**Компьютерная академия «ШАГ»**

**Харьковский филиал**

**Кафедра Разработки программного обеспечения**

**КУРСОВОЙ ПРОЕКТ**

**“Менеджер достижений”**

**Студента группы 24ППР12  
Байрак В.И.**

**Научный руководитель:  
Задерей Ю.Н.**

**Харьков 2016**

**Содержание**

Оглавление

[Введение 3](#_Toc446749642)

[Техническое задание 4](#_Toc446749643)

[1.Выбор технологии реализации проекта 5](#_Toc446749644)

[2.Разработка графического интерфейса 6](#_Toc446749645)

[3.Разработка базы данных для системы 7](#_Toc446749646)

[4.Разработка алгоритмов функционирования системы 8](#_Toc446749647)

[5.Руководство пользователя 10](#_Toc446749648)

[Выводы 20](#_Toc446749649)

[Список использованной литературы 21](#_Toc446749650)

## Введение

Сегодня трудно себе представить современный мир информационных технологий без использования баз данных и информационных систем. Практически все системы в той или иной степени связаны с функциями долгосрочного хранения и обработки информации. Фактически информация становится фактором, определяющим эффективность любой сферы деятельности. В связи с этим все большую актуальность приобретает выбор и применение систем управления базами данных, например, такой как Microsoft SQL Server.

Microsoft SQL Server — система управления реляционными базами данных (СУБД), разработанная корпорацией Microsoft. Основной используемый язык запросов — Transact-SQL, создан совместно Microsoft и Sybase. Transact-SQL является реализацией стандарта ANSI/ISO по структурированному языку запросов (SQL) с расширениями. Используется для работы с небольшими и средними по размеру базами данных до крупных баз данных масштаба предприятия; конкурирует с другими СУБД в этом сегменте рынка.

В данной курсовой работе была разработана база данных для предметной области “**Менеджер достижений**”. Предложенная база данных помогает человеку сохранять и анализировать свои достижения за любой период времени и по различным темам.

Для разработки приложения и базы данных была использована технология .NET Framework от компании Microsoft, которая на сегодняшний день является самым популярным разработчиком операционных систем начиная от домашнего пользователя до обслуживания серверов огромной мощности.

## Техническое задание

При разработке курсового проект были решены такие задачи:

1. Добавление нового достижения:
   1. Добавление всей необходимой информации
   2. Изменении информации
2. Хранение информации о пользователях
   1. Добавление информации пользователя
   2. Хранение информации пользователя
3. Анализ достижений
   1. Фильтрация достижений по датам и категориям
   2. Рейтинг по темам
4. Управление категориями
   1. Отображение зависимостей тема-подтема
   2. Детальная информация по теме/подтеме
   3. Изменение тем-подтем
   4. Удаление подтем
5. Трей
   1. Добавление темы из трея

Также был разработан пользовательский графический интерфейс с интуитивным пониманием функционала для облегчения и увлечения пользователей, чтобы не напрягать пользователей обилием отвлекающих и утомляющих факторов.

Требования к программным характеристикам операционной системы минимальны, и не требуют больших ресурсов, что положительно скажется на закупке необходимого оборудования для работы.

## 1.Выбор технологии реализации проекта

Для разработки приложения была выбрана технология .NET Framework от корпорации Microsoft - одна из крупнейших [транснациональных компаний](https://ru.wikipedia.org/wiki/%D0%A2%D1%80%D0%B0%D0%BD%D1%81%D0%BD%D0%B0%D1%86%D0%B8%D0%BE%D0%BD%D0%B0%D0%BB%D1%8C%D0%BD%D0%B0%D1%8F_%D0%BA%D0%BE%D0%BC%D0%BF%D0%B0%D0%BD%D0%B8%D1%8F) по производству [проприетарного](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D0%BF%D1%80%D0%B8%D0%B5%D1%82%D0%B0%D1%80%D0%BD%D0%BE%D0%B5_%D0%9F%D0%9E" \o "Проприетарное ПО) [программного обеспечения](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%BD%D0%BE%D0%B5_%D0%BE%D0%B1%D0%B5%D1%81%D0%BF%D0%B5%D1%87%D0%B5%D0%BD%D0%B8%D0%B5) для различного рода вычислительной техники,  разработчик наиболее широко распространённой на данный момент в мире программной платформы — семейства [операционных систем](https://ru.wikipedia.org/wiki/%D0%9E%D0%BF%D0%B5%D1%80%D0%B0%D1%86%D0%B8%D0%BE%D0%BD%D0%BD%D0%B0%D1%8F_%D1%81%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0) [Windows](https://ru.wikipedia.org/wiki/Microsoft_Windows" \o "Microsoft Windows).

Язык для написания приложения, который бы соответствовал выбранной идеологии .NET, я выбрал C#, который позволяет написать приложение, работающее с базами данных.

C# — элегантный, типобезопасный объектно-ориентированный язык, предназначенный для разработки разнообразных безопасных и мощных приложений, выполняемых в среде .NET Framework. С помощью языка C# можно создавать обычные приложения Windows, XML-веб-службы, распределенные компоненты, приложения "клиент-сервер", приложения баз данных и т. д. Visual C# предоставляет развитый редактор кода, конструкторы с удобным пользовательским интерфейсом, встроенный отладчик и множество других средств, упрощающих разработку приложений на базе языка C# и .NET Framework.

Для хранения данных используются различные системы управления базами данных: MS SQL Server, Oracle, MySQL и так далее. И большинство крупных приложений, так или иначе, используют для хранения данных эти системы управления базами данных. Однако, чтобы осуществлять связь между базой данных и приложением на C# необходим посредник. И именно таким посредником является технология ADO.NET.

## 2.Разработка графического интерфейса

При реализации графической части приложения .NET нам предлагает несколько API-интерфейсов, основные из них Windows Forms (WinForms) и Windows Presentation Foundation (WPF).

**WPF** представляет собой обширный API-интерфейс для создания настольных графических программ, имеющих насыщенный дизайн и интерактивность. В отличие от устаревшей технологии WinForms, WPF включает новую модель построения пользовательских приложений (в основе WPF лежит мощная инфраструктура, основанная на DirectX). Это означает возможность применения развитых графических эффектов, не платя за это производительностью, как это было в WinForms. Фактически даже становятся доступными такие расширенные средства, как поддержка видеофайлов и трехмерное содержимое. Используя эти средства (при наличии хорошего инструмента графического дизайна), можно создавать бросающиеся в глаза пользовательские интерфейсы и визуальные эффекты, которые были просто невозможны в WinForms. В добавок ко всем положительным сторонам WPF относится ещё одно очень важное качество: привязки к размерам и координатам элементов отсутствует, то есть окно и все элементы внутри него измеряются в независимых от устройства единицах. Такая единица определена как 1/96 дюйма. Применяется система DPI (Dots Per Inch), в зависимости от системной установки разрешения монитора. Что несет положительный характер при запуске программы на компьютерах с разными разрешениями мониторов. Поэтому я остановил свой выбор на WPF.

## 3.Разработка базы данных для системы

При разработке базы данных возник вопрос: каким способом её реализовывать?

1. Воспользоваться более классическим и проверенным способом создания базы данных с помощью редактора, всё той же компании Microsoft, MS SQL Server. Написании вручную скрипта создания таблиц, ключей и всех необходимых частей базы данных.
2. Воспользоваться сравнительно новой технологией Entity Framework, которая предоставляет возможность создания базы данных через объектно-ориентированный код C# одним из трёх подходов:
   1. Database First – подход для проектировщиков баз данных, формирование модели на основе готовой базы
   2. Model First – подход для архитекторов, формирование базы с модели в Visual Studio
   3. Code First – подход для программистов.

Я остановил свой выбор на технологии Entity Framework и выбрал подход Code First – при данном подходе модель EDMX вообще не используется и классы (будущие таблицы) настраиваются вручную. База данных генерируется из созданной вручную модели объектов (классов) C#.

При использовании технологии Entity Framework, любой из её подходов, разработчики приложений затрачивают меньше работы/часов на разработку приложения, что ускоряет релиз готового продукта. Упрощается понимание кода, для дальнейшего его улучшения. Делает код менее ошибка-опасным, так как всю работу с данными в приложении и в базе данных берёт на себя Entity Framework.

Для хранения данных был выбран MS SQL Server, который предоставляет хорошую надёжность, производительность и безопасность данных. В добавок ко всем положительным сторонам система SQL Server разработана Microsoft, а значит адаптировано и совместимо под остальные продукты этой корпорации, которые используются при реализации приложения.

Структура базы данных со всеми ее таблицами представлена на рисунке 3.1.

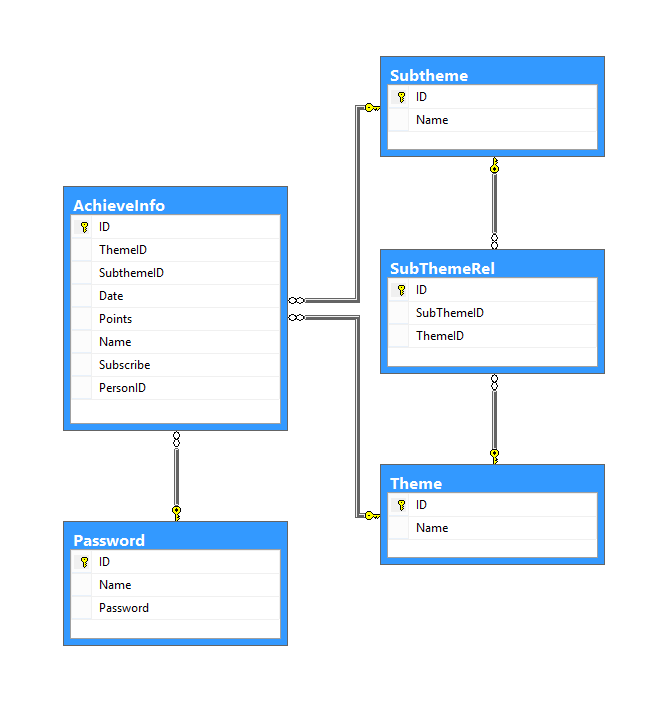
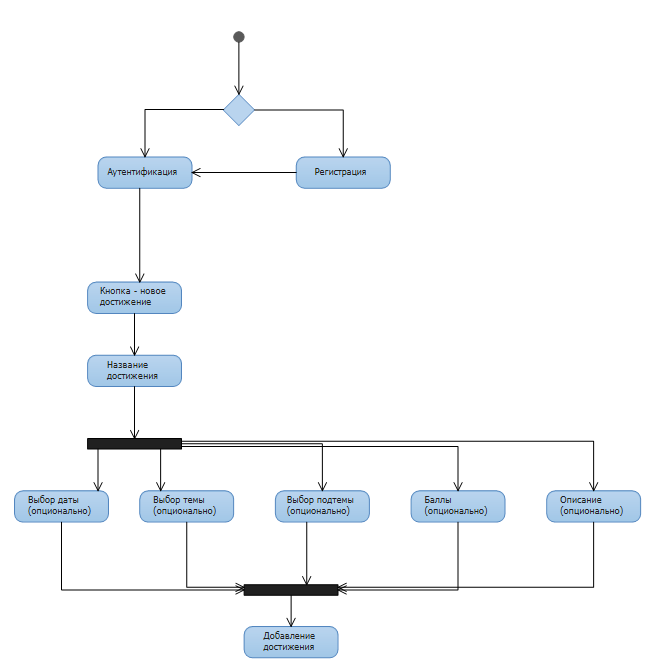
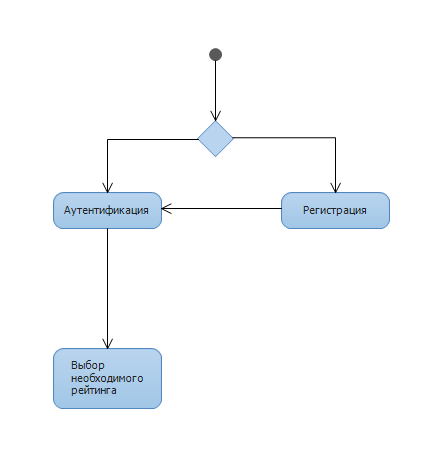


Рисунок 3,1 Структура Базы данных

## 4.Разработка алгоритмов функционирования системы



Блок схема 4.1. Первичное добавление достижений



Блок схема 4.2. Получение данных статистики

## 5.Руководство пользователя

**Окно Авторизации**

При запуске программа начинается с окна **Авторизации пользователя** (Рис. 5.1), где необходимо:

1. Ввод логина
2. Ввод пароля

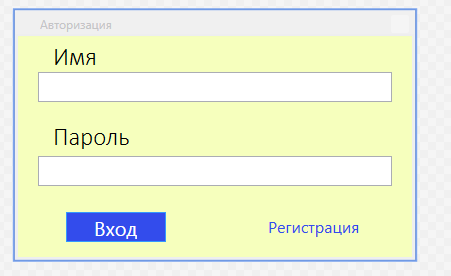


Рисунок 5.1 Авторизация пользователя

Также есть возможность зарегистрировать нового пользователя

Она аналогична окну Авторизации, но предлагает ввести пароль дважды.

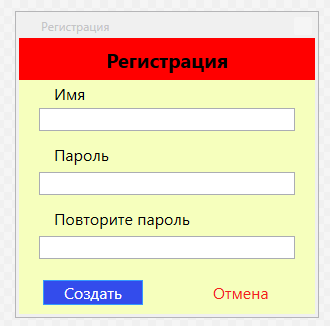


Рисунок 5.2 Регистрация пользователя

При входе в программу, пользователь видит главное окно, в котором отображена большая кнопка «добавить» при нажатии на которую пользователь добавляет новое достижение, а также справочные материалы и аналитика в «Справочник» и «Статистика», информацию по пользователю и разработчику в «Общая информация», либо общие операции в «Файл» (Сверху страницы).

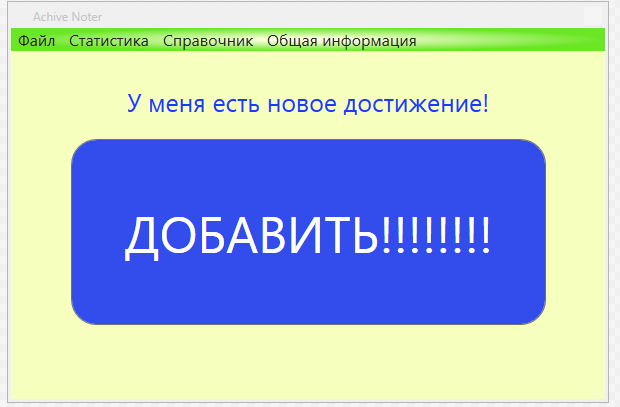


Рисунок 5.3 Главное окно

**Добавление достижения**

При нажатии кнопки добавить пользователю предлагается ввести имя достижения.

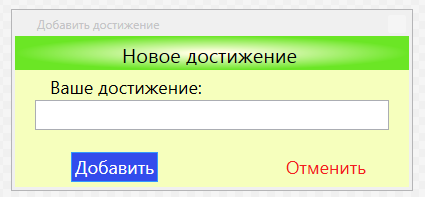


Рисунок 5.4 Название достижения

При его вводе он переходит на окно детализации информации по достижению.

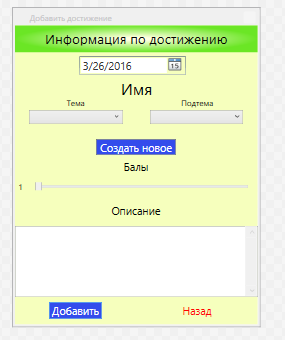


Рисунок 5.5 Детальная информация по достижению

Также он может изменить добавить новое достижение прямо из этого окна (детальнее в разделе «Работа с достижениями»).

При нажатии кнопки готово его достижение добавляется в бд.

**Статистика**

При выборе пункта «статистика» пользователь получает выбор из двух пунктов - «общая статистика» и «рейтинг тем». Оба этих пункта предоставляют пользователю возможность просмотра как информации за сегодня, так и за весь период.

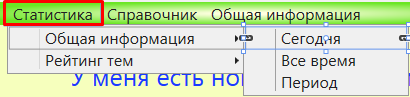


Рисунок 5.6 Пункт «Статистика»

При выборе пункта «Общая статистика» пользователь получает окно, в котором отображаться все темы, соответствующие данному критерию (вначале – временной период) и три характеристики: тема, подтема и период (если выбрано сегодня - отсутствует). Для применения фильтрации необходимо нажать «применить», для сброса – «сбросить».

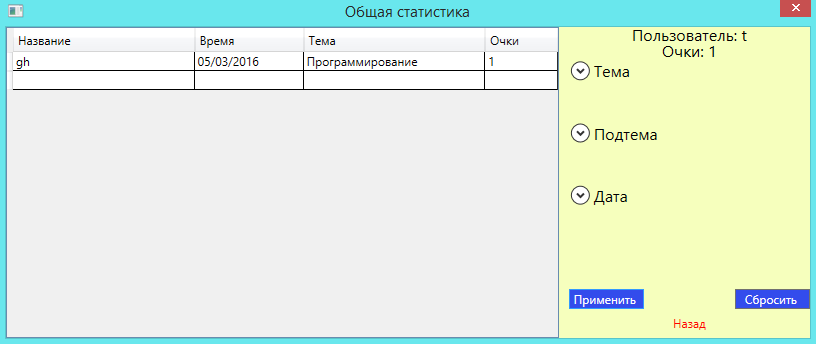


Рисунок 5.7 Окно «Общая статистика»

Рейтинг тем – показывает рейтинг тем для конкретного пользователя (исходя из баллов по данной теме) и так же имеет возможность фильтрации (по времени). Для применения фильтрации необходимо нажать «применить», для сброса – «сбросить».

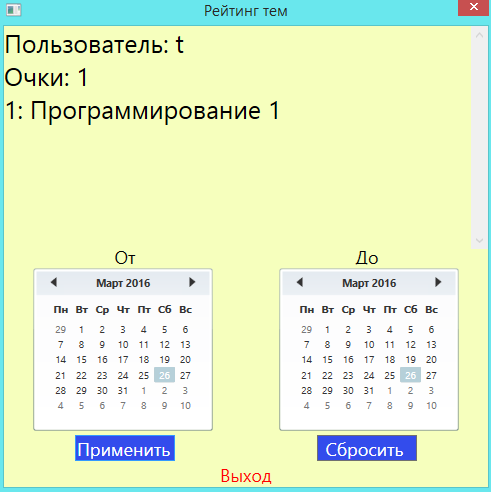


Рисунок 5.8 Окно «Рейтинг тем»

**Справочник**

При выборе пункта «Справочник» пользователь получает выбор из двух пунктов - «Общий» и «Детально». Второй дает возможность выбора информации по теме или подтеме.

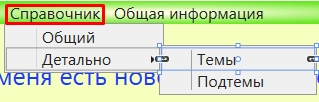


Рисунок 5.9 Пункт «Справочник»

При выборе пункта «Общий» пользователь получает окно, в котором отображаться все темы (слева) и относящиеся к ним подтемы (справа)

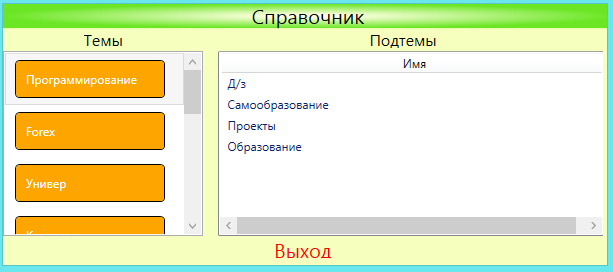


Рисунок 5.10 Окно «Общий»

«Детально» дает возможность получения списка всех тем или подтем.

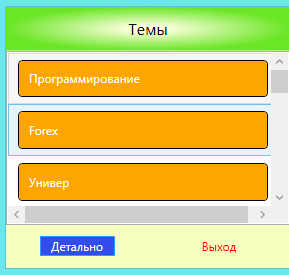


Рисунок 5.10 Окно «Детально»

При нажатии кнопки детально пользователь получает полную информацию по теме – название, общее количество очков, все относящиеся к ней достижения и имеем возможность изменить данную тему/подтему (детальнее в разделе «Работа с достижениями»).

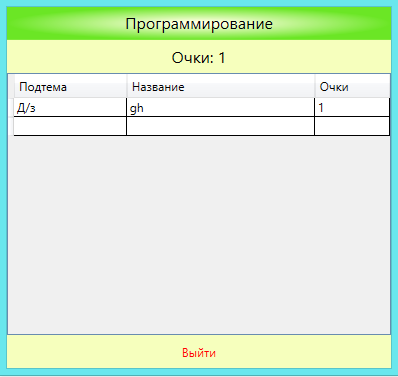


Рисунок 5.11 Полная информация по теме/подтеме

**Общая информация**

При выборе пункта «Общая информация» пользователь получает выбор из двух пунктов – «Разработчик» и «пользователь».

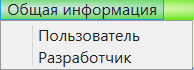


Рисунок 5.12 Пункт «Общая информация»

При выборе пункта «Разработчик» пользователь получает окно, в котором отображаться информация по разработчику

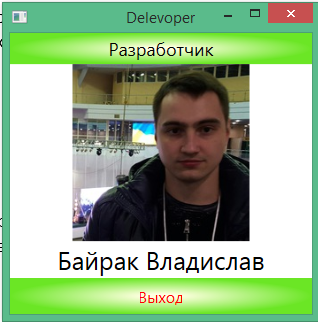


Рисунок 5.13 Окно «Разработчик»

При выборе пункта «Пользователь» пользователь получает окно, в котором отображаться информация по пользователю. В неё входят все очки пользователя, лучшие и худший навыки (за месяц и за всё время).

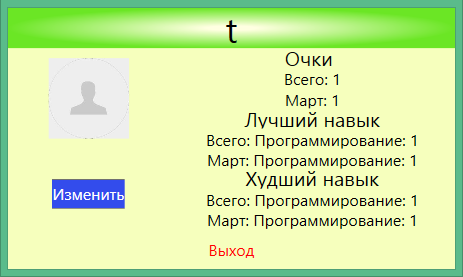


Рисунок 5.14 Окно «Пользователь»

Также возможна изменение информации по пользователю – кнопка «изменить». В ней сперва вводиться старый пароль пользователя

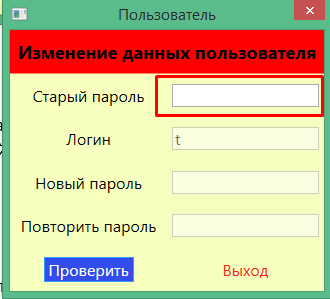


Рисунок 5.15 Ввод старого пароля

А после добавляться новый логин (опционально) и новый пароль.

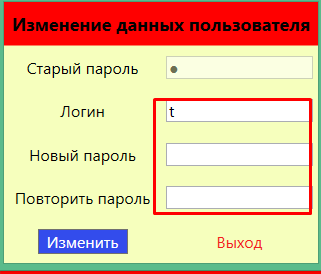


Рисунок 5.16 Ввод новых данных

**Общие операции**

В пункте «файл» пользователю предоставляется выбор нескольких окон: Добавить, изменить и выход. Выход служит для окончательного закрытия программы (без запуска трея – детальнее в разделе «Трей»), а добавить, изменить удалить для работы с категориями (детальнее в разделе «Работа с достижениями»).

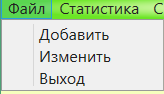


Рисунок 5.17 Пункт «Файл»

**Работа с достижениями**

При выборе пункта добавить пользователь получает окно выбора – тема или подтема.



Рисунок 5.18 Окно выбора

При выборе тема пользователь должен ввести название темы. При этом автоматически добавляется подтема с аналогичным названием

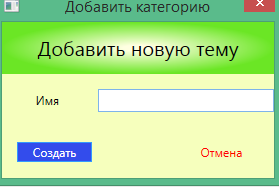


Рисунок 5.19 Новая тема

При выборе подтемы пользователь выбирает пользователь выбирает основную тему пи вводит название подтемы

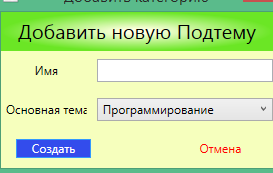


Рисунок 5.20 Новая подтема

При выборе пункта «Изменить» пользователь получает окно выбора – тема или подтема.

Независимо от выбора пользователь получает список категорий слева и текстовое окно с названием категории справа и кнопка изменить и удалить(для подтем). Общую подтему (схожее с темой название) изменить или удалить невозможно!

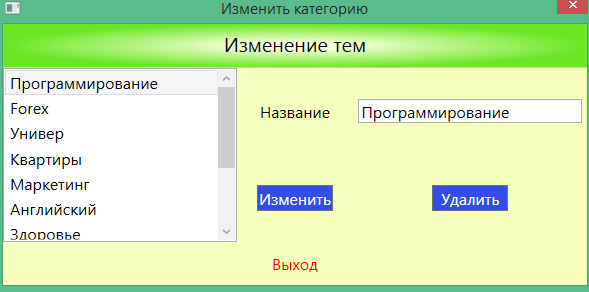


Рисунок 5.21 Изменение информации по подтеме

**Трей**

При закрытии программы не через пункт «Выход» пользователь поучает окно «Трей» в панеле задач

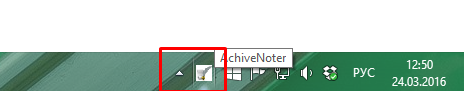


Рисунок 5.22 трей в панеле задач

При нажатии на него левой кнопкой мыши открывается главное окно программы

При нажатии на него правой кнопкой мыши открывается меню трея

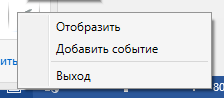


Рисунок 5.23 меню трея

В нем есть возможность отобразить главное окно, добавить достижение, либо окончально выйти из программы

## Выводы

В данной курсовой работе было разработано приложения для добавления и хранения достижений. Для реализации данного задания были применены такие технологии .NET Framework как:

1. Язык программирования C# - написание кода приложения
2. API-интерфейс Windows Presentation Foundation - написание графического интерфейса приложения
3. ADO.NET – технология для работы с базами данных, в частности её элемент Entity Framework
4. MS SQL Server – система для хранения информации, базы данных

В процессе самостоятельной работы над проектом были углублены знания по написании графического интерфейса приложения на технологии WPF, выявлены её положительные стороны.

Также были получены новые знания по технологии ADO.NET её части Entity Framework – подход Code First. Получен опыт работы с наследником класса DbContext, который по сути в себе содержит классы-объекты с помощью которых можно обращаться к данным находящимся на SQL Server.

Увеличена скорость написания кода на языке C#, а также улучшены навыки по написанию читабельного кода.

Были закреплены практические знания, умения и навыки в соответствии с содержанием дисциплины на основе теоретических сведений, полученных в процессе обучения.

Был получен важнейший навык самостоятельной работы, формулировка суждений и выводов, разбивание сложных задач на мелкие и решение их логически и последовательно.

## Список использованной литературы

1. Мак-Дональд Мэтью, WPF 4: Windows Presentation Foundation в .NET 4.0 с примерами на C# 2010 для профессионалов. : Пер. с англ. – М. : ООО «И.Д. Вильямс», 2011. – 1024 с.

2. Троелсен, Э. Язык программирования С# 2005 и платформа .NET 2.0 / Э. Троелсен. – 3-е издание. – Москва–Санкт-Петербург–Киев: Издательский дом «Вильямс», 2007. – 1168с.

3. Шилдт, Г. Полный справочник по С# / Г. Шилдт. – Москва: Санкт-Петербург; Киев: Издательский дом «Вильямс», 2006. – 750с.

4. Прайс, Дж. Visual C# .NET/ Дж. Прайс, М. Гандерлой - Полное руководство. – Киев: ВЕК, Санкт-Петербург: КОРОНА, 2004. – 960с.

5. Нортроп, Т. Основы разработки приложений на платформе Microsoft .NET Framework // Т. Нортроп, Ш. Уилдермьюс, Б. Райан - Учебный курс Microsoft- Москва: «Русская редакция», Санкт-Петербург: «Питер», 2007. – 864с.

6. <https://msdn.microsoft.com>

7. <http://professorweb.ru/>