СОДЕРЖАНИЕ

[Введение 5](#__RefHeading___Toc518_228746720)

[1 Анализ предметной области 6](#__RefHeading___Toc520_228746720)

[2 Постановка задачи 7](#__RefHeading___Toc522_228746720)

[3 Проектирование программного модуля 8](#__RefHeading___Toc524_228746720)

[3.1 Инфологическая модель 8](#__RefHeading___Toc526_228746720)

[3.2 Описание входных и выходных данных 9](#__RefHeading___Toc528_228746720)

[4 Технологии и инструменты разработки 10](#__RefHeading___Toc530_228746720)

[4.1 Технологии для разработки 10](#__RefHeading___Toc532_228746720)

[4.2 Инструменты разработки 11](#__RefHeading___Toc534_228746720)

[5 Разработка программного модуля 12](#__RefHeading___Toc536_228746720)

[5.1 Пользовательский интерфейс программного модуля 12](#__RefHeading___Toc538_228746720)

[5.2 Описание методов разработки 14](#__RefHeading___Toc540_228746720)

[Заключение 15](#__RefHeading___Toc542_228746720)

[Список использованных источников 16](#__RefHeading___Toc544_228746720)

Table of Contents

[Введение 4](#__RefHeading___Toc518_228746720)

[1 Анализ предметной области 5](#__RefHeading___Toc520_228746720)

[2 Постановка задачи 6](#__RefHeading___Toc522_228746720)

[3 Проектирование программного модуля 7](#__RefHeading___Toc524_228746720)

[3.1 Инфологическая модель 7](#__RefHeading___Toc526_228746720)

[3.2 Описание входных и выходных данных 8](#__RefHeading___Toc528_228746720)

[4 Технологии и инструменты разработки 9](#__RefHeading___Toc530_228746720)

[4.1 Технологии для разработки 9](#__RefHeading___Toc532_228746720)

[4.2 Инструменты разработки 10](#__RefHeading___Toc534_228746720)

[5 Разработка программного модуля 11](#__RefHeading___Toc536_228746720)

[5.1 Пользовательский интерфейс программного модуля 11](#__RefHeading___Toc538_228746720)

[5.2 Описание методов разработки 13](#__RefHeading___Toc540_228746720)

[Заключение 14](#__RefHeading___Toc542_228746720)

[Список использованных источников 15](#__RefHeading___Toc544_228746720)

СОДЕРЖАНИЕ

Введение 3

1 Анализ предметной области 4

2 Постановка задачи 5

3 Проектирование программного модуля 6

3.1 Инфологическая модель……………………………………………………………6

3.2 Описание входных и выходных данных………………………………………….7

4 Технологии и инструменты разработки 8

4.1 Технологии для разработки 8

4.2 Инструменты разработки 9

5 Разработка программного модуля 10

5.1 Пользовательский интерфейс программного модуля 10

5.2 Описание методов разработки 13

Заключение 14

Список использованных источников 15

# Введение

Объектом исследования данной работы является учет на предприятии ООО «Консалтинг».

Для основной деятельности предприятия необходимо иметь четкую структурированную систему работы, позволяющую задействовать все подразделения, а также осуществление контроля над этими подразделениями. Поэтому нужно создать единую систему, обеспечивающую доступ к данным всех отделов, инвентаря, должностей и сотрудников предприятия. Это упростит работу с информацией.

# 1 Анализ предметной области

ООО «Консалтинг» является предриятием по производству древесной продукции. Свою деятельность предприятие осуществляет с 1992 года.

Сегодня ООО «Консалтинг» является членом Ассоциации мебельной и деревообрабатывающей промышленности России, Ассоциации мебельщиков Южного региона, Ростовской ТПП.

Основные виды деятельности предприятия: производство древесностружечных плит, облицовывание ДСтП, МДФ, методом ламинирования, выпуск мебели в ассортименте, изготовление столярных изделий из ценных и хвойных пород дерева.

Данное заведение находится по адресу ул. 30 лет победы 14-9

# 2 Постановка задачи

Цель данного проекта – разработка программного модуля обеспечивающего использование структурированной базы данных для автоматизации деятельности и обеспечения хранения, накопления, предоставления, редактирования учета предприятия. Так, например, данная программа должна иметь возможность добавления, удаления информации о сотрудниках, отделах, инвентаре и должностях.

Программный модуль должен выполнять следующие требования:

⦁ позволять просматривать таблицы;

⦁ позволять вносить и удалять новые данные;

⦁ автоматизированная работа с несколькими базами данных

⦁ иметь интуитивно понятный интерфейс.

Программный модуль должен быть реализован с помощью языка C#, графический интерфейс пользователя с помощью WPF фреймворка, являющийся частью платформы .NET. Графический интерфейс пользователя должен быть простой и понятный, сама программа не должна использовать для освоения персоналом специализированное образование или прохождение курсов.

# 3 Проектирование программного модуля

## 3.1 Инфологическая модель

Инфологическая модель – это набор сущностей и связей между ними. Она содержит информацию о сущностях системы и способах их взаимодействия, включает идентификацию объектов, важных для предметной области (сущностей), свойств этих объектов (атрибутов) и их отношений с другими объектами (связей).

Для выполнения проекта была создана база данных, инфологическая модель которой представлена на рисунке 3.1.

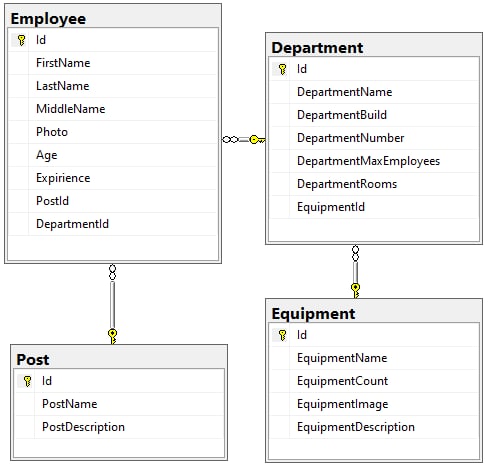


Рисунок 3.1 – Инфологическая модель базы данных

## 3.2 Описание входных и выходных данных

Входная информация – это информация, представляющая собой внесенные/измененные/удаленые или другие возможности автоматизированной системы над данными о хранящихся отделах или снаряжении на предприятии находящиеся в базе данных.

Выходная информация – это информация, получаемая в результате внесения/изменения/удаления данных в базе данных через автоматизированную систему.

# 4 Технологии и инструменты разработки

## 4.1 Технологии для разработки

C# (произносится как "си шарп") — 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# [6].

C# — объектно-ориентированный, ориентированный на компоненты язык программирования. C# предоставляет языковые конструкции для непосредственной поддержки такой концепции работы. Благодаря этому C# подходит для создания и применения программных компонентов. 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 [2].

C# поддерживает управление версиями , чтобы обеспечить совместимость программ и библиотек с течением времени. Вопросы управления версиями существенно повлияли на такие аспекты разработки C#, как раздельные модификаторы virtual и override, правила разрешения перегрузки методов и поддержка явного объявления членов интерфейса [1].

Для организации баз данных MS SQL Server использует реляционную модель. Эта модель баз данных была разработана еще в 1970 году Эдгаром Коддом. А на сегодняшний день она фактически является стандартом для организации баз данных [7].

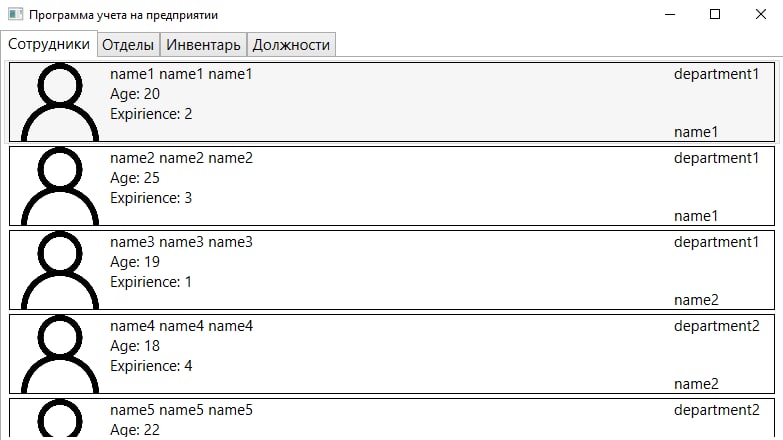
## 4.2 Инструменты разработки

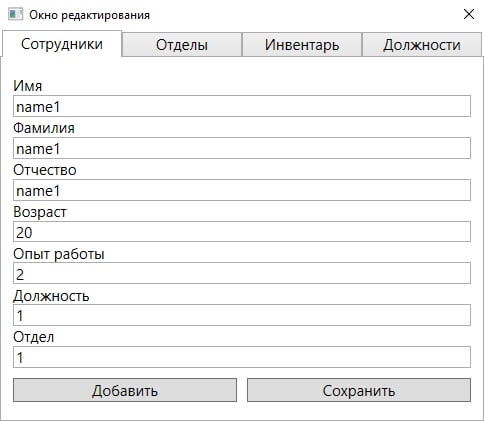
В качестве среды для разработки приложения была выбрана программа Microsoft Visual Studio, линейка продуктов компании Microsoft, включающих интегрированную среду разработки программного обеспечения и ряд других инструментов. Данные продукты позволяют разрабатывать как консольные приложения, так и игры и приложения с графическим интерфейсом, в том числе с поддержкой технологии Windows Forms, а также веб-сайты, веб-приложения, веб-службы как в родном, так и в управляемом кодах для всех платформ, поддерживаемых Windows, Windows Mobile, Windows CE, .NET Framework, Xbox, Windows Phone .NET Compact Framework и Silverlight. Существует множество сред разработки, но данная была выбрана в ввиду ее удобного графического интерфейса и средств отладки. Microsoft Visual Studio включает в себя редактор исходного кода с поддержкой технологии IntelliSense и возможностью простейшего рефакторинга кода. Встроенный отладчик может работать как отладчик уровня исходного кода, так и отладчик машинного уровня. Остальные встраиваемые инструменты включают в себя редактор форм для упрощения создания графического интерфейса приложения, веб-редактор, дизайнер классов и дизайнер схемы базы данных. Microsoft Visual Studio позволяет создавать и подключать сторонние дополнения (плагины) для расширения функциональности практически на каждом уровне, включая добавление поддержки систем контроля версий исходного кода (как, например, Subversion и Visual SourceSafe), добавление новых наборов инструментов (например, для редактирования и визуального проектирования кода на предметно-ориентированных языках программирования) или инструментов для прочих аспектов процесса разработки программного обеспечения (например, клиент Team Explorer для работы с Team Foundation Server) [4].

# 5 Разработка программного модуля

## 5.1 Пользовательский интерфейс программного модуля

При открытии программы пользователь взаимодействует с основным окном, в котором есть 4 панели: Сотрудники, Отделы, Инвентарь и Должности. При выборе сотрудника открывается окно редактирования, при необходимости внесения информации в зависимости от таблицы можно добавить сотрудника, отдел, инвентарь или должность. Они представлены на рисунках 5.1 – 5.2 соответственно.

Рисунок 5.1 – Панель «Сотрудники»

Рисунок 5.2 – Окно редактирования «Сотрудников»

## 5.2 Описание методов разработки

Для примера взята часть кода для добавления сотрудника в таблице из базы данных через приложение. Перед нажатием на кнопку добавления предмета надо заполнить данные в поля на форме.

private void ButtonAddEmployee\_Click(object sender, RoutedEventArgs e)

{

Employee employee = new Employee();

if (TB\_FirstName.Text != "" || TB\_LastName.Text != "" || TB\_MiddleName.Text != "" || TB\_Age.Text != "" || TB\_Expirience.Text != "" || TB\_PostId.Text != "" || TB\_DepartmentId.Text != "")

{

employee.FirstName = TB\_FirstName.Text;

employee.LastName = TB\_LastName.Text;

employee.MiddleName = TB\_MiddleName.Text;

employee.Age = Convert.ToInt32(TB\_Age.Text);

employee.Expirience = Convert.ToInt32(TB\_Expirience.Text);

employee.PostId = Convert.ToInt32(TB\_PostId.Text);

employee.DepartmentId = Convert.ToInt32(TB\_DepartmentId.Text);

}

else

{

MessageBox.Show("Не все данные введены");

return;

}

using(ApplicationDbContext context = new ApplicationDbContext())

{

context.Employees.Add(employee);

context.SaveChanges();

}

}

# Заключение

Разработанная программа учета для ООО «Консалтинг» представляет собой простое и минималистичное программное обеспечение для предприятий.

Программа разработана на языке C# в интегрированной среде разработки Microsoft Visual Studio с использованием СУБД SQL Server. Приложение интуитивно понятно даже неопытному пользователю и не затрудняет его работу с базами данных.

В дальнейшем в программу будут добавляться новые возможности и функции, необходимые для расширения возможностей работы с базой данных и упрощением пользовательского интерфейса.

В процессе написания кода для данной программы были приобретены практические навыки в работе в СУБД SQL Server и языке программирования C#.

Цель курсового проекта достигнута. Задачи выполнены.

# Список использованных источников

Интернет ресурсы:

1. Рихтер, Д. CLR via C#. Программирование на платформе Microsoft .NET Framework 4.5 на языке C# / Д. Рихтер. – Санкт-Петербург : Издательство Питер, 2019. – 896 с.
2. Фримен, Э. Head First. Паттерны проектирования. / Э. Фримен, Э. Робсон, К. Сьерра. – Санкт-Петербург : Питер, 2021. – 640 с.
3. Документация по Visual Studio [Электронный ресурс] – Режим доступа: https://docs.microsoft.com/ru-RU/visualstudio/windows/?view=vs-2022&preserve-view=true
4. Что такое Windows Presentation Foundation (WPF)? [Электронный ресурс] – Режим доступа: https://docs.microsoft.com/ru-ru/visualstudio/designers/getting-started-with-wpf?view=vs-2022
5. C Sharp – Википедия [Электронный ресурс] – Режим доступа: https://ru.wikipedia.org/wiki/C\_Sharp
6. MS SQL Server и T-SQL [Электронный ресурс] – Режим доступа: https://metanit.com/sql/sqlserver/1.1.php