Частное учреждение образования

«Колледж бизнеса и права»

ПРОГРАММНОЕ СРЕДСТВО ДЛЯ ПОДСЧЁТА ПРОМИЛЛЕ «АЛКОГОЛЬНЫЙ КАЛЬКУЛЯТОР»

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

КП Т.991020.401

Руководитель проекта (В.Ю.Михалевич)

Учащегося (М.В.Суконнов )

2022

Содержание

Изм.

Изм.

Изм.

Изм.

Лист

Лист

Лист

Лист

№ докум.

№ докум.

№ докум.

№ докум.

Подпись

Подпись

Подпись

Подпись

Дата

Дата

Дата

Дата

Лист

Лист

Лист

Лист

3

3

3

3

КП Т.991020.401 ПЗ

КП Т.716007.401 ПЗ

КП Т.716009.401 ПЗ

КП Т.716007.401 ПЗ

Разраб.

Разраб.

Разраб.

Разраб.

*Суконнов М.В*

*Голубев И.В.*

*Гринь А.М.*

*Голубев И.В.*

Провер.

Провер.

Провер.

Провер.

Михалевич В.Ю.

Банцевич. С.В

Банцевич. С.В

Банцевич. С.В

Т. контр.

Т. контр.

Т. контр.

Т. контр.

Н. контр.

Н. контр.

Н. контр.

Н. контр.

Утверд.

Утверд.

Утверд.

Утверд.

Программное средство для подсчёта промилле «Алкогольный калькулятор»

Лит.

У

Лит.

У

Лит.

У

Лит.

У

Листов

Листов

Листов

Листов

43

50

55

50

КБП

КБП

КБП

КБП

[Введение 4](#_Toc106188032)

[1 Объектно-ориентированный анализ и проектирование систем 5](#_Toc106188033)

[1.1 Сущность задачи 5](#_Toc106188034)

[1.2 Проектирование модели 5](#_Toc106188035)

[2 Вычислительная система 9](#_Toc106188036)

[2.1 Требования к аппаратным и операционным ресурсам 9](#_Toc106188037)

[2.2 Инструменты разработки 9](#_Toc106188038)

[3 Проектирование задачи 11](#_Toc106188039)

[3.1 Требования к приложению 11](#_Toc106188040)

[3.2 Концептуальный прототип 11](#_Toc106188041)

[3.3 Организация данных 13](#_Toc106188042)

[3.4 Функции: логическая и физическая организация 15](#_Toc106188043)

[3.5 Проектирование справочной системы приложения 21](#_Toc106188044)

[4 Описание программного средства 22](#_Toc106188045)

[4.1 Общие сведения 22](#_Toc106188046)

[4.2 Функциональное назначение 22](#_Toc106188047)

[4.3 Входные и выходные данные 23](#_Toc106188048)

[5 Методика испытаний 24](#_Toc106188049)

[5.1 Технические требования 24](#_Toc106188050)

[5.2 Функциональное тестирование 24](#_Toc106188051)

[6 Применение 26](#_Toc106188052)

[6.1 Назначение программы 26](#_Toc106188053)

[6.2 Условия применение 26](#_Toc106188054)

[6.3 Справочная система 26](#_Toc106188055)

[Заключение 27](#_Toc106188056)

[Список информационных источников 28](#_Toc106188057)

[Приложение А Текст программных модулей 29](#_Toc106188058)

[Приложение Б Формы входных и выходных данных 39](#_Toc106188059)

[Приложение В Результаты работы программы 40](#_Toc106188060)

Введение

-Алкогольные напитки – [напитки](https://ru.wikipedia.org/wiki/%D0%9D%D0%B0%D0%BF%D0%B8%D1%82%D0%BE%D0%BA), содержащие [этиловый спирт](https://ru.wikipedia.org/wiki/%D0%AD%D1%82%D0%B8%D0%BB%D0%BE%D0%B2%D1%8B%D0%B9_%D1%81%D0%BF%D0%B8%D1%80%D1%82). Существует множество разновидностей спиртных напитков, большинство из них производят [брожением](https://ru.wikipedia.org/wiki/%D0%91%D1%80%D0%BE%D0%B6%D0%B5%D0%BD%D0%B8%D0%B5). [Этанол](https://ru.wikipedia.org/wiki/%D0%AD%D1%82%D0%B0%D0%BD%D0%BE%D0%BB) является природным [психоактивным веществом](https://ru.wikipedia.org/wiki/%D0%9F%D1%81%D0%B8%D1%85%D0%BE%D0%B0%D0%BA%D1%82%D0%B8%D0%B2%D0%BD%D1%8B%D0%B5_%D0%B2%D0%B5%D1%89%D0%B5%D1%81%D1%82%D0%B2%D0%B0), оказывающим угнетающее действие на [центральную нервную систему](https://ru.wikipedia.org/wiki/%D0%A6%D0%B5%D0%BD%D1%82%D1%80%D0%B0%D0%BB%D1%8C%D0%BD%D0%B0%D1%8F_%D0%BD%D0%B5%D1%80%D0%B2%D0%BD%D0%B0%D1%8F_%D1%81%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0). Среди [спиртов](https://ru.wikipedia.org/wiki/%D0%A1%D0%BF%D0%B8%D1%80%D1%82%D1%8B) этанол имеет относительно невысокую [токсичность](https://ru.wikipedia.org/wiki/%D0%A2%D0%BE%D0%BA%D1%81%D0%B8%D1%87%D0%BD%D0%BE%D1%81%D1%82%D1%8C), обладая при этом значительным психоактивным эффектом. Употребление этанола вызывает [опьянение](https://ru.wikipedia.org/wiki/%D0%90%D0%BB%D0%BA%D0%BE%D0%B3%D0%BE%D0%BB%D1%8C%D0%BD%D0%BE%D0%B5_%D0%BE%D0%BF%D1%8C%D1%8F%D0%BD%D0%B5%D0%BD%D0%B8%D0%B5), в результате чего у человека снижается скорость реакции и [внимание](https://ru.wikipedia.org/wiki/%D0%92%D0%BD%D0%B8%D0%BC%D0%B0%D0%BD%D0%B8%D0%B5), нарушается координация движений и [мышление](https://ru.wikipedia.org/wiki/%D0%9C%D1%8B%D1%88%D0%BB%D0%B5%D0%BD%D0%B8%D0%B5_(%D0%BF%D1%81%D0%B8%D1%85%D0%BE%D0%BB%D0%BE%D0%B3%D0%B8%D1%8F)). Употребление больших количеств может привести к смерти. Чрезмерное и регулярное употребление алкоголя может вызывать [алкоголизм](https://ru.wikipedia.org/wiki/%D0%90%D0%BB%D0%BA%D0%BE%D0%B3%D0%BE%D0%BB%D0%B8%D0%B7%D0%BC).

Целью курсового проекта на тему «Программное средство для подсчёта промилле «Алкогольный калькулятор» является создание программы «ppmCalculator.exe», предназначенной для помощи людям страдающим алкоголизмом. Данная программа позволит узнать количество промилле, калорий, доз на основе выбранных выпитых вами алкогольных напитков.

Решение поставленных задач отражено в пояснительной записке, которая состоит из шести разделов и содержит необходимую и достаточную информацию по использованию данного программного средства.

В первом разделе «Объектно-ориентированный анализ и проектирование системы» рассматривается сущность и актуальность поставленной задачи, описание существующих аналогов, проектирование модели, отображающей функциональную структуру объекта.

Второй раздел «Вычислительная система» содержит описание вычислительной системы: технические характеристики персонального компьютера, требования, которые будут предъявляться к персональному компьютеру, описание операционной системы, языка реализации и языка моделирования.

Третий раздел «Проектирование задачи» включает требования к программному средству, концептуальный прототип, логическую и физическую организацию данных в контексте среды разработки, описание функций пользователя в рамках среды разрабатываемого программного средства, проектирование справочной системы программного средства.

Четвертый раздел «Описание программного средства» отражает общие сведения о программе, функциональное назначение, структуру входных и выходных данных.

В пятом разделе «Методика испытаний» рассматриваются требования к техническим средствам для проведения испытания, требования к характеристикам программы применительно к условиям эксплуатации, требования к информационной и программной совместимости. Также описывается порядок проведения функционального тестирования.

Шестой раздел «Применение» содержит информацию, необходимую в процессе эксплуатации программного средства: его назначение, условия применения, а также приводится описание справочной системы.

В заключении описывается выполнение поставленной задачи, степень соответствия проектных решений задания, причины несоответствия, если таковые имеются.

В приложении А представлен текст программных модулей и в приложении Б – формы выходных документов.

Графическая часть представлена диаграммами вариантов использования, классов, последовательности, деятельности и компонентов. Диаграмма последовательности сделана для функций отображения таблиц и взаимодействие с бизнес-логикой приложения. Диаграмма деятельности сделана для функции добавления напитков в базу данных и их экспорта в Word.

1. Объектно-ориентированный анализ и проектирование систем
   1. Сущность задачи

Предметной областью решаемой задачи является калькулятор «Алкогольный калькулятор». Объектом решаемой задачи является человек употребляющий алкогольные напитки.

Для человека, который хочет бросить употреблять алкоголь – нужен контроль. Человек пьющий алкоголь, не зная, что значит промилле или какой его предел выпитых доз, может попасть в ужасные ситуации, одна из которых – это смерть. Следовательно, человек-алкоголик обязан знать, что значит промилле и какие последствия его ждут, если он не будет соблюдать общепринятые нормы.

Так же чрезмерное употребление спиртного может привести к увеличению веса.

Алкогольная промилле – это концентрация чистого алкоголя в спиртном напитке. Считается она по формуле C=A/(m\* r), где

* С – концентрация алкоголя в промилле,
* A – выпитый чистый алкоголь,
* m – вес тела,
* r – коэффициент, который равняется 0,7 для мужчин и 0,6 для женщин.

С каждым часом промилле в организме убывает на 0.15. Исходя из этого, мы можем сделать прогноз, через какой промежуток времени человек будет трезв или же сможет сесть за руль.

Алкогольная доза – это единица в который измеряется алкоголь. 1 доза – это 10г чистого алкоголя. Считается по формуле D = V\*s\*0.789, где

* D – доза алкоголя,
* V – объём в литрах,
* S – крепость напитка.

Алкогольный калькулятор предоставляет такие функции, как:

* подсчёт промилле на основе выпитых вами напитков;
* подсчёт дозы на основе выпитых вами напитков;
* подсчёт калорий на основе выпитых вами напитков;
* подсчёт индивидуальной верхней границы разового потребления;
* проведение анализа на основе выбранных дней и количестве доз которые были выпиты в эти дни.
* Формирование отчётов(список выпитых вами напитков).

У данного продукта есть аналоги, но данный продукт имеет функции, которых нет у аналогов, такие как построение диаграммы, сохранение в Word списка напитков, которые были выпиты и т.д.

* 1. Проектирование модели

Цель моделирования данных состоит в обеспечении разработчика информационной системы концептуальной схемой базы данных в форме одной или нескольких локальных моделей, которые относительно легко могут быть отображены в любую систему без данных.

Наиболее распространенным средством моделирования данных является диаграмма «Сущность-связь» (ERD). С их помощью определяется важные для предметной области объекты (сущности), их свойства (атрибуты) и отношения друг с другом (связи). ERD непосредственно используются для проектирования реляционных баз данных. Нотация ERD была впервые введена П. Ченном и получился дальнейшее развитие в работах Баркера.

Диаграмма «Сущность-связь» представлена на рисунке 1.

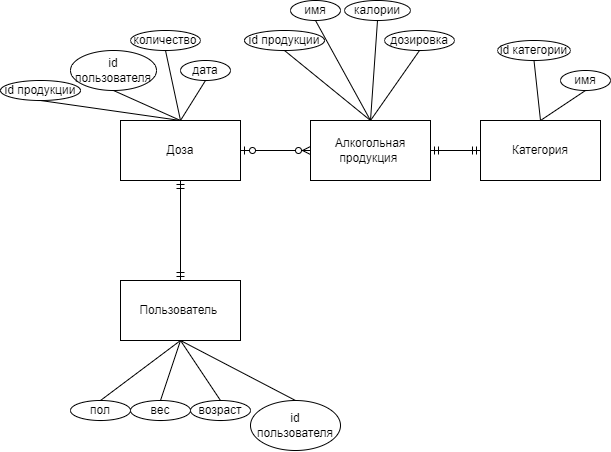


Рисунок 1

Исходя из предметной области можно выделить следующие сущности разработки: «Пользователь», «Доза», «Алкогольная продукция», «Категория».

Для сущности «Пользователь» атрибутами будут являться:

* пол;
* вес;
* возраст;
* id пользователя.

Для сущности «Доза» атрибутами будут являться:

* id продукции;
* id пользователя;
* дата
* количество.

Для сущности «Алкогольная продукция» атрибутами будут являться:

* id продукции;
* имя;
* калории;
* дозировка.

Для сущности «Категория» атрибутами будут являться:

* id категории;
* имя.

Суть диаграммы вариантов использования заключается в том, что проектируемая система представляется в виде множества сущностей или актеров, взаимодействующих с системой с помощью, так называемых вариантов использования.

К «extend» относится входная информация:

* календарь;
* OxyPlot диаграмма;
* об услугах;
* добавление употребимого;
* удаление употребимого;
* база данных;
* справочная система.

К «include» относится:

* калькулятор промилле;
* калькулятор калорий;
* использование ПО;
* калькулятор дозы;
* верхняя граница разового потребления;
* недельный калькулятор;

Диаграмма вариантов использования представлена на листе 1 графической части.

Диаграмма классов служит для представления статической структуры модели системы в терминологии классов объектно-ориентированного программирования. Диаграмма может отражать различные взаимосвязи между отдельными сущностями предметной области, такими как объекты и подсистемы, а также описывать их внутреннюю структуру и типы отношений. Для организации приложения «Алкогольный калькулятор» в программном средстве реализовано 10 классов: «MainWindow», «StartMenu» , «calorieCalculator», «doseCalculator», «weeklyCalculator», «drinkedDrinks», «drinkedDose», «drinkedCalorie», «upperBound», «Information». Класс «MainWindow» содержит в себе методы, благодаря которым вычисляется количество выпитых промилле на основе выбранных напитков. Класс «StartMenu» включает в себя методы, благодаря которым вы можете выбрать калькулятор, которым хотите воспользоваться. «calorieCalculator» содержит в себе методы для подсчёта калорий на основе выбранных вами напитков. «doseCalculator» содержит в себе методы для подсчёта доз алкоголя на основе выбранных вами напитков. «weeklyCalculator» включает в себя методы для работы с календарём, oxyplot диаграммы, подсчёта доз. «upperBound» включает в себя методы для подсчёта дозы при которой вы достигнете одной промилле с учётом ваших индивидуальных показателей(пола и веса). «drinkedDrinks», «drinkedDose», «drinkedCalorie» включают в себя методы для работы с одной и той же базой данных. Описанные классы представлены на листе 2 графической части.

При моделировании поведения проектируемой или анализируемой системы возникает необходимость детализировать особенности алгоритмической и логической реализации выполняемых системой операций. Для моделирования процесса выполнения операций в языке UML используются так называемые диаграммы деятельности.

Диаграммы деятельности – частный случай диаграмм состояний. Основная цель использования таких диаграмм – визуализация особенностей реализации операций классов, когда необходимо представить алгоритмы их выполнения. Диаграмма деятельности представлена на листе 3 графической части.

Временной аспект поведения имеет существенное значение при моделировании синхронных процессов, описывающих взаимодействия объектов. Именно для этой цели и используются диаграммы последовательности, в которых ключевым моментом является динамика взаимодействия объектов во времени. Он включает программу, вводит данные, сохраняет данные – последовательность действий. Например, пользователь переходит в выбранные им калькуляторы, добавляет и сохраняет данные, что является временным аспектом поведения пользователя. Он включает программу, вводит данные, сохраняет данные – последовательность действий. Диаграмма последовательности представлена на листе 4 графической части.

Диаграмма компонентов описывает объекты реального мира – компоненты программного обеспечения. Эта диаграмма позволяет определить архитектуру разрабатываемой системы, установив зависимости между программными компонентами, в роли которых может выступать исходный, бинарный и исполняемый код.

Определим компоненты для данного программного средства и построим диаграмму компонентов. Главным исполняемым файлом является файл «ppmCalculator.exe». Этот файл является результатом компиляции проекта «ppmCalculator.sln». Компонент «АРМКольцо.sln» реализован в среде разработки Visual Studio 2019. В состав компонента «ppmCalculator.sln» входят следующие файлы:

* файлы проекта с расширениями \*.cs, \*.resx, \*.cd, \*.xaml, \*.config, \*.png;
* файлы отчетов \*.docx, реализованные в текстовом процессоре MS Word;
* файлы базы данных «AlcoCalc.mdf» и «AlcoCalc\_log.ldf», созданные в  SQL Server 2019.

Вид диаграммы компонентов для данной проектируемой системы представлен в графической части на листе 5.

1. Вычислительная система
   1. Требования к аппаратным и операционным ресурсам

Минимальными требованиями к аппаратному и программному обеспечению, необходимыми для корректной работы программы являются:

* операционная система Windows 7 и выше;
* процессор Intel Core i5 8300 2800 МГц;
* объем оперативной памяти 1024 МБ и более;
* мышь;
* монитор;
* клавиатура.

Наиболее удобной операционной системой для проведения испытаний является Windows 7, так как она ориентирована на максимальное использование всех возможностей персонального компьютера, сетевых ресурсов и обеспечение комфортных условий работы.

На компьютере должны быть установлены программы Microsoft SQL Server 2016 и Microsoft Office Word 2016.

* 1. Инструменты разработки

Ниже описаны инструменты разработки, которые будут использоваться для написания программного средства.

Персональный компьютер со следующей аппаратной конфигурацией:

* процессор Intel Core i5 8300 2800 МГц;
* оперативная память DDR4 8 ГБ;
* встроенная видеокарта NVIDIA GeForce GTX 1050 Ti;
* винчестер SSD 512 ГБ;
* клавиатура и мышь.

Программное обеспечение:

* операционная система Windows 10;
* среда программирования Visual Studio 2019 c использованием интерфейса программирования приложений WPF;
* система управления базами данных (СУБД) SQL Server 2019;
* интегрированная среда SQL Server Management Studio (SSMS) 2018;
* программная платформа Microsoft .NET Framework 4.5;
* офисный пакет приложений Microsoft Office (MS Office) 2016;
* программа для построения диаграмм Diagrams.net;

Операционная система – это набор управляющих программ, предназначенных для управления ресурсами вычислительной системы как единого комплекса, другими словами, операционная система – это набор программного обеспечения, который обеспечивает работу компьютера.

При разработке программного средства использовалась операционная система Windows 10, так как на данный момент эта операционная система является самой распространённой операционной системой. В Windows 10 были исправлены практически все недостатки предыдущих операционных систем. Аппаратные требования Windows 10 скромнее, она способна работать даже на маломощных компьютерах и ещё добавлено множество функций, существенно облегчающих работу за компьютером.

Visual Studio 2019 – это лучший инструмент для разработки любого приложения под любую платформу. Система управления версиями в этом выпуске делает разработку гибкой, а совместную работу – эффективной [10].

Windows Presentation Foundation – аналог WinForms, система для построения клиентских приложений [Windows](https://ru.wikipedia.org/wiki/Windows) с визуально привлекательными возможностями взаимодействия с пользователем, графическая (презентационная) подсистема в составе [.NET Framework](https://ru.wikipedia.org/wiki/.NET_Framework) (начиная с версии [3.0](https://ru.wikipedia.org/wiki/.NET_Framework_3.0)), использующая язык [XAML](https://ru.wikipedia.org/wiki/XAML). [12].

СУБД Microsoft SQL Server 2019 – основной продукт Microsoft по обработке данных. Версия 2018 – это наиболее значительный прорыв в истории платформ данных Microsoft: более быстрые транзакции и запросы, ценные сведения на любом устройстве, расширенная аналитика, новые технологии безопасности и новые варианты использования при применении гибридного облака. SQL Server 2016 предоставляет передовые функции для решения задач высокой важности за счет выполнения операций в памяти и встроенной операционной аналитики [9].

SQL Server – это хорошо масштабируемый, полностью реляционный, быстродействующий многопользовательский сервер баз данных масштаба предприятия, способный обрабатывать большие объемы данных для клиент-серверных приложений [9].

.NET Framework – [программная платформа](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%BD%D0%B0%D1%8F_%D0%BF%D0%BB%D0%B0%D1%82%D1%84%D0%BE%D1%80%D0%BC%D0%B0#%D0%BA%D1%80%D0%BE%D1%81%D1%81%D0%BF%D0%BB%D0%B0%D1%82%D1%84%D0%BE%D1%80%D0%BC%D0%B5%D0%BD%D0%BD%D0%BE%D0%B5_%D0%BF%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), выпущенная компанией [Microsoft](https://ru.wikipedia.org/wiki/Microsoft) в [2002 году](https://ru.wikipedia.org/wiki/2002_%D0%B3%D0%BE%D0%B4). Основой платформы является общеязыковая среда исполнения [Common Language Runtime (CLR)](https://ru.wikipedia.org/wiki/Common_Language_Runtime), которая подходит для разных языков программирования. Функциональные возможности CLR доступны в любых языках программирования, использующих эту среду [7].

Microsoft Office 2016 – офисный пакет приложений, созданных корпорацией Microsoft для операционных систем Microsoft Windows [8].

1. Проектирование задачи
   1. Требования к приложению

Разрабатываемое приложение должно иметь понятный и удобный в использовании интерфейс, чтобы взаимодействие между программой и пользователем было максимально упрощено.

Согласно общим требованиям, графический интерфейс разрабатываемого приложения должен:

* ориентироваться на пользователя, который общается с программой на внешнем уровне взаимодействия;
* сохранять стандартизированное назначение и местоположение на экране графических объектов, работающих в среде Windows.

Интерфейс программного приложения будет разрабатываться с учетом общих требований к пользовательскому интерфейсу. Пользователь может изменять настройки по своему усмотрению при работе с программным средством.

Существуют общие требования, которые предъявляются к программным приложениям:

* соответствие стандартам организации интерфейса: использование многооконного подхода, реализация управления работой программного приложения с помощью элементов управления;
* требования к выбранной цветовой схеме рабочего экрана;
* выполнение одной функции с помощью разных элементов управления.

Пользователю необходимо установить пакеты SQL Server 2016 и Microsoft Office 2016 для работы с базами данных и переноса отчетов в документ Microsoft Office Word 2016.

Для обучения пользователя необходимо разработать справочную систему, в которой должны быть раскрыты все аспекты работы с программой, возможные трудности, возникшие во время работы и пути их решения.

Для удобства работы пользователя с программным средством необходимо при разработке форм придерживаться единого стиля оформления. Формы не должны быть перегружены излишней информацией или содержать информацию, не относящуюся к данной форме. Также необходимо предусмотреть защиту данных от удаления и изменения, а также от ввода некорректных данных. В случае ввода некорректных данных или попытке совершить запрещенные действия, пользователь должен быть проинформирован о своих действиях с помощью диалоговых окон.

* 1. Концептуальный прототип

Концептуальный прототип состоит из описания внешнего пользовательского интерфейса, а именно, элементов управления.

При создании данного приложения важную роль играют окна WPF, так как они являются основным средством работы с пользователем. Разрабатываемое приложение будет содержать несколько окон.

В рабочем режиме программы, пользователю, для удобной навигации, будет предоставлено меню, с которой можно получить справочную информацию о работе с приложением.

Приложение имеет несколько основных окон, которые будут иметь ScrollViewer с перечнем напитков, ScrollBar передвижение которого позволит вам установить ваш вес, RadioBatton’s которые позволят указать ваш пол, Label’s которые будут показывать вам результат подсчёта программы, а так же Button’s.

Главное меню формы будет состоять из восьми кнопок. Кнопка со знаком вопроса будет открывать окно со справкой о приложении. Кнопки «Калькулятор промилле», «Калькулятор дозы», «Верхняя граница разового потребления», «Недельный калькулятор», «Калькулятор калорий» будут открывать калькуляторы, выполняющие подсчёты согласно их названиям. Кнопка «Русский» и «Английский» будут меня локализацию данного приложения.

Макет главного меню представлен на рисунке 2.



Рисунок 2

При нажатии на кнопку «Недельный калькулятор» будет загружено окно, которое будет позволять пользователю выбрать дни в которые употреблял, используя встроенный календарь и записывать количество выпитых доз в соответствующий день. При конечном подсчёте на экран отобразиться диаграмма, построенная на основе введённых вами данных (дни-дозы).

Макет окна недельного калькулятора для отчетов представлен на рисунке 3.

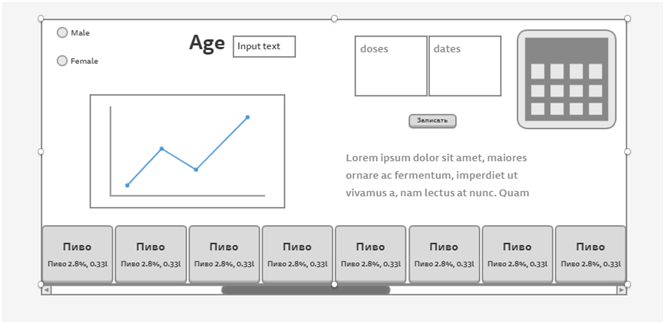


Рисунок 3

При нажатии на вкладку «Калькулятор промилле» будет загружено окно, которое будет позволять пользователю подсчитать промилле, записать выбранные напитки в базу данных. Так же покажет прогноз времени, через которое алкоголь из организма выветрится.

Макет окна калькулятора промилле представлен на рисунке 4.

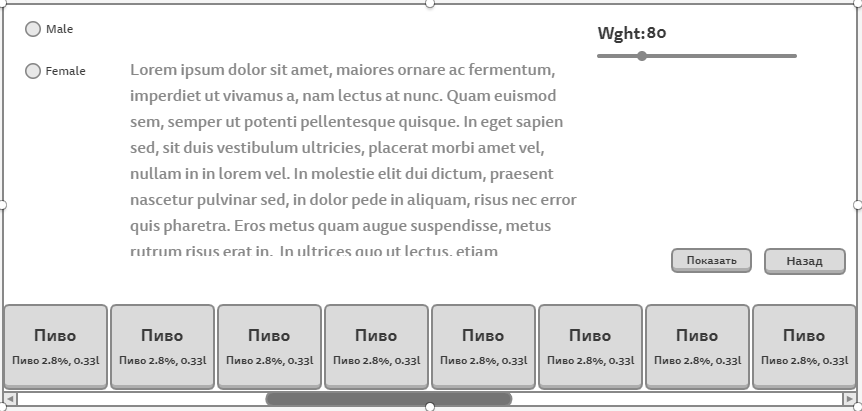


Рисунок 4

При нажатии на кнопку «Показать» будет загружено окно с таблицей, которая позволит пользователю просмотреть совершенные действия в базе данных.

* 1. Организация данных

Организация данных подразумевает создание модели данных, главными элементами которой являются сущности и их связи.

Реляционная модель основана на математическом понятии отношения, представлением которого является таблица. В реляционной модели отношения используются для хранения информации об объектах, представленных в базе данных. Отношение имеет вид двухмерной таблицы, в которой строки соответствуют записям, а столбцы – атрибутам. Каждая запись должна однозначно характеризоваться в таблице. Для этого используют первичные и внешние ключи. Достоинством реляционной модели является простота и удобство физической реализации.

Структура базы данных разрабатываемого программного средства включает одну таблицу. Описание таблицы приводится в таблицах 1.

Таблица «AlcoCalcs» хранит информацию об выпитых пользователем напитков, структура которой представлена в таблице 1.

Таблица 1 – Структура таблицы «AlcoCalcs»

|  |  |  |  |
| --- | --- | --- | --- |
| Имя поля | Тип  поля | Размер поля,  байт | Описание поля |
| id | int | 4 | Идентификатор арендатора |
| drink | nvarchar | 50 | Название выпитого напитка |
| strength | nvarchar | 50 | Крепость выпитого напитка |
| volume | nvarchar | 50 | Объём напитка |
| ppm | nvarchar | 50 | Количество промилле |
| clearalc | nvarchar | 50 | Количество чистого алкоголя |
| weight | nvarchar | 50 | Вес пользователя |
| hours | nvarchar | 50 | Время через которое выветрится алкоголь |

Физическая схема базы данных представлена на рисунке 5.

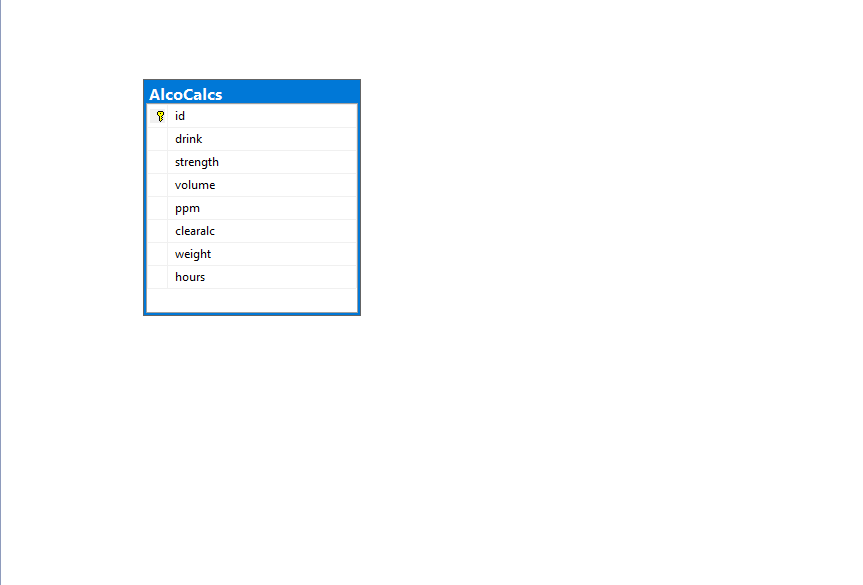


Рисунок 5

* 1. Функции: логическая и физическая организация

На основании диаграммы вариантов использования в программном средстве «ppmCalculator.exe» реализованы функции выбора калькулятора, расчета промилле, калорий, доз, верхней границы разового потребления.

Рассмотрим данные функции программы. Для того чтобы открыть один из калькуляторов, необходимо нажать на кнопку из меню. Код функции представлен ниже.

public partial class StartMenu : Window

{

public string connectionString;

public StartMenu()

{

InitializeComponent();

connectionString = ConfigurationManager.ConnectionStrings["DefaultConnection"].ConnectionString;

}

//метод для удаления данных из таблицы AlcoCalcs

public void TruncateTable()

{

SqlConnection connection = null;

string sql = "TRUNCATE TABLE AlcoCalcs";

connection = new SqlConnection(connectionString);

SqlCommand command = new SqlCommand(sql, connection);

connection.Open();

command.ExecuteNonQuery();

connection.Close();

}

//метод для открытия калькулятора промилле

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

{

TruncateTable();

MainWindow obj = new MainWindow();

obj.Show();

this.Close();

}

//метод для открытия калькулятора дозы

private void Button\_Click\_1(object sender, RoutedEventArgs e)

{

doseCalculator obj = new doseCalculator();

obj.Show();

this.Close();

}

//метод для открытия верхняя граница разового потребления

private void Button\_Click\_2(object sender, RoutedEventArgs e)

{

upperBound obj = new upperBound();

obj.Show();

this.Close();

}

//метод для открытия калькулятора калорий

private void Button\_Click\_3(object sender, RoutedEventArgs e)

{

calorieCalculator obj = new calorieCalculator();

obj.Show();

this.Close();

}

//метод для открытия недельного калькуляторого

private void Button\_Click\_4(object sender, RoutedEventArgs e)

{

weeklyCalculator obj = new weeklyCalculator();

obj.Show();

this.Close();

}

Для того, чтобы посчитать количество промилле, надо перейти в одноимённый калькулятор. При подсчёте промилле, выбранные напитки записываются в базу данных. Так же вам покажет прогноз времени, через которое алкоголь из вашего организма выветрится. Код функции представлен ниже.

public partial class MainWindow : Window

{

public double weigth = 0;

public double ppm = 0;

public double ppmperone = 0;

public double hourperone = 0;

public double []ppms = new double[20];

public double[] hour = new double[20];

public int i = 0;

public double time = 0;

public string connectionString;

public MainWindow()

{

InitializeComponent();

connectionString = ConfigurationManager.ConnectionStrings["DefaultConnection"].ConnectionString;

}

//C=A/(m\* r), где

//С – концентрация алкоголя в промилле,

//A – выпитый чистый алкоголь(можно рассчитать, взяв указанный на этикетке напитка процент),

//m – вес тела,

//r – коэффициент, который равняется 0,7 для мужчин и 0,6 для женщин.

//метод для подсчёта прогноза

private void HowManyHours()

{

hourperone = ppmperone / 0.15;

time = ppm / 0.15;

var str = string.Format("{0:0.#}", time);

hours.Content = str;

}

//метод для записи в массивы количество промилле за один напиток и прогноз выветривания

public void SavePpmsAndHoursPerOne()

{

ppms[i] = ppmperone;

hour[i] = hourperone;

i++;

}

//метод для записи напитков в БД

public void EnterDataIntoDB(string type, string strength, string volume, double clearAlc)

{

SqlConnection connection = null;

string query = $"INSERT INTO AlcoCalcs(drink, strength, volume, ppm, clearalc) values('{type}','{strength}','{volume}','{ppm}','{clearAlc}')";

connection = new SqlConnection(connectionString);

SqlCommand command = new SqlCommand(query, connection);

connection.Open();

if (command.ExecuteNonQuery() == 1) { }

connection.Close();

}

//подсчёт промилле

public void CalculatePpm(double clearAlc)

{

if (male.IsChecked == false && female.IsChecked == false && Title== "Калькулятор промилле")

MessageBox.Show("Укажите ваш пол");

if(male.IsChecked == false && female.IsChecked == false && Title == "PPM Calculator")

MessageBox.Show("Choose your sex");

if (male.IsChecked == true)

{

ppmperone = clearAlc / (0.7 \* weigth);

ppm += clearAlc / (0.7 \* weigth);

var str = string.Format("{0:0.#}", ppm);

result.Content = str;

}

if (female.IsChecked == true)

{

ppmperone = clearAlc / (0.6 \* weigth);

ppm += clearAlc / (0.6 \* weigth);

var str = string.Format("{0:0.#}", ppm);

result.Content = str;

}

}

//метод для открытия главного меню

private void Button\_Click\_1(object sender, RoutedEventArgs e)

{

StartMenu obj = new StartMenu();

obj.Show();

this.Close();

}

//метод для открытия таблицы с БД

private void Button\_Click\_3(object sender, RoutedEventArgs e)

{

drinkedDrinks obj = new drinkedDrinks(ppm, ppms, time, hour, ppmperone, hourperone);

obj.Show();

this.Close();

}

//кнопка при нажатии на которую считается промилле, прогноз, ввод данных в БД

private void beer28\_Click(object sender, RoutedEventArgs e)

{

var type = " ";

if (Title=="Калькулятор промилле")

{

type = "Пиво";

}

else

{

type = "Beer";

}

var strength = "2.8";

var volume = "0.33";

double clearAlc = 9.24;

weigth = Weight.Value;

EnterDataIntoDB(type, strength, volume, clearAlc);

CalculatePpm(clearAlc);

HowManyHours();

SavePpmsAndHoursPerOne();

}

Для того, чтобы посчитать количество доз, надо перейти в одноимённый калькулятор. При подсчёте доз, выбранные напитки записываются в базу данных. Код функции представлен ниже.

public partial class doseCalculator : Window

{

public int i = 0;

public double doseperone = 0;

public double[] doses = new double[20];

public double dose = 0;

public string connectionString;

//количество единиц алкоголя = количество(литры) х крепость(%) х 0,789

//Для расчета количества единиц алкоголя в конкретном напитке нужно умножить объем напитка в литрах на его

//крепость, а затем умножить полученный результат на удельный вес алкоголя, т.е.на 0,789

//(соотношение между плотностью алкоголя и плотностью воды).

//метод для подсчёта дозы

public void CalculateDose(double volume, double strength)

{

doseperone = volume \* strength \* 0.789;

dose += volume \* strength \* 0.789;

var str = string.Format("{0:0.#}", dose);

doseRes.Content = str;

doses[i] = doseperone;

++i;

}

//метод предназначенный для очистки таблицы БД

public void TruncateTable()

{

SqlConnection connection = null;

string sql = "TRUNCATE TABLE AlcoCalcs";

connection = new SqlConnection(connectionString);

SqlCommand command = new SqlCommand(sql, connection);

connection.Open();

command.ExecuteNonQuery();

connection.Close();

}

public doseCalculator()

{

InitializeComponent();

connectionString = ConfigurationManager.ConnectionStrings["DefaultConnection"].ConnectionString;

}

//метод для ввода данных в БД

public void EnterDataIntoDB(string type, string strength, string volume)

{

SqlConnection connection = null;

string query = $"INSERT INTO AlcoCalcs(drink, strength, volume) values('{type}','{strength}','{volume}')";

connection = new SqlConnection(connectionString);

SqlCommand command = new SqlCommand(query, connection);

connection.Open();

if (command.ExecuteNonQuery() == 1) { }

connection.Close();

}

//кнопка при нажатии на которую считается промилле, прогноз, ввод данных в БД

private void beer28\_Click(object sender, RoutedEventArgs e)

{

var type = " ";

if (Title == "Калькулятор дозы")

{

type = "Пиво";

}

else

{

type = "Beer";

}

double volume = 0.33;

double strength = 2.8;

CalculateDose(volume, strength);

EnterDataIntoDB(type, strength.ToString(), volume.ToString());

}

Для того, чтобы узнать сколько пользователь выпил за определённое количество дней в месяце доз, надо перейти в окно недельного калькулятора. В нём он можете выбрать дни в которые употреблял используя встроенный календарь и записывать количество выпитых доз в соответствующий день. При конечном подсчёте на экран отобразиться диаграмма, построенная на основе введённых данных (дни-дозы). Код функции представлен ниже.

public partial class weeklyCalculator : Window

{

public int age = 0;

public double dose\_ = 0;

public double dose = 0;

public double allDose = 0;

public int i = 0;

public weeklyCalculator()

{

InitializeComponent();

}

//метод для возвращения в главное меню

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

{

StartMenu obj = new StartMenu();

obj.Show();

this.Close();

}

//метод который уведомиляет вас о превышении предела доз на основе ваших показателей

public void MessageForBadass()

{

age = Convert.ToInt32(howold.Text);

if (age < 18 && Convert.ToDouble(AllDose.Content) >= 19.2)

MessageBox.Show("Вы употребляете СЛИШКОМ МНОГО для своего возраста");

if(age>=18 && age<65 && Convert.ToDouble(AllDose.Content) >= 56)

MessageBox.Show("Вы употребляете БОЛЬШЕ чем следовало бы для своего возраста");

if (age >= 18 && age < 65 && Convert.ToDouble(AllDose.Content) >= 92)

MessageBox.Show("Вы употребляете СЛИШКОМ МНОГО для своего возраста");

if (age > 65 && Convert.ToDouble(AllDose.Content) >= 7)

MessageBox.Show("Вы употребляете БОЛЬШЕ чем следовало бы для своего возраста");

}

//метод который создаёт диаграмму

public void CreateDiagramm()

{

DateTimeAxis xAxis = new DateTimeAxis

{

Position = AxisPosition.Bottom,

Title = "Day-Month",

};

FunctionSeries fs = new FunctionSeries();

for (int i = 0; i < dates.Items.Count; i++)

{

fs.Points.Add(new DataPoint(DateTimeAxis.ToDouble(dates.Items[i]), Convert.ToDouble(doses.Items[i])));

}

PlotModel n = new PlotModel();

n.Series.Add(fs);

n.Axes.Add(xAxis);

n.Axes.Add(new LinearAxis());

grafic.Model = n;

}

//метод который проверяет - указали ли вы ваш пол, возраст и указали ли вы дату(ы) в которые употребляли спиртное

public void Checkes()

{

if (male.IsChecked == false && female.IsChecked == false && Title == "Недельный калькулятор")

MessageBox.Show("Укажите ваш пол");

if (male.IsChecked == false && female.IsChecked == false && Title == "Weekly calculator")

MessageBox.Show("Choose your sex");

if (howold.Text == " " && Title == "Недельный калькулятор")

MessageBox.Show("Укажите ваш возраст");

if (howold.Text == " " && Title == "Weekly calculator")

MessageBox.Show("Enter your age");

if (calendar.SelectedDate == null && Title == "Недельный калькулятор")

MessageBox.Show("Выберите хотя бы одну дату");

if (calendar.SelectedDate == null && Title == "Weekly calculator")

MessageBox.Show("Select at least one date");

}

//метод который ситает дозы

public void CalculateDose(double volume, double strength)

{

dose\_ += volume \* strength \* 0.789;

dose += volume \* strength \* 0.789;

var str = string.Format("{0:0.#}", dose);

var zxc = string.Format("{0:0.#}", dose\_);

doseRes.Content = str;

AllDose.Content = zxc;

}

//кнопка записи доз в TextBox

private void Button\_Click\_1(object sender, RoutedEventArgs e)

{

Checkes();

var str = string.Format("{0:0.#}", dose);

doses.Items.Add(str);

++i;

dose = 0;

doseRes.Content = dose;

if (dates.Items.Count.Equals(doses.Items.Count))

{

MessageBox.Show("Вы успешно заполнили все дни");

CreateDiagramm();

MessageForBadass();

}

}

//кнопка считающая дозы

private void beer28\_Click(object sender, RoutedEventArgs e)

{

double volume = 0.33;

double strength = 2.8;

CalculateDose(volume, strength);

}

Полный текст программы представлен в приложении А.

* 1. Проектирование справочной системы приложения

Для эффективной работы пользователя с приложением необходимо обеспечить его качественной справочной системой, в которой должны быть приведены методы и приемы работы с приложением, включающие данные о том, что произойдет после нажатия на определенную кнопку или при выборе пункта меню, сведения о том, какую информацию и в каком виде следует вводить в соответствующие поля, каким образом можно работать.

Справочная система будет представлена в виде отдельного окна в приложении, содержащая информацию по эксплуатации программного средства.

Справочная система будет вызываться при нажатии на кнопку со знаком вопроса на главной меню.

1. Описание программного средства
   1. Общие сведения

Программное средство «Алкогольный калькулятор» разрабатывалось с целью помочь людям, которые страдают алкоголизмом или же просто людям, которые хотят меньше употреблять спиртных напитков. Установка данного программного обеспечения позволяет проделывать сложные операции без ошибок. Данная программа была разработана на персональном компьютере со следующей конфигурацией:

* процессор Intel Core i5 8300 2800 МГц;
* оперативная память DDR4 8 ГБ;
* встроенная видеокарта NVIDIA GeForce GTX 1050 Ti;
* винчестер SSD 512 ГБ;
* клавиатура и мышь.

Программа создана в средстве разработки Microsoft Visual Studio 2019 на языке ограммирования C# в операционной системе Windows 10. Она может работать в средах операционных систем семейства Microsoft Windows, начиная с Windows 7.

В программе присутствует база данных «ppmCalculator.mdf». Программа не требовательна к системным ресурсам, также проста в использовании и не требует специальных навыков при работе. Для работы данного программного средства необходима предварительная установка и настройка следующих программных продуктов:

* интегрированная среда SQL Management Studio 2018;
* среда разработки Microsoft Visual Studio 2019;
* платформа Microsoft Net Framework версии 3.5.

В состав программных входят:

* программа «ppmCalculator.exe»;
* база данных «AlcoCalc.mdf»;
* журнал транзакций «AlcoCalc\_log.ldf».

Объем установленного приложения не превышает 30 МБ, файлов – 100 МБ.

Программное средство не требует логина и пароля.

* 1. Функциональное назначение

Программное средство разработано с целью – помочь избавиться людям от алкогольной зависимости.

Программное средство позволяет подсчитывать количество выпитых вами промилле, калорий, доз и верхнюю границу разового потребления. Также присутствует недельный калькулятор, в котором вы сможете записывать – сколько вы выпили в выбранный вами день.

Программа использует стандартные элементы управления, такие как кнопки, списка, поля ввода, что обеспечивает единство интерфейса системы и программного средства.

В работе программного средства предусмотрены некоторые ситуации, которые должны предупреждать пользователя, чтобы он выполнял все необходимые требования по эксплуатации программы. Для этого существуют сообщения системы, например, если пользователь не указал пол, не ввёл возраст, не выбрал хотя бы один день в недельном калькуляторе, то в этом случае на экран выводится сообщение с информацией о характерной ошибке.

Таким образом, программа может применяться в реальных условиях, представляя собой достаточно удобный помощник.

* 1. Входные и выходные данные

Входными являются данные, введённые пользователем в таблицу базы данных: «AlcoCalcs».

Входными данным при добавлении данных о выпитом напитке будут являться:

* drink;
* strength;
* volume;
* ppm;
* clearalc;
* weight;
* hours.

Выходные данные – это сформированные списки, которые необходимы для расчёта промилле, доз, калорий.

Чтобы сформировать отчет о выпитых напитках нужно нажать в окне с таблицей кнопку «Сохранить». Отчет будет добавлен на рабочий стол. Пример отчёта представлен в приложении Б на рисунке Б.1.

1. Методика испытаний
   1. Технические требования

Основными минимальными требованиями, выдвигаемыми к аппаратному обеспечению персонального компьютера, являются:

* процессор Intel Core i7 7600U 2800 МГц;
* оперативная память DDR4 8 ГБ;
* встроенная видеокарта Intel HD Graphics 620;
* винчестер SSD 512 ГБ;
* клавиатура и мышь.

Для работы с программой необходимо наличие монитора, клавиатуры и мыши.

Компьютер должен работать под управлением операционной системы, начиная с Windows 7 и выше. Необходимо наличие Microsoft Office 2016, SQL Server Management Studio 2019, Microsoft SQL Server 2019 и Microsoft .NET Framework 4.5.

* 1. Функциональное тестирование

Функциональное тестирование – это тестирование функций приложения на соответствие требованиям и проводится для выявления неполадок и недочетов программы на этапе ее сдачи в эксплуатацию.

Проведем тестирование проверки каждого пункта меню с целью проверки всех функций.

Тестирование программы будет производиться последовательно, переходя из одной части программы в другую. Во время теста будут проверяться все действия с программой, навигация пунктам меню, которые может произвести пользователь. После чего, все собранные и найденные ошибки будут исправлены.

В таблице 2 представлены тест-кейсы, подготовленные для проведения функционального тестирования. Работа каждой функции будет осуществляться с главного меню, калькуляторов промилле, доз, калорий и недельного.

Таблица 2 – Тест-кейсы для добавления данных о новом арендаторе

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Модуль / Функция | Шаги выполнения | Ожидаемый результат | Фактический результат |
| 1 | Добавление данных о выпитом напитке | 1. Выбрать калькулятор.  2.Нажать на кнопку с напитком. | Добавление в базу данных выпитые напитки. | Добавление в базу данных о выпитом напитке и отображения на форме. Результат заполнения строки данными представлен в приложении В на рисунке В.1. |
| 2 | Экспорт данных о напитках вWord | 1. Перейти в окно с таблицей.  2. Нажать на кнопку «Сохранить». | Экспорт данных о напитках в Word. | Экспорт данных о выпитых напитках в Word. Результат представлен в приложении Б на рисунке Б.1. |

Продолжение таблицы №2

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 3 | Удаление напитков | 1.Перейти в окно с таблицей.  2. Выбрать поле которое хотите удалить.  3. Нажать на кнопку «Удалить». | Напиток удалён. | Напиток удалился. Результат представлен в приложении В на рисунке В.2. |
| 4 | Смена языка | 1. Зайти в стартовое меню.  2. Нажать на кнопку «Английский» | Локализация приложения изменилась на английскую. | Смена языка. Результат локализации представлен в приложении В на рисунке В.3. |
| 5 | Построение диаграммы | 1. Перейти в окно «Недельный калькулятор».  2. Выбрать пол.  3. Заполнить поле с возрастом.  4. Выбрать даты в календаре.  5. Записать выпитые вами дозы в дни. | Диаграмма построена. | Диаграмма построилась. Результат построенной диаграммы представлена в приложении В на рисунке В.4. |
| 6 | Подсчёт промилле | 1. Перейти в окно «Калькулятор промилле»  2. Указать пол.  3. Нажимать на кнопки с напитками которые вы пили. | Промилле посчиталась | Промилле посчиталась. Результат посчитанной промилле представлен в приложении В на рисунке В.5. |
| 7 | Подсчёт верхней границы разового потребления | 1. Перейти в окно «Верхняя граница разового потребления»  2. Указать пол.  3. Указать ваш вес.  4. Нажать на кнопку «Посчитать». | Подсчёт верхней границы разового потребления | ВГРП посчиталась. Результат посчитанной ВГРП представлен в приложении В на рисунке В.6. |
| 8 | Подсчёт дозы | 1. Перейти в окно «Калькулятор дозы»  2. Указать пол.  3. Нажимать на кнопки с напитками которые вы пили. | Доза посчиталась | Доза посчиталась. Результат посчитанной дозы представлен в приложении В на рисунке В.7. |

1. Применение
   1. Назначение программы

Приложение «ppmCalculator.exe» предназначено для помощи людям страдающим алкоголизмом. Программное средство предоставляет пользователю посчитать количество промилле, калорий, доз на основе выбранных выпитых вами алкогольных напитков. Осуществляет формирование отчетов о выпитых вами напитков. Программное средство формирует диаграмму на основе дней в которые вы употребляли и сколько вы употребляли.

* 1. Условия применение

Для работы с программой необходимо наличие программного обеспечения:

* операционная система, начиная с Windows 7;
* система управления базами данных Microsoft SQL Server Express 2019;
* библиотека Microsoft .NET Framework 4.5 с пакетом обновления 1 (SP1);
* офисный пакет приложений Microsoft Office (MS Office) 2016.

На случай редактирования проекта программы необходимо наличие программного обеспечения:

* интегрированная среда Microsoft SQL Server Management Studio 2018;
* среда разработки приложений Microsoft Visual Studio 2017 или 2019.
  1. Справочная система

Справочная система представлена в виде отдельного окна в приложении, содержащая информацию по эксплуатации программного средства. Структура начальной страницы справочной системы представлена на рисунке 6.

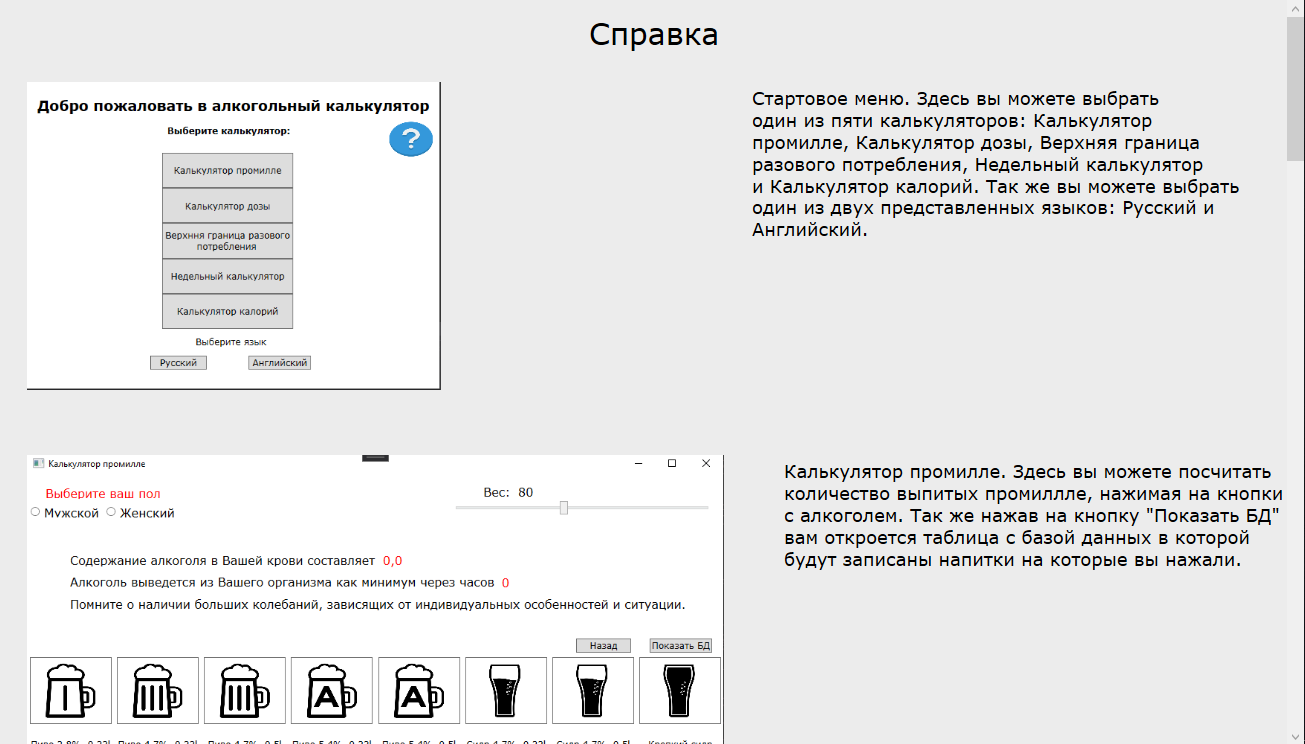


Рисунок 6

Справочная система будет вызываться при нажатии на кнопку с иконкой вопроса на главной форме.

Заключение

В рамках курсового проекта на тему «Программное средство для подсчёта промилле «Алкогольный калькулятор» было разработано программное средство «ppmCalculator.exe», подсчитывающее промилле, дозы и калории.

Для достижения целей курсового проекта были решены следующие задачи:

* определена вычислительная система, необходимая для создания программного средства;
* разработана физическая и логическая модель данных;
* по модели выполнено проектирование задачи;
* разработано программное средство;
* описано созданное программное средство;
* выбрана методика испытаний;
* описан процесс тестирования;
* приведены примеры области применения.

Разработка имеет интуитивно понятный графический интерфейс, позволяющий даже с минимальным знанием компьютера использовать данное программного средство.

Программа реализована в полном объеме и в соответствии с заданными требованиями, полностью отлажена и протестирована. Поставленные задачи выполнены.

К достоинствам данного программного средства можно отнести: интуитивно-понятный интерфейс, простоту в использовании.

В процессе разработки данного программного средства были применены и закреплены знания по уже изученному материалу, были отработаны навыки владения методами надежного программирования и эффективности разработки программного обеспечения в Microsoft Visual Studio 2019 с использованием языка программирования C#, разработана база данных средствами системы управления базами данных Microsoft SQL Server 2019.

Программное средство готово к практическому использованию.

Данная программа может быть дополнена и модернизирована.

Список информационных источников

1. Багласова, Т.Г. Методические указания по оформлению курсовых и дипломных проектов / Т.Г. Багласова, К.О. Якимович. – Минск : КБП, 2017. – 29 c.
2. Михалевич, В.Ю. Методические указания для курсового проектирования для учащихся по специальности 2-40 01 01 «Программное обеспечение технологий» / В.Ю. Михалевич. – Минск : КБП, 2022. – 24 c.
3. Общие требования к тестовым документам : ГОСТ 2.105-95. – Введ. 01.01.1996 – Минск : Межгос. совет по стандартизации, метрологии и сертификации, 1995. – 84 с.
4. Скит, Джон. С# для профессионалов: тонкости программирования / Джон Скит. – 3-е изд. : Пер. с англ. – М. : ООО «И.Д. Вильямс», 2022. – 608 с. : ил. – Парал. тит. англ.
5. Шарп, Джон. Microsoft Visual C#. Подробное руководство / Джон Шарп. – 8-е изд. – СПб.: Питер, 2022. – 848 с.: ил. – Серия : «Библиотека программиста»
6. Шварц, Б., Зайцев, П., Ткаченко, В. MySQL по максимуму / Б. Шварц, П. Зайцев, В. Ткаченко. – 3-е изд. – СПб : Питер, 2022.  – 864 с. : ил. – Серия : «Бестселлеры O’Reily»
7. .NET Framework [Электронный ресурс]. – Microsoft, 2022. – Режим доступа : https://docs.microsoft.com/en-us/dotnet/framework/. – Дата доступа : 24.05.2022.
8. MS Office [Электронный ресурс]. – Microsoft, 2022. – Режим доступа : https://officeapplications.net/microsoft-word/. – Дата доступа : 24.05.2022.
9. SQL Server 2019 [Электронный ресурс]. – Microsoft, 2022. – Режим доступа : https://docs.microsoft.com/ru-ru/sql/tutorials/tutorials-for-sql-server-2019?view=sql-server-2019/. – Дата доступа : 17.04.2022.
10. Visual Studio [Электронный ресурс]. – Microsoft, 2022. – Режим доступа : https://visualstudio.microsoft.com/vs/. – Дата доступа : 18.04.2022.
11. Полное руководство по языку программирования С# 10 и платформе .NET 6 [Электронный ресурс] / METANIT.COM. Сайт о программировании. – metanit.com, 2012-2022. – Режим доступа : http://metanit.com/sharp/tutorial. – Дата доступа : 15.03.2022.
12. Начало работы с WPF [Электронный ресурс]. – Microsoft, 2022. – Режим доступа : [https://docs.microsoft.com/en-us/visualstudio/designers/getting-started-with-wpfview=vs-2022](%20https://docs.microsoft.com/en-us/visualstudio/designers/getting-started-with-wpfview=vs-2022). – Дата доступа : 05.04.2022.

Приложение А

(обязательное)

Текст программных модулей

public partial class StartMenu : Window

{

public string connectionString;

//конструктор класса StartMenu

public StartMenu()

{

InitializeComponent();

connectionString = ConfigurationManager.ConnectionStrings["DefaultConnection"].ConnectionString;

} //метод для удаления данных из таблицы AlcoCalcs

public void TruncateTable()

{

SqlConnection connection = null;

string sql = "TRUNCATE TABLE AlcoCalcs";

connection = new SqlConnection(connectionString);

SqlCommand command = new SqlCommand(sql, connection);

connection.Open();

command.ExecuteNonQuery();

connection.Close();

}

//метод для открытия калькулятора промилле

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

{

TruncateTable();

MainWindow obj = new MainWindow();

obj.Show();

this.Close();

}

//метод для открытия калькулятора дозы

private void Button\_Click\_1(object sender, RoutedEventArgs e)

{

doseCalculator obj = new doseCalculator();

obj.Show();

this.Close();

}

//метод для открытия верхняя граница разового потребления

private void Button\_Click\_2(object sender, RoutedEventArgs e)

{

upperBound obj = new upperBound();

obj.Show();

this.Close();

}

//метод для открытия калькулятора калорий

private void Button\_Click\_3(object sender, RoutedEventArgs e)

{

calorieCalculator obj = new calorieCalculator();

obj.Show();

this.Close();

}

//метод для открытия недельного калькуляторого

private void Button\_Click\_4(object sender, RoutedEventArgs e)

{

weeklyCalculator obj = new weeklyCalculator();

obj.Show();

this.Close();

}

//метод для смены языка на русский

private void Button\_Click\_5(object sender, RoutedEventArgs e)

{

System.Threading.Thread.CurrentThread.CurrentUICulture = new System.Globalization.CultureInfo("ru-RU");

StartMenu obj = new StartMenu();

obj.Show();

this.Close();

}

//метод для смены языка на английский

private void Button\_Click\_6(object sender, RoutedEventArgs e)

{

System.Threading.Thread.CurrentThread.CurrentUICulture = new System.Globalization.CultureInfo("en");

StartMenu obj = new StartMenu();

obj.Show();

this.Close();

}

//метод для открытия окна справки

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

{

Information obj = new Information();

obj.Show();

this.Close();

}

<Window x:Class="ppmCalculator.StartMenu"

xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"

xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"

xmlns:d="http://schemas.microsoft.com/expression/blend/2008"

xmlns:p = "clr-namespace:ppmCalculator.Properties"

xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"

xmlns:local="clr-namespace:ppmCalculator"

mc:Ignorable="d"

Title="StartMenu" Height="445.333" Width="570">

<Grid>

<Label Content="{x:Static p:Resources.menuPres}" Margin="14,10,10,0" VerticalAlignment="Top" FontSize="19" HorizontalAlignment="Center" FontFamily="Verdana" FontWeight="Bold" Width="538"/>

<Label Content="{x:Static p:Resources.menuChoose}" Margin="184,48,194,329" FontWeight="Bold" FontFamily="Verdana"/>

<Grid HorizontalAlignment="Left" Height="236" Margin="138,90,0,0" VerticalAlignment="Top" Width="152" >

<Grid.RowDefinitions>

<RowDefinition></RowDefinition>

<RowDefinition></RowDefinition>

<RowDefinition></RowDefinition>

<RowDefinition></RowDefinition>

<RowDefinition></RowDefinition>

</Grid.RowDefinitions>

<Button Grid.Row="0" Content="{x:Static p:Resources.ppmTitle}" Click="Button\_Click" FontFamily="Verdana" Margin="44,0,-68,0"/>

<Button Grid.Row="1" Content="{x:Static p:Resources.doseTitle}" Click="Button\_Click\_1" FontFamily="Verdana" Margin="44,0,-68,0"/>

<Button Grid.Row="2" Content="{x:Static p:Resources.upperBoundButton}" Click="Button\_Click\_2" FontFamily="Verdana" Margin="44,0,-68,0"/>

<Button Grid.Row="3" Content="{x:Static p:Resources.weeklyTitle}" Click="Button\_Click\_4" FontFamily="Verdana" Margin="44,0,-68,0"/>

<Button Grid.Row="4" Content="{x:Static p:Resources.calorieTitle}" Click="Button\_Click\_3" FontFamily="Verdana" Margin="44,0,-68,0"/>

</Grid>

<Button Content="{x:Static p:Resources.russian}" HorizontalAlignment="Left" Margin="166,362,0,0" VerticalAlignment="Top" Width="76" Click="Button\_Click\_5" FontFamily="Verdana"/>

<Button Content="{x:Static p:Resources.usa}" HorizontalAlignment="Left" Margin="298,362,0,0" VerticalAlignment="Top" Width="84" Click="Button\_Click\_6" FontFamily="Verdana"/>

<Label Content="{x:Static p:Resources.chooseLang}" HorizontalAlignment="Left" Margin="222,331,0,0" VerticalAlignment="Top" FontFamily="Verdana"/>

<Button x:Name="infobutton" BorderBrush="White" HorizontalAlignment="Left" Margin="480,43,0,0" VerticalAlignment="Top" Width="72" Height="55" Click="infobutton\_Click">

<Button.Background>

<ImageBrush ImageSource="info.png"/>

</Button.Background>

</Button>

</Grid>

</Window>

public partial class MainWindow : Window

{

public double weigth = 0;//поле в котором находится вес

public double ppm = 0;//поле общей промилле

public double ppmperone = 0;//поле промилле за один выпитый напиток

public double hourperone = 0;//поле времени за один выпитый напиток

public double []ppms = new double[20];//поле набора промилле

public double[] hour = new double[20];//поле набора времени

public int i = 0;//поле итератора

public double time = 0;//поле общего времени через которое вы полностью станете трезвый

public string connectionString;

//конструткор класса MainWindow

public MainWindow()

{

InitializeComponent();

connectionString = ConfigurationManager.ConnectionStrings["DefaultConnection"].ConnectionString;

}

//метод для подсчёта прогноза

private void HowManyHours()

{

hourperone = ppmperone / 0.15;

time = ppm / 0.15;

var str = string.Format("{0:0.#}", time);

hours.Content = str;

}

//метод для записи в массивы количество промилле за один напиток и прогноз выветривания

public void SavePpmsAndHoursPerOne()

{

ppms[i] = ppmperone;

hour[i] = hourperone;

i++;

}

//метод для записи напитков в БД

public void EnterDataIntoDB(string type, string strength, string volume, double clearAlc)

{

SqlConnection connection = null;

string query = $"INSERT INTO AlcoCalcs(drink, strength, volume, ppm, clearalc) values('{type}','{strength}','{volume}','{ppm}','{clearAlc}')";

connection = new SqlConnection(connectionString);

SqlCommand command = new SqlCommand(query, connection);

connection.Open();

if (command.ExecuteNonQuery() == 1) { }

connection.Close();

}

//подсчёт промилле

public void CalculatePpm(double clearAlc)

{

if (male.IsChecked == false && female.IsChecked == false && Title== "Калькулятор промилле")

MessageBox.Show("Укажите ваш пол");

if(male.IsChecked == false && female.IsChecked == false && Title == "PPM Calculator")

MessageBox.Show("Choose your sex");

if (male.IsChecked == true)

{

ppmperone = clearAlc / (0.7 \* weigth);

ppm += clearAlc / (0.7 \* weigth);

var str = string.Format("{0:0.#}", ppm);

result.Content = str;

}

if (female.IsChecked == true)

{

ppmperone = clearAlc / (0.6 \* weigth);

ppm += clearAlc / (0.6 \* weigth);

var str = string.Format("{0:0.#}", ppm);

result.Content = str;

}

}

//метод для открытия главного меню

private void Button\_Click\_1(object sender, RoutedEventArgs e)

{

StartMenu obj = new StartMenu();

obj.Show();

this.Close();

}

//метод для открытия таблицы с БД

private void Button\_Click\_3(object sender, RoutedEventArgs e)

{

drinkedDrinks obj = new drinkedDrinks(ppm, ppms, time, hour, ppmperone, hourperone);

obj.Show();

this.Close();

}

//кнопка при нажатии на которую считается промилле, прогноз, ввод данных в БД

private void beer28\_Click(object sender, RoutedEventArgs e)

{

var type = " ";

if (Title=="Калькулятор промилле")

{

type = "Пиво";

}

else

{

type = "Beer";

}

var strength = "2.8";

var volume = "0.33";

double clearAlc = 9.24;

weigth = Weight.Value;

EnterDataIntoDB(type, strength, volume, clearAlc);

CalculatePpm(clearAlc);

HowManyHours();

SavePpmsAndHoursPerOne();

}

<Window x:Class="ppmCalculator.MainWindow"

xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"

xmlns:p = "clr-namespace:ppmCalculator.Properties"

xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"

xmlns:d="http://schemas.microsoft.com/expression/blend/2008"

xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"

xmlns:local="clr-namespace:ppmCalculator"

mc:Ignorable="d"

Title="{x:Static p:Resources.ppmTitle}" Height="453" Width="953" x:Name="ppmCal" x:ClassModifier="public">

<Grid HorizontalAlignment="Right" Width="953" Margin="0,0,-6,0">

<Label x:Name="sexText" Content="{x:Static p:Resources.sexText}" HorizontalAlignment="Left" Margin="30,10,0,0" VerticalAlignment="Top" Foreground="Red" FontFamily="Verdana" Height="27" Width="168" FontSize="16"/>

<Slider x:Name="Weight" Maximum="160" Minimum="20" HorizontalAlignment="Left" Margin="582,35,0,0" VerticalAlignment="Top" Width="350" TickFrequency="1" Value="80" IsSnapToTickEnabled="True" FontSize="16"/>

<RadioButton x:Name="male" Content="{x:Static p:Resources.male}" HorizontalAlignment="Left" Margin="14,42,0,0" VerticalAlignment="Top" FontFamily="Verdana" Height="15" Width="102" FontSize="16" />

<RadioButton x:Name="female" Content="{x:Static p:Resources.female}" HorizontalAlignment="Left" Margin="116,42,0,0" VerticalAlignment="Top" FontFamily="Verdana" Height="15" Width="114" FontSize="16"/>

<Label x:Name="weight" Content= "{Binding ElementName=Weight, Path=Value, UpdateSourceTrigger=PropertyChanged}" HorizontalAlignment="Left" Margin="666,8,0,0" VerticalAlignment="Top" FontFamily="Verdana" FontSize="16"/>

<Label x:Name="weightText" Content="{x:Static p:Resources.weightText}" HorizontalAlignment="Left" Margin="608,8,0,0" VerticalAlignment="Top" FontFamily="Verdana" FontSize="16"/>

<ScrollViewer CanContentScroll="True" HorizontalScrollBarVisibility="Auto" VerticalScrollBarVisibility="Hidden" Height="169" VerticalAlignment="Bottom" Margin="10,0,0,0" FontFamily="Verdana">

<DockPanel x:Name="alco" HorizontalAlignment="Left" Height="209" VerticalAlignment="Top" Width="3048" >

<Grid>

<Grid.ColumnDefinitions>

<ColumnDefinition></ColumnDefinition>

<ColumnDefinition></ColumnDefinition>

<ColumnDefinition></ColumnDefinition>

<ColumnDefinition></ColumnDefinition>

<ColumnDefinition></ColumnDefinition>

<ColumnDefinition></ColumnDefinition>

<ColumnDefinition></ColumnDefinition>

<ColumnDefinition></ColumnDefinition>

<ColumnDefinition></ColumnDefinition>

<ColumnDefinition></ColumnDefinition>

<ColumnDefinition></ColumnDefinition>

<ColumnDefinition></ColumnDefinition>

<ColumnDefinition></ColumnDefinition>

<ColumnDefinition></ColumnDefinition>

<ColumnDefinition></ColumnDefinition>

<ColumnDefinition></ColumnDefinition>

<ColumnDefinition></ColumnDefinition>

<ColumnDefinition></ColumnDefinition>

<ColumnDefinition></ColumnDefinition>

<ColumnDefinition></ColumnDefinition>

<ColumnDefinition></ColumnDefinition>

<ColumnDefinition></ColumnDefinition>

<ColumnDefinition></ColumnDefinition>

<ColumnDefinition></ColumnDefinition>

<ColumnDefinition></ColumnDefinition>

<ColumnDefinition></ColumnDefinition>

</Grid.ColumnDefinitions>

<Grid.RowDefinitions>

<RowDefinition></RowDefinition>

<RowDefinition></RowDefinition>

</Grid.RowDefinitions>

<Button x:Name="beer28" Content="9.24" Height="90" Width="110" HorizontalAlignment="Center" VerticalAlignment="Top" Grid.Column="0" Grid.Row="0" FontSize="0.01" Click="beer28\_Click">

<Button.Background>

<ImageBrush ImageSource="пиво 2,8.png"/>

</Button.Background>

</Button>

<Label Content="{x:Static p:Resources.beer2\_8}" Grid.Row="1" Grid.Column="0" HorizontalAlignment="Center"/>

<Button x:Name="beer47033" Height="90" Width="110" HorizontalAlignment="Center" VerticalAlignment="Top" Grid.Column="1" Grid.Row="0" Click="beer47033\_Click">

<Button.Background>

<ImageBrush ImageSource="пиво 4,7.png"/>

</Button.Background>

</Button>

<Label Content="{x:Static p:Resources.beer4\_7\_033}" Grid.Row="1" Grid.Column="1" HorizontalAlignment="Center" FontFamily="Verdana"/>

<Button x:Name="beer4705" Height="90" Width="110" HorizontalAlignment="Center" VerticalAlignment="Top" Grid.Column="2" Grid.Row="0" Click="beer4705\_Click">

<Button.Background>

<ImageBrush ImageSource="пиво 4,7.png"/>

</Button.Background>

</Button>

<Label Content="{x:Static p:Resources.beer4\_7\_05}" Grid.Row="1" Grid.Column="2" HorizontalAlignment="Center" FontFamily="Verdana"/>

<Button x:Name="beer54033" Height="90" Width="110" HorizontalAlignment="Center" VerticalAlignment="Top" Grid.Column="3" Grid.Row="0" Click="beer54033\_Click">

<Button.Background>

<ImageBrush ImageSource="пиво 5,4.png"/>

</Button.Background>

</Button>

<Label Content="{x:Static p:Resources.beer5\_4\_033}" Grid.Row="1" Grid.Column="3" HorizontalAlignment="Center" FontFamily="Verdana"/>

<Button x:Name="beer5405" Height="90" Width="110" HorizontalAlignment="Center" VerticalAlignment="Top" Grid.Column="4" Grid.Row="0" Click="beer5405\_Click">

<Button.Background>

<ImageBrush ImageSource="пиво 5,4.png"/>

</Button.Background>

</Button>

<Label Content="{x:Static p:Resources.beer5\_4\_05}" Grid.Row="1" Grid.Column="4" HorizontalAlignment="Center" FontFamily="Verdana"/>

<Button x:Name="sidr47033" Height="90" Width="110" HorizontalAlignment="Center" VerticalAlignment="Top" Grid.Column="5" Grid.Row="0" Click="sidr47033\_Click">

<Button.Background>

<ImageBrush ImageSource="сидр 4,7.png"/>

</Button.Background>

</Button>

<Label Content="{x:Static p:Resources.cider4\_7\_033}" Grid.Row="1" Grid.Column="5" HorizontalAlignment="Center" FontFamily="Verdana"/>

<Button x:Name="sidr4705" Height="90" Width="110" HorizontalAlignment="Center" VerticalAlignment="Top" Grid.Column="6" Grid.Row="0" Click="sidr4705\_Click">

<Button.Background>

<ImageBrush ImageSource="сидр 4,7.png"/>

</Button.Background>

</Button>

<Label Content="{x:Static p:Resources.cider4\_7\_05}" Grid.Row="1" Grid.Column="6" HorizontalAlignment="Center" FontFamily="Verdana"/>

<Button x:Name="strongsidr58033" Height="90" Width="110" HorizontalAlignment="Center" VerticalAlignment="Top" Grid.Column="7" Grid.Row="0" Click="strongsidr58033\_Click">

<Button.Background>

<ImageBrush ImageSource="крепкий сидр .png"/>

</Button.Background>

</Button>

<Label Content="{x:Static p:Resources.strongcider5\_8\_033}" Grid.Row="1" Grid.Column="7" HorizontalAlignment="Center" FontFamily="Verdana"/>

<Button x:Name="strongsidr5805" Height="90" Width="110" HorizontalAlignment="Center" VerticalAlignment="Top" Grid.Column="8" Grid.Row="0" Click="strongsidr5805\_Click">

<Button.Background>

<ImageBrush ImageSource="крепкий сидр .png"/>

</Button.Background>

</Button>

<Label Content="{x:Static p:Resources.strongcider5\_8\_05}" Grid.Row="1" Grid.Column="8" HorizontalAlignment="Center" FontFamily="Verdana"/>

<Button x:Name="coctail55033" Height="90" Width="110" HorizontalAlignment="Center" VerticalAlignment="Top" Grid.Column="9" Grid.Row="0" Click="coctail55033\_Click">

<Button.Background>

<ImageBrush ImageSource="коктейль.png"/>

</Button.Background>

</Button>

<Label Content="{x:Static p:Resources.cocktail5\_5\_\_033}" Grid.Row="1" Grid.Column="9" HorizontalAlignment="Center" FontFamily="Verdana"/>

<Button x:Name="coctail5505" Height="90" Width="110" HorizontalAlignment="Center" VerticalAlignment="Top" Grid.Column="10" Grid.Row="0" Click="coctail5505\_Click">

<Button.Background>

<ImageBrush ImageSource="коктейль.png"/>

</Button.Background>

</Button>

<Label Content="{x:Static p:Resources.cocktail5\_5\_\_05}" Grid.Row="1" Grid.Column="10" HorizontalAlignment="Center" FontFamily="Verdana"/>

<Button x:Name="whitewine101512" Height="90" Width="110" HorizontalAlignment="Center" VerticalAlignment="Top" Grid.Column="11" Grid.Row="0" Click="whitewine101512\_Click">

<Button.Background>

<ImageBrush ImageSource="белое вино .png"/>

</Button.Background>

</Button>

<Label Content="{x:Static p:Resources.whitewine101512}" Grid.Row="1" Grid.Column="11" HorizontalAlignment="Center" FontFamily="Verdana"/>

<Button x:Name="whitewine101516" Height="90" Width="110" HorizontalAlignment="Center" VerticalAlignment="Top" Grid.Column="12" Grid.Row="0" Click="whitewine101516\_Click">

<Button.Background>

<ImageBrush ImageSource="белое вино16.png"/>

</Button.Background>

</Button>

<Label Content="{x:Static p:Resources.whitewine101516}" Grid.Row="1" Grid.Column="12" HorizontalAlignment="Center" FontFamily="Verdana"/>

<Button x:Name="whitewine101524" Height="90" Width="110" HorizontalAlignment="Center" VerticalAlignment="Top" Grid.Column="13" Grid.Row="0" Click="whitewine101524\_Click">

<Button.Background>

<ImageBrush ImageSource="белое вино24.png"/>

</Button.Background>

</Button>

<Label Content="{x:Static p:Resources.whitewine101524}" Grid.Row="1" Grid.Column="13" HorizontalAlignment="Center" FontFamily="Verdana"/>

<Button x:Name="whitewine1015075" Height="90" Width="110" HorizontalAlignment="Center" VerticalAlignment="Top" Grid.Column="14" Grid.Row="0" Click="whitewine1015075\_Click">

<Button.Background>

<ImageBrush ImageSource="белое вино75.png"/>

</Button.Background>

</Button>

<Label Content="{x:Static p:Resources.whitewine1015075}" Grid.Row="1" Grid.Column="14" HorizontalAlignment="Center" FontFamily="Verdana"/>

<Button x:Name="redwine101512" Height="90" Width="110" HorizontalAlignment="Center" VerticalAlignment="Top" Grid.Column="15" Grid.Row="0" Click="redwine101512\_Click">

<Button.Background>

<ImageBrush ImageSource="красное вино12.png"/>

</Button.Background>

</Button>

<Label Content="{x:Static p:Resources.redwine101512}" Grid.Row="1" Grid.Column="15" HorizontalAlignment="Center" FontFamily="Verdana"/>

<Button x:Name="redwine101516" Height="90" Width="110" HorizontalAlignment="Center" VerticalAlignment="Top" Grid.Column="16" Grid.Row="0" Click="redwine101516\_Click">

<Button.Background>

<ImageBrush ImageSource="красное вино 16.png"/>

</Button.Background>

</Button>

<Label Content="{x:Static p:Resources.redwine101516}" Grid.Row="1" Grid.Column="16" HorizontalAlignment="Center" FontFamily="Verdana"/>

<Button x:Name="redwine101524" Height="90" Width="110" HorizontalAlignment="Center" VerticalAlignment="Top" Grid.Column="17" Grid.Row="0" Click="redwine101524\_Click">

<Button.Background>

<ImageBrush ImageSource="красное вино 24.png"/>

</Button.Background>

</Button>

<Label Content="{x:Static p:Resources.redwine101524}" Grid.Row="1" Grid.Column="17" HorizontalAlignment="Center" FontFamily="Verdana"/>

<Button x:Name="redwine1015075" Height="90" Width="110" HorizontalAlignment="Center" VerticalAlignment="Top" Grid.Column="18" Grid.Row="0" Click="redwine1015075\_Click">

<Button.Background>

<ImageBrush ImageSource="красное вино75.png"/>

</Button.Background>

</Button>

<Label Content="{x:Static p:Resources.redwine1015075}" Grid.Row="1" Grid.Column="18" HorizontalAlignment="Center" FontFamily="Verdana"/>

<Button x:Name="champagne11510" Height="90" Width="110" HorizontalAlignment="Center" VerticalAlignment="Top" Grid.Column="19" Grid.Row="0" Click="champagne11510\_Click">

<Button.Background>

<ImageBrush ImageSource="шампанское.png"/>

</Button.Background>

</Button>

<Label Content="{x:Static p:Resources.champ115\_10}" Grid.Row="1" Grid.Column="19" HorizontalAlignment="Center" FontFamily="Verdana"/>

<Button x:Name="fortifiedwine16228" Height="90" Width="110" HorizontalAlignment="Center" VerticalAlignment="Top" Grid.Column="20" Grid.Row="0" Click="fortifiedwine16228\_Click">

<Button.Background>

<ImageBrush ImageSource="креплённое вино.png"/>

</Button.Background>

</Button>

<Label Content="{x:Static p:Resources.fortifiedwine16228}" Grid.Row="1" Grid.Column="20" HorizontalAlignment="Center" FontFamily="Verdana"/>

<Button x:Name="liquor164" Height="90" Width="110" HorizontalAlignment="Center" VerticalAlignment="Top" Grid.Column="21" Grid.Row="0" Click="liquor164\_Click">

<Button.Background>

<ImageBrush ImageSource="ликёр.png"/>

</Button.Background>

</Button>

<Label Content="{x:Static p:Resources.liqour164}" Grid.Row="1" Grid.Column="21" HorizontalAlignment="Center" FontFamily="Verdana"/>

<Button x:Name="cognac404" Height="90" Width="110" HorizontalAlignment="Center" VerticalAlignment="Top" Grid.Column="22" Grid.Row="0" Click="cognac404\_Click">

<Button.Background>

<ImageBrush ImageSource="коньяк.png"/>

</Button.Background>

</Button>

<Label Content="{x:Static p:Resources.cognac404}" Grid.Row="1" Grid.Column="22" HorizontalAlignment="Center" FontFamily="Verdana"/>

<Button x:Name="rum404" Height="90" Width="110" HorizontalAlignment="Center" VerticalAlignment="Top" Grid.Column="23" Grid.Row="0" Click="rum404\_Click">

<Button.Background>

<ImageBrush ImageSource="ром.png"/>

</Button.Background>

</Button>

<Label Content="{x:Static p:Resources.rum404}" Grid.Row="1" Grid.Column="23" HorizontalAlignment="Center" FontFamily="Verdana"/>

<Button x:Name="strongspirt404" Height="90" Width="110" HorizontalAlignment="Center" VerticalAlignment="Top" Grid.Column="24" Grid.Row="0" Click="strongspirt404\_Click">

<Button.Background>

<ImageBrush ImageSource="крепкое спиртное .png"/>

</Button.Background>

</Button>

<Label Content="{x:Static p:Resources.strongspirt404}" Grid.Row="1" Grid.Column="24" HorizontalAlignment="Center" FontFamily="Verdana"/>

<Button x:Name="whiskey404" Height="90" Width="110" HorizontalAlignment="Center" VerticalAlignment="Top" Grid.Column="25" Grid.Row="0" Click="whiskey404\_Click">

<Button.Background>

<ImageBrush ImageSource="виски.png"/>

</Button.Background>

</Button>

<Label Content="{x:Static p:Resources.whiskey404}" Grid.Row="1" Grid.Column="25" HorizontalAlignment="Center" FontFamily="Verdana"/>

</Grid>

</DockPanel>

</ScrollViewer>

<Label x:Name="ppmText" Content="{x:Static p:Resources.ppmText}" HorizontalAlignment="Left" Margin="63,100,0,0" VerticalAlignment="Top" Width="630" Height="28" FontFamily="Verdana" FontSize="16"/>

<Label Content="{x:Static p:Resources.ppmText1}" HorizontalAlignment="Left" Margin="63,129,0,0" VerticalAlignment="Top" FontFamily="Verdana" FontSize="16"/>

<Label x:Name="result" x:FieldModifier="public" Content="0,0" HorizontalAlignment="Left" Margin="484,100,0,0" VerticalAlignment="Top" Foreground="Red" Height="29" Width="47" FontFamily="Verdana" FontSize="16" RenderTransformOrigin="0.34,0.362"></Label>

<Button x:Name="back" Content="{x:Static p:Resources.back}" HorizontalAlignment="Left" Margin="749,220,0,0" VerticalAlignment="Top" Width="74" Click="Button\_Click\_1" FontFamily="Verdana"/>

<Button x:Name="show" Content="{x:Static p:Resources.show}" HorizontalAlignment="Left" Margin="848,220,0,0" VerticalAlignment="Top" Width="84" Click="Button\_Click\_3" FontFamily="Verdana"/>

<Label x:Name="hours" Content="0" HorizontalAlignment="Left" Foreground="Red" Margin="644,130,0,0" VerticalAlignment="Top" RenderTransformOrigin="0.5,0.538" FontSize="16" FontFamily="Verdana"/>

<Label Content="{x:Static p:Resources.ppmText2}" HorizontalAlignment="Left" Margin="63,159,0,0" VerticalAlignment="Top" FontFamily="Verdana" FontSize="16"/>

</Grid>

</Window>

public partial class upperBound : Window

{

public double upDoseMale = 0.0;//верхняя граница разового потребления для мужчин

public double upDoseFemale = 0.0;//верхняя граница разового потребления для женщин

//конструктор класса upperBound

public upperBound()

{

InitializeComponent();

}

//метод для возвращения в главное меню

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

{

StartMenu obj = new StartMenu();

obj.Show();

this.Close();

}

//метод который считает верхние границы разового потребления для мужчин и женщин

private void Weight\_ValueChanged(object sender, RoutedPropertyChangedEventArgs<double> e)

{

upDoseMale = e.NewValue \* 0.06;

upDoseFemale = e.NewValue \* 0.055;

}

//метод который проверит указали ли вы пол и выведет в label значение ВГРП

private void Button\_Click\_1(object sender, RoutedEventArgs e)

{

if (male.IsChecked == false && female.IsChecked == false && Title=="Верхняя граница разового употребления")

MessageBox.Show("Укажите ваш пол");

if (male.IsChecked == false && female.IsChecked == false && Title == "Upper limit of single use")

MessageBox.Show("Choose your sex");

if (male.IsChecked == true && female.IsChecked == false)

{

var str = string.Format("{0:0.#}", upDoseMale);

dose.Content = str;

}

if (male.IsChecked == false && female.IsChecked == true)

{

var str = string.Format("{0:0.#}", upDoseFemale);

dose.Content = str;

}

}

}

Приложение Б

(справочное)

Формы входных и выходных документов

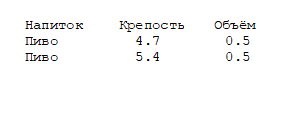


Рисунок Б.1

Приложение В

(справочное)

Результаты работы программы

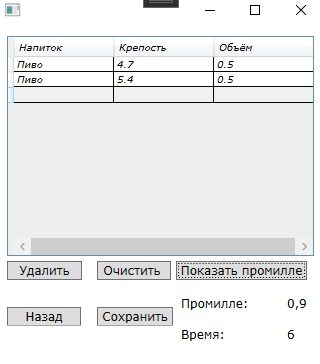


Рисунок В.1

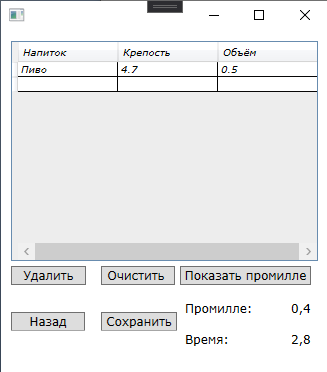


Рисунок В.2

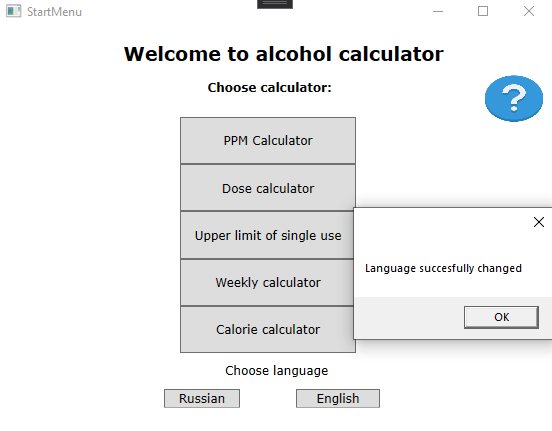


Рисунок В.3

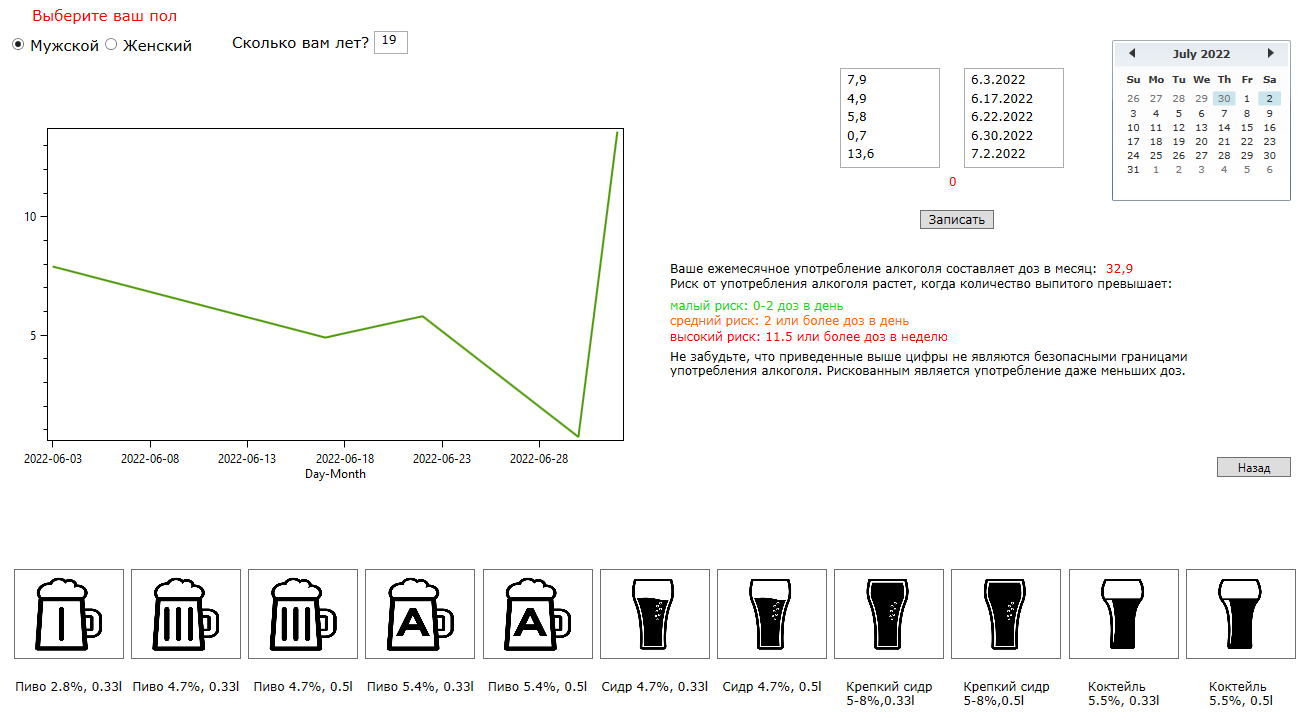


Рисунок В.4



Рисунок В.5

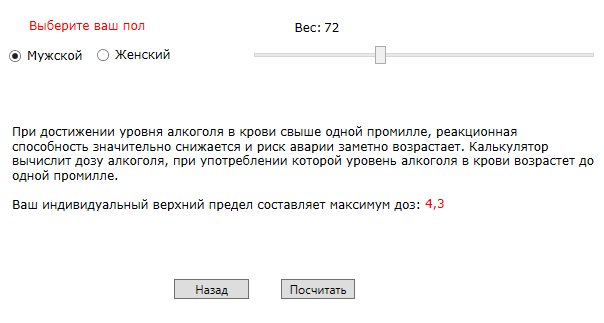


Рисунок В.6

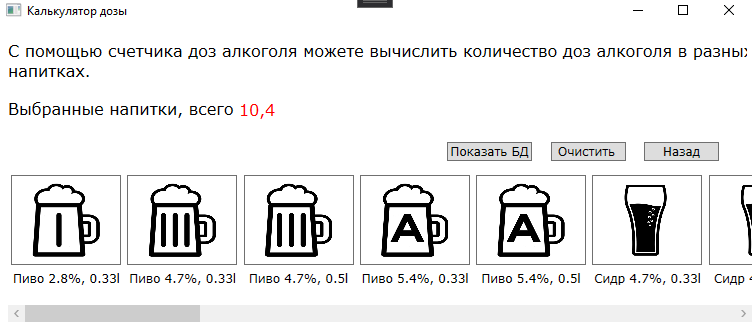


Рисунок В.7