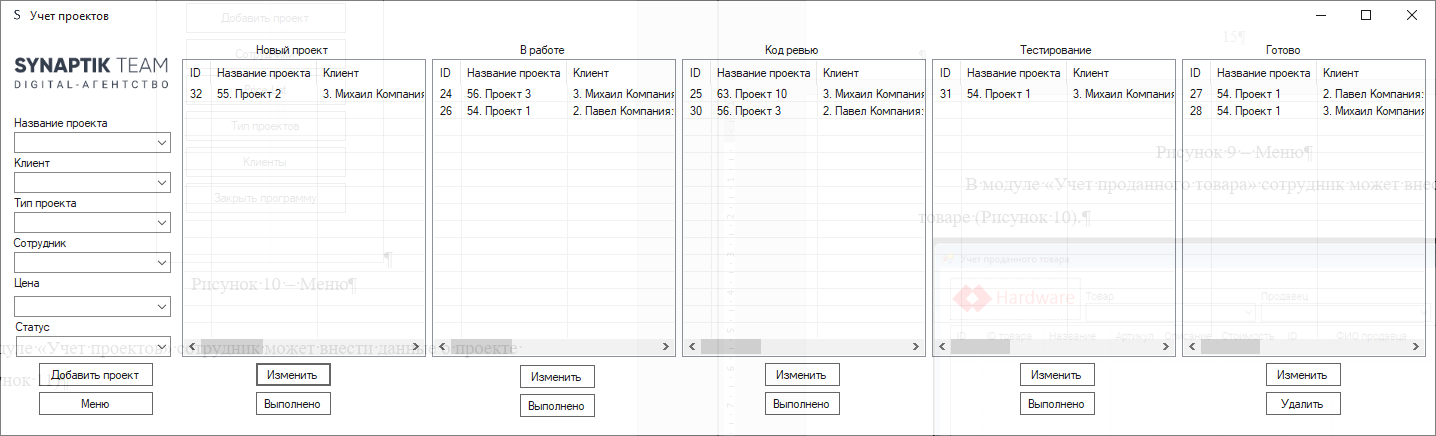


Рисунок 11 – Учет проектов

В модуле «Добавить проект» сотрудник может внести название проекта (Рисунок 12)

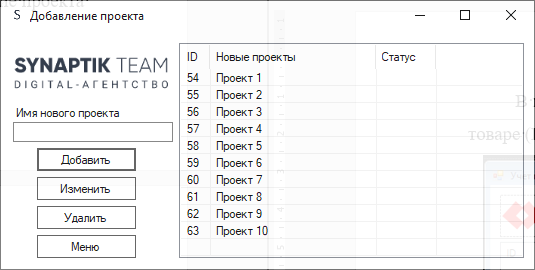
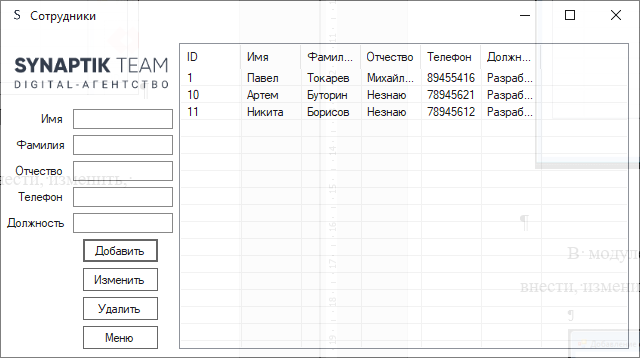


Рисунок 12 – Добавить проект

В модуле «Сотрудники» сотрудник компании может внести, изменить, удалить данные о сотруднике (Рисунок 13).

  
Рисунок 13 – Сотрудники

В модуле «Price List» сотрудник компании может внести информацию о цене проекта (Рисунок 14).

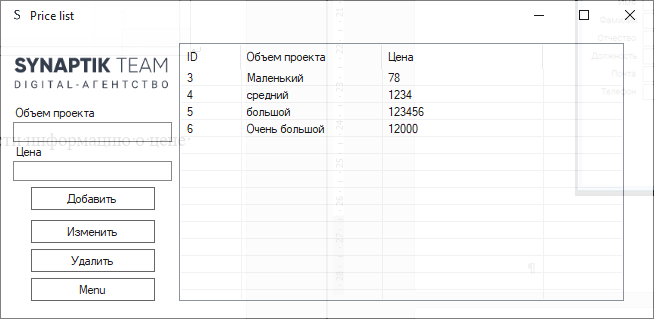


Рисунок 14 – Price List

В модуле «Тип проекта» сотрудник компании может внести информацию о типе проекта (Рисунок 15).

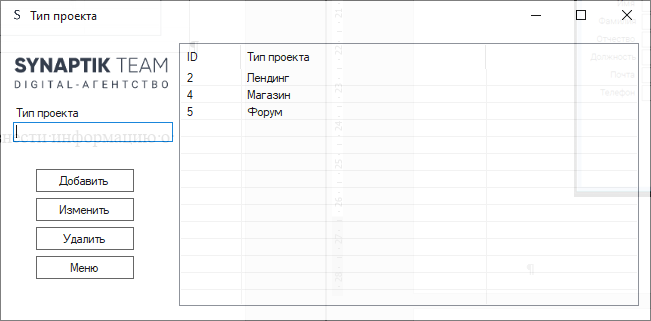


Рисунок 15 – Тип проекта

В модуле «Клиенты» сотрудник компании может внести информацию о клиенте (Рисунок 16).

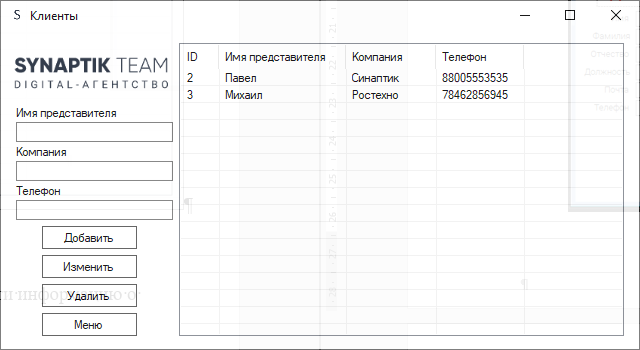


Рисунок 16 – Клиенты

# Разработка тестов для контроля правильности работы

Тестирование – это проверка работы программ с данными, подобным реальным, которые будут обрабатываться в процессе эксплуатации системы. Процесс тестирования программного обеспечения осуществляется на основе фактических или смоделированных входных данных (как стандартных, так и не стандартных) при определённых контролируемых условиях.

Тестирование модулей и в частности тестирование разработанных компонентов является обязательной составляющей процесса аттестации и верификации разрабатываемой подсистемы.

Из существующих способов тестирования был выбран «черный ящик». Этот способ является одним из наиболее устоявшихся способов обеспечения качества разработки программного обеспечения и входит в набор эффективных средств современной системы обеспечения качества программного продукта.

Для тестирования разрабатываемого проекта была выбрана методика тестирования «черного ящика». Эта методика применяется в качестве средства тестирования функционала разрабатываемого программного обеспечения.

Цель метода состоит в том, чтобы протестировать работоспособность программного обеспечения исходя из спецификации выполняемых системой функций.

При таком подходе система представляется неким черным ящиком, у которого имеется вход и выход. На входе мы имеем входные данные, на выходе – переработанные системой данные.

Таблица 4 – Набор тестов

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Действия | Входные данные | Ожидаемый результат |
| Вход в систему | Ввести логин и пароль | Логин: admin  Пароль: admin | Вход в систему |
| Вход в систему | Ввести логин и пароль | Логин: -  Пароль: - | Сообщение «Логин или пароль неверны» |
| Тестирование формы «Добавить проект» | Ввести данные о проекте | Имя проекта: New Project | Новый проект добавляется в таблицу |
| Тестирование формы «Добавить проект» | 1. Выбрать строчку 2. Изменить название проекта в текстовом поле 3. Нажать кнопку «Изменить» | Имя нового проекта: New Project | Имя проекта изменяется в таблице |
| Тестирование формы «Добавить проект» | 1. Выбрать строчку 2. Нажать на кнопку «Удалить» | Нет | Имя проекта удаляется из таблицы |
| Тестирование формы «Учет проектов» | 1)Ввести данные в поля с выбором  2)Нажать на кнопку «Добавить» | Любые | Новый проект добавляется в таблицу |
| Тестирование формы «Учет проектов» | 1)Ввести данные в поля с выбором  2)Нажать на кнопку «Добавить | Любые | Новый проект добавляется в таблицу |
| Тестирование формы «Учет проектов» | 1)Ввести данные в поля с выбором  2)Нажать на кнопку «Добавить | Любые | Новый проект добавляется в таблицу |

# Проведение тестирования и отладки разрабатываемых приложений

Таблица 5 – Набор тестов

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Действия | Входные данные | Реальный результат |
| Вход в систему | Ввести логин и пароль | Логин: admin  Пароль: admin | Успешный вход |
| Вход в систему | Ввести логин и пароль | Логин: -  Пароль: - | Сообщение «Логин или пароль неверны» |
| Тестирование формы «Добавить проект» | Ввести данные о проекте | Имя проекта: New Project | Успешное добавление нового проекта |
| Тестирование формы «Добавить проект» | 1. Выбрать строчку 2. Изменить название проекта в текстовом поле 3. Нажать кнопку «Изменить» | Имя нового проекта: New Project | Успешное изменение название проекта |
| Тестирование формы «Добавить проект» | 1. Выбрать строчку 2. Нажать на кнопку «Удалить» | Выбрать строчку New Project | Успешное удаление строчки New Project |
| Тестирование формы «Учет проектов» | 1)Ввести данные в поля с выбором  2)Нажать на кнопку «Добавить» | Любые | Успешное добавление нового проекта |
| Тестирование формы «Учет проектов» | 1. Выбрать строчку 2. Нажать на кнопку «Удалить» | Любую | Успешное удаление строчки |

# Разработка руководства по инсталляции информационной системы

Для установки информационной системы необходимо установить Server Management Studio:

1. Установка SQL Server Management Studio (SSMS) – везде жмём далее и ждём завершения установки. (Доп. материал по установке: https://metanit.com/sql/sqlserver/1.3.php).

2. Установка SQL Server Express:

a. Выбираем тип установки «Базовый», щёлкнув по нему (рисунок 17):

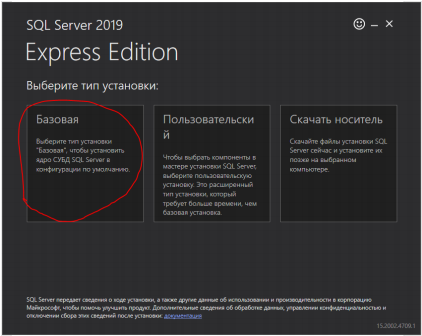


Рисунок 17 – Выбор типа установки

b. Далее принимаем лицензионное соглашение и жмём «Далее» (рисунок 18):

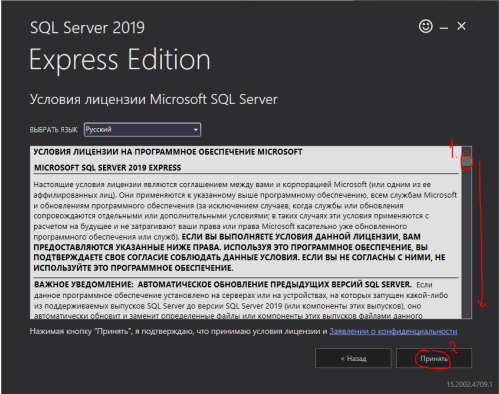


Рисунок 18 – Лицензионное соглашение

c. Место расположения оставляем как есть и жмём «Установить». После завершения (т.к. SSMS у нас уже установлен) жмём просто «Закрыть». При желании можете установить SSMS при установке SQL Server

3. Установка Visual Studio

Инструкция по установке: <https://docs.microsoft.com/ru-ru/visualstudio/install/install-visualstudio?view=vs-2019>.

1. При выборе рабочих нагрузок в указанной выше ссылке, выберем следующее (см. рисунок 19), также данные пункты можно будет изменить позже.

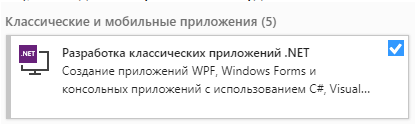


Рисунок 19 – Выбор рабочих нагрузок

b. Далее следуем инструкции в указанной ссылке.

4. Далее необходимо добавить базу данных

1. Скачайте файл https://drive.google.com/file/d/1oHYSCHlXo9KjAqcfkp8LwkrzfNDwK1PG/view?usp=sharing и распакуйте.
2. Зайдите в SQL Server Management Studio и нажмите "Соединиться"

(рисунок 20).

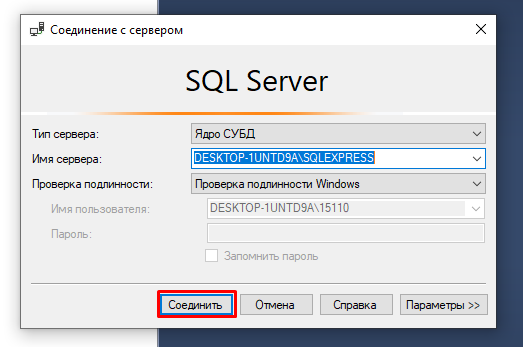


Рисунок 20

1. Нажмите "Импорт приложения уровня данных" (рисунок 21).

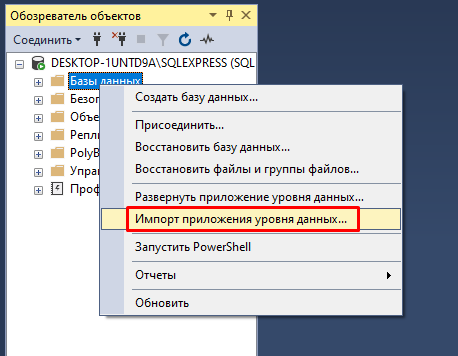


Рисунок 21 – Импорт базы данных

1. Нажмите "Обзор" и выдерете путь к базе данных (рисунок 22).

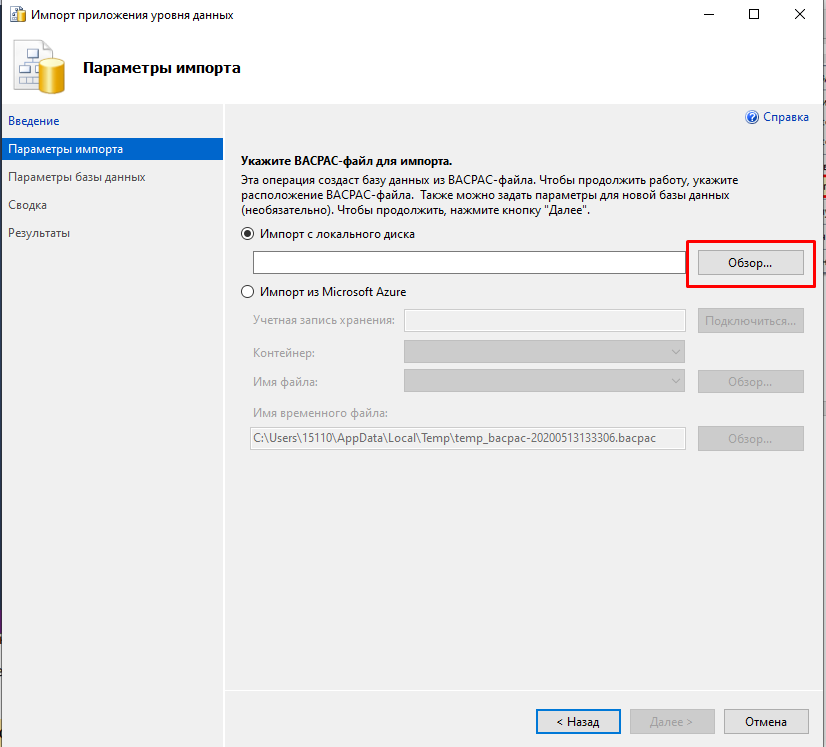


Рисунок 22 – выбор базы данных

1. Измените модель базы данных в проекте в Visual Studio и скомпилируйте.
2. Далее открываем папку с проектом и переходим по такому пути:

ProjectForSynaptic\bin\Debug.

1. Создаем ярлык для файла ProjectForSynaptic.exe и переносим на рабочий стол

Готово.

# Разработка руководства пользователя информационной системы

Данная информационная система предназначена для учета проектов компании. Пользователями системы являются сотрудники компании.

Краткое описание возможностей. В ИС предусмотрены возможности введения базы данных клиентов, сотрудников и проектов,

# Проведение оценки качества и экономической эффективности информационной системы

# ПРОВЕДЕНИЕ ОЦЕНКИ КАЧЕСТВА И ЭКОНОМИЧЕСКОЙ ЭФФЕКТИВНОСТИ ИНФОРМАЦИОННОЙ СИСТЕМЫ

В данном пункте рассматриваются вопросы финансово-экономического обоснования разработки данной системы. Проводится экономическое обоснование необходимости разработки, расчёт затрат на создание программы, внедрение и использование, а также расчёт цены и прибыли от реализации.

Обоснование необходимости разработки ИС.

Данная система позволяет упростить работу сотрудников компании, а именно процесс учета проектов.

В связи с этим можно утверждать об актуальности разработки программного продукта, с её помощью можно быстро получить доступ к определённой информации предприятия.

Использование данной ИС позволяет сократить время на учет проектов.

Расчёт затрат на создание информационной системы.

Таблица 1 – Состав разработчиков

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Наименование | Численность, человек (Р) | Базовая ставка заработной платы, рублей | Повышающий коэффициент | Месячный оклад, рублей () |
| Практикант | 1 | 12130 | 1,6 | 19408 |

Размер базовой ставки заработной платы равен минимальной заработной плате в Кировской области, которая составляет 12130 рублей. В зависимости от повышающего коэффициента был получен месячный оклад.

Произведём расчёт трудоёмкости выполнения отдельных видов работ по разработке ИС. Трудоёмкость выполнения работ следует рассчитать, используя формулу.

,

где – расчётная трудоёмкость выполнения работы;

– минимальное время, необходимое для выполнения работы;

– минимальное время, необходимое для выполнения работы.

Трудоёмкость выполнения работ приведена в таблице 2.

Таблица 2 – Трудоёмкость выполнения работ

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № строки | Наименование работ |  |  |  | Практиканта |
| 1 | Описание структуры предприятия | 6 | 12 | 11,4 | 6,1 |
| 2 | Выбор направления автоматизируемой области | 2 | 8 | 7,4 | 4,3 |
| 3 | Разработка и оформление технического задания для ИС | 2 | 6 | 5,6 | 3,5 |
| 4 | Разработка информационно-логической модели предметной области | 2 | 6 | 5,6 | 3 |
| 5 | Разработка требований безопасности системы | 2 | 6 | 5,6? | 2,1 |
| 6 | Проектирование и разработка баз данных | 2 | 6 | 5,6 | 4 |
| 7 | Проектирование и разработка интерфейса ИС | 2 | 6 | 5,6 | 5,2 |
| 8 | Разработка алгоритмов и программ отдельных модулей ИС | 10 | 18 | 17,2 | 15,4 |
| 9 | Разработка тестов и проведение тестирования | 5 | 12 | 11,3 | 7,6 |
| 10 | Разработка руководства по инсталляции ИС | 2 | 6 | 5,6 | 2,2 |
| 11 | Разработка руководства пользователя ИС | 2 | 6 | 5,6 | 3,3 |
| 12 | Проведение оценки качества и экономической эффективности ИС | 5 | 10 | 9,5 | 8,5 |
| 13 | Работы после разработки | 2 | 6 | 5,6 | 5,2 |
|  | Всего | 44 | 108 | 101,6 | 70,4 |

Для удобства проведения дальнейших расчётов сгруппируем по комплексам работ данные, приведённые в таблице 2. Полученные результат представлен в таблице 3.

Таблица 3 – Комплексы работ по созданию ИС

|  |  |  |  |
| --- | --- | --- | --- |
| Наименование комплекса работ | Обозначение |  | Практикант, часов |
| Создание математического обеспечения и написание программы и т.д. |  | 34 | 16,5 |
| Ввод программы, редактирование, тестирование, выполнение и т.д. |  | 11,3 | 7,6 |
| Прочие затраты по разработке ИС |  | 35,6 | 22,1 |
| Маркетинговые исследования |  | 9,5 | 8,5 |
| Оформление программного продукта |  | 11,2 | 5,5 |
| Всего |  | 101,6 | 60,2 |

Данные, представленные в таблице 3, служат для расчёта затрат на создание ИС.

Общение затраты на создание ИС определяются по формуле:

где – затраты на разработку ИС;

– затраты на маркетинговые исследования (принимается в размере 10% от );

– затраты на оформление программного продукта и подготовка к эксплуатации (принимается в размере 15% от ).

Общий фонд оплаты труда работников, участвующих в создании ИС, определяется по формуле:

где – время участия в создании ИС;

– общий фонд оплаты труда людей, участвующих в создании ИС;

– число участников в создании ИС (в данном случае 1);

– месячный оклад;

– длительность смены (8 часов);

– среднее число рабочих дней в месяце (20 дней);

П – премия (20% от ;

– выплаты по районному коэффициенту (для г.Кирова 15% от (.

Затраты на разработку ИС рассчитываются по формуле:

где – затраты на создание математического обеспечения и написание программы;

– затраты, связанные с работой компьютера при разработке ИС;

– прочие затраты, связанные с разработкой ИС (25% от ).

Затраты на создание математического обеспечения и написание программы определяются по формуле:

,

где ЗПМО - затраты на выплату заработной платы работникам, участвующим в создании математического обеспечения и написании программы;

*ССН* – общая ставка страховых взносов (30% и ФСС НС 0,2%);

*ФОТМО* – фонд оплаты труда работников, участвующих в создании математического обеспечения и написании программы (включает в себя зарплату, премию и выплаты по районному коэффициенту – 15%);

*НР* – накладные расходы организации, где разрабатывается ПО (затраты на отопление, освещение, на содержание административно-управленческого персонала и др.; принимается в размере 100% от *ЗПМО*).

Затраты на заработную плату работников, участвующих в создании математического обеспечения и написании программы, определяются по формуле:

,

где *ЗПП* – заработная плата программистов, участвующих в создании математического обеспечения и написании программы;

*ЗПДр* – заработная плата других работников временной творческой группы, принимающих участие в разработке математического обеспечения и написании программы (не учитывается).

Затраты на зарплату программистов рассчитываются по формуле:

,

где *Р* – число работников, участвующих в разработке математического обеспечения и написании программы;

*ВМО* – время участия работника в разработке математического обеспечения и написании программы, в часах.

Заработная плата других работников временной творческой группы, принимающих участие в разработке математического обеспечения и написании программы, ЗПдр определяется по формуле:

где Р – число работников, участвующих в разработке математического обеспечения и написании программы;

Вмо – время участия работника в разработке математического обеспечения и написании программы, в часах;

Ом – месячный оклад работника в соответствии с его категорией или тарифным разрядом;

Ds – длительность смены (8 часов);

Dр – среднее число рабочих дней в месяце (21 день).

Но, в данном случае она не нужна, поэтому расчёты не проводим.

Затраты при разработке ПО, связанные с работой компьютера, рассчитываются укрупнено по формуле:

,

где *ВМ* – машинное время, необходимое для ввода программы в компьютер, трансляцию, редактирование, отладку, тестирование, корректировку и выполнение, в часах;

*СМ* – стоимость одного часа эксплуатации компьютера определенной модели.

Стоимость одного часа эксплуатации компьютера рассчитывается укрупнено по следующей формуле

,

где *ТС1* – минимальная заработная плата, в настоящее время составляет 12130 руб.;

*ТК* – повышающий коэффициент (программиста) (1,6);

*КНР* – коэффициент, учитывающий накладные и другие расходы, связанные с работой компьютера (принимается равным 3%).

По формуле заработная плата программистов, участвующих в создании математического обеспечения и написании программы, составляет:

руб.

руб.

По формуле затраты на заработную плату работников, участвующих в написании программы, равна руб. (т.к. 1 работник-практикант).

Фонд оплаты на создание математического обеспечения составляет:

5145 руб

Затраты на создание математического обеспечения и написание программы вычисляются по формуле и составляют:

руб.

Стоимость одного часа эксплуатации компьютера определяется по формуле и составляет:

руб.

Затраты, связанные с работой компьютера при разработке ПО, рассчитываются по формуле и составляют:

руб.

Затрата на разработку ИС определяется по формуле и составляет:

руб.

Общий фонд оплаты труда работников, участвующих в создании программного обеспечения, определяется по формуле и составляет:

руб.

Общие затраты на создание информационной системы вычисляются по формуле и составляют:

руб.

Вывод: учитывая среднюю стоимость информационной системы, данная ИС выгодна для покупки.

# Заключение

В данной работе был проведен анализ поставленной проблемы с последовательным её решением. Был рассмотрен и усвоен предложенный теоретический материал, что позволило более качественно и быстро справиться с поставленными целями и задачами.

В процессе работы были приобретены навыки анализа производства, применены и усовершенствованы полученные знания в области разработки систем управления базами данных для решения конкретной задачи.

Разработанная база данных может существенно упростить работу сотрудникам компании, облегчить процедуру получения информации о проектах.

# СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Техническое задание [Электронный ресурс]: Режим доступа:

https://www.prj-exp.ru/patterns/pattern\_tech\_task.php#tech\_task22 Дата обращения: 12.06.2020

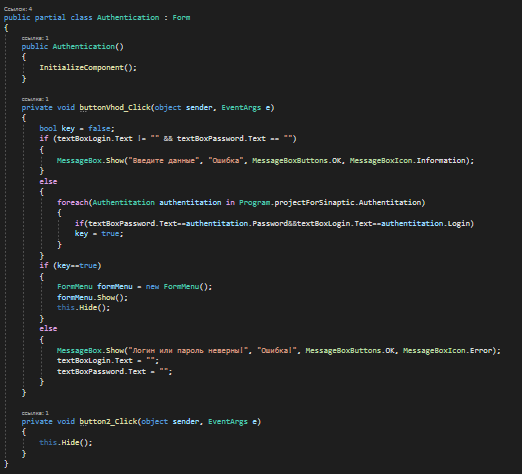
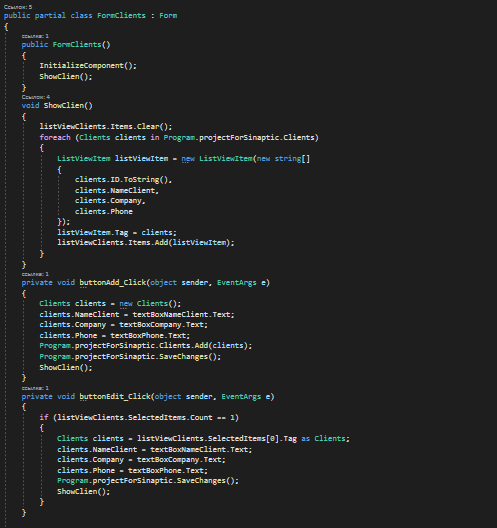
1. База данных SQL [Электронный ресурс]: Режим доступа:

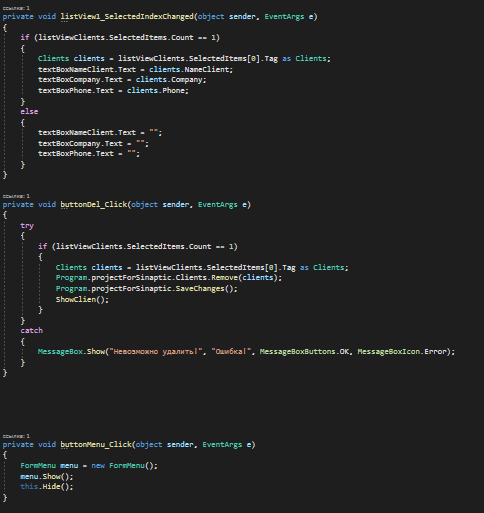
https://docs.microsoft.com/ru-ru/sql/ssms/sql-server-management-studio-ssms?view=sql-server-ver15 Дата обращения: 15.06.2020

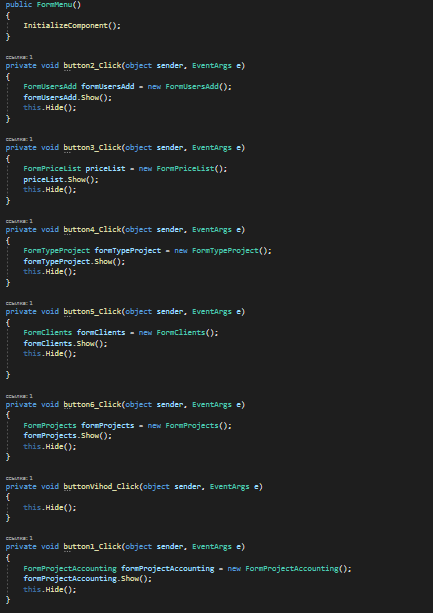
1. Обзор средств SQL [Электронный ресурс]: Режим доступа:

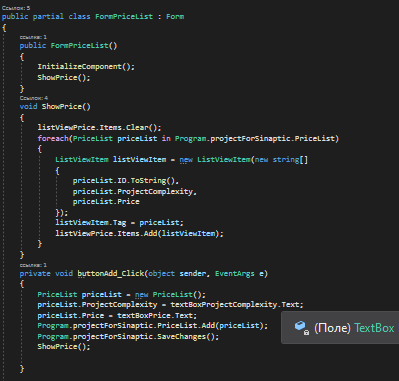
## https://docs.microsoft.com/ru-ru/sql/tools/overview-sql-tools?view=sql-server-ver15 Дата обращения: 16.06.2020

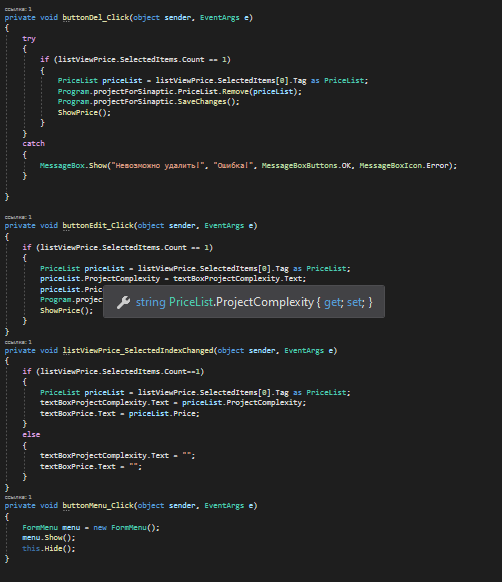
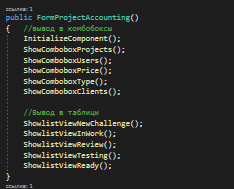
1. Руководство по языку программирования С# [Электронный ресурс]: Режим доступа: <https://metanit.com/sharp/tutorial/> Дата обращения: 18.06.2020
2. Не доделал

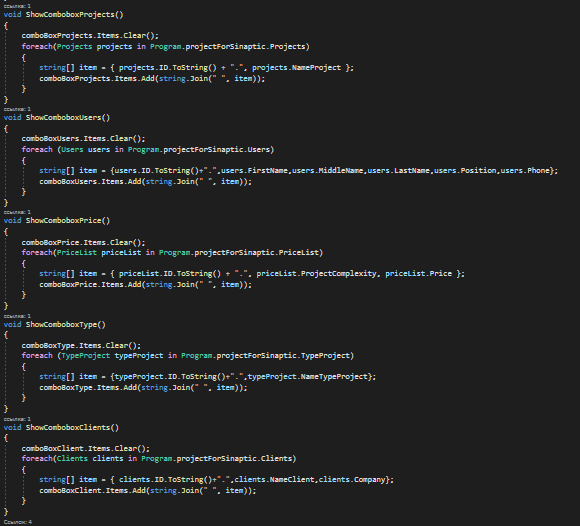
**ПРИЛОЖЕНИЕ А** 









void ShowlistViewNewChallenge()

{

listViewNewChallenge.Items.Clear();

foreach (ProjectAccounting projectAccounting in Program.projectForSinaptic.ProjectAccounting)

{

if (projectAccounting.Status == "New")

{

ListViewItem item = new ListViewItem(new string[]

{

projectAccounting.ID.ToString(),

projectAccounting.IDProject.ToString() + ". "+projectAccounting.Projects.NameProject,

projectAccounting.IDClient.ToString()+". "+projectAccounting.Clients.NameClient+" Компания:"+projectAccounting.Clients.Company+" Телефон:"+projectAccounting.Clients.Phone,

projectAccounting.IDType.ToString()+". "+projectAccounting.TypeProject.NameTypeProject,

projectAccounting.IDUser.ToString()+". "+projectAccounting.Users.FirstName+" "+ projectAccounting.Users.MiddleName+" "+projectAccounting.Users.LastName+" ("+projectAccounting.Users.Position+") "+projectAccounting.Users.Phone,

projectAccounting.IDPrice.ToString()+". "+projectAccounting.PriceList.Price

});

item.Tag = projectAccounting;

listViewNewChallenge.Items.Add(item);

}

}

listViewNewChallenge.AutoResizeColumns(ColumnHeaderAutoResizeStyle.HeaderSize);

}

void ShowlistViewInWork()

{

listViewInWork.Items.Clear();

foreach (ProjectAccounting projectAccounting in Program.projectForSinaptic.ProjectAccounting)

{

if (projectAccounting.Status == "InWork")

{

ListViewItem item = new ListViewItem(new string[]

{

projectAccounting.ID.ToString(),

projectAccounting.IDProject.ToString() + ". "+projectAccounting.Projects.NameProject,

projectAccounting.IDClient.ToString()+". "+projectAccounting.Clients.NameClient+" Компания:"+projectAccounting.Clients.Company+" Телефон:"+projectAccounting.Clients.Phone,

projectAccounting.IDType.ToString()+". "+projectAccounting.TypeProject.NameTypeProject,

projectAccounting.IDUser.ToString()+". "+projectAccounting.Users.FirstName+" "+ projectAccounting.Users.MiddleName+" "+projectAccounting.Users.LastName+" ("+projectAccounting.Users.Position+") "+projectAccounting.Users.Phone,

projectAccounting.IDPrice.ToString()+". "+projectAccounting.PriceList.Price

});

item.Tag = projectAccounting;

listViewInWork.Items.Add(item);

}

}

listViewInWork.AutoResizeColumns(ColumnHeaderAutoResizeStyle.HeaderSize);

}

void ShowlistViewReview()

{

listViewReview.Items.Clear();

foreach (ProjectAccounting projectAccounting in Program.projectForSinaptic.ProjectAccounting)

{

if (projectAccounting.Status == "Review")

{

ListViewItem item = new ListViewItem(new string[]

{

projectAccounting.ID.ToString(),

projectAccounting.IDProject.ToString() + ". "+projectAccounting.Projects.NameProject,

projectAccounting.IDClient.ToString()+". "+projectAccounting.Clients.NameClient+" Компания:"+projectAccounting.Clients.Company+" Телефон:"+projectAccounting.Clients.Phone,

projectAccounting.IDType.ToString()+". "+projectAccounting.TypeProject.NameTypeProject,

projectAccounting.IDUser.ToString()+". "+projectAccounting.Users.FirstName+" "+ projectAccounting.Users.MiddleName+" "+projectAccounting.Users.LastName+" ("+projectAccounting.Users.Position+") "+projectAccounting.Users.Phone,

projectAccounting.IDPrice.ToString()+". "+projectAccounting.PriceList.Price

});

item.Tag = projectAccounting;

listViewReview.Items.Add(item);

}

}

listViewReview.AutoResizeColumns(ColumnHeaderAutoResizeStyle.HeaderSize);

}

void ShowlistViewTesting()

{

listViewTesting.Items.Clear();

foreach (ProjectAccounting projectAccounting in Program.projectForSinaptic.ProjectAccounting)

{

if (projectAccounting.Status == "Testing")

{

ListViewItem item = new ListViewItem(new string[]

{

projectAccounting.ID.ToString(),

projectAccounting.IDProject.ToString() + ". "+projectAccounting.Projects.NameProject,

projectAccounting.IDClient.ToString()+". "+projectAccounting.Clients.NameClient+" Компания:"+projectAccounting.Clients.Company+" Телефон:"+projectAccounting.Clients.Phone,

projectAccounting.IDType.ToString()+". "+projectAccounting.TypeProject.NameTypeProject,

projectAccounting.IDUser.ToString()+". "+projectAccounting.Users.FirstName+" "+ projectAccounting.Users.MiddleName+" "+projectAccounting.Users.LastName+" ("+projectAccounting.Users.Position+") "+projectAccounting.Users.Phone,

projectAccounting.IDPrice.ToString()+". "+projectAccounting.PriceList.Price

});

item.Tag = projectAccounting;

listViewTesting.Items.Add(item);

}

}

listViewTesting.AutoResizeColumns(ColumnHeaderAutoResizeStyle.HeaderSize);

}

void ShowlistViewReady()

{

listViewReady.Items.Clear();

foreach (ProjectAccounting projectAccounting in Program.projectForSinaptic.ProjectAccounting)

{

if (projectAccounting.Status == "Ready")

{

ListViewItem item = new ListViewItem(new string[]

{

projectAccounting.ID.ToString(),

projectAccounting.IDProject.ToString() + ". "+projectAccounting.Projects.NameProject,

projectAccounting.IDClient.ToString()+". "+projectAccounting.Clients.NameClient+" Компания:"+projectAccounting.Clients.Company+" Телефон:"+projectAccounting.Clients.Phone,

projectAccounting.IDType.ToString()+". "+projectAccounting.TypeProject.NameTypeProject,

projectAccounting.IDUser.ToString()+". "+projectAccounting.Users.FirstName+" "+ projectAccounting.Users.MiddleName+" "+projectAccounting.Users.LastName+" ("+projectAccounting.Users.Position+") "+projectAccounting.Users.Phone,

projectAccounting.IDPrice.ToString()+". "+projectAccounting.PriceList.Price

});

item.Tag = projectAccounting;

listViewReady.Items.Add(item);

}

}

listViewReady.AutoResizeColumns(ColumnHeaderAutoResizeStyle.HeaderSize);

}

