

**Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Владимирский государственный университет
имени Александра Григорьевича и Николая Григорьевича Столетовых»
(ВлГУ)**

КАФЕДРА ФИЗИКИ И ПРИКЛАДНОЙ МАТЕМАТИКИ

**ОТЧЕТ
ПО УЧЕБНОЙ ПРАКТИКЕ
ПО МОДУЛЮ ПМ.03 «Участие в интеграции программных
модулей»**

Студента Ильиной Екатерины Евгеньевны
 Штурмина Евгения Андреевича

Колледжа инновационных технологий и предпринимательства

Специальности 09.02.03 "Программирование в компьютерных системах
"

Время прохождения практики

с «18» апреля 2019 г.

по «24» апреля 2019 г.

Руководитель от ВлГУ:

преподаватель КИТП

_____Лобова А.А

Руководитель от профильной организации:

преподаватель КИТП

_____Лобова А.А

Владимир 2020

УТВЕРЖДАЮ

Зав. кафедрой _____

« ____ » _____ 20 ____ г.

З А Д А Н И Е

на учебную практику по модулю ПМ.03 «Участие в интеграции программных модулей»
студента Ильиной Екатерины Евгеньевны, Штурмина Евгения Андреевича
4 курса, группы ПКсп-116 специальности 09.02.03"Программирование в компьютерных системах"
Предприятие Кафедра ФиПМ ВлГУ, г. Владимир

Последовательность прохождения практики получить задание, пройти инструктаж по технике безопасности и охране труда, выполнить все задания, оформить отчет согласно требованиям.

За время прохождения практики необходимо:

1. Изучить вопросы, предусмотренные программой по всем разделам.
2. На языке UML построить необходимые диаграммы для выбранного приложения.
3. Реализовать приложение «Калькулятор расчета суточной нормы калорий».
4. Развернуть клиентскую часть системы контроля версий SmartGit.
5. Поместить в репозиторий исходные файлы разработанного приложения.
6. Изучить принципы использования компилятора вне интегрированной среды разработки.
7. Разработать тест-кейсы, на их основании протестировать приложение.
8. Задание по стандартизации: оформить отчет по результатам практики согласно требованиям, к оформлению документации.
9. Задание по охране труда, технике безопасности и охране окружающей среды: изучить правила техники безопасности при работе за компьютером, вопросы охраны труда на месте прохождения практики.

Отчет по практике составить к 24.04.2019

Задание выдал: _____ преподаватель КИТП Лобова А.А

Задание получил: _____ Ильина Е.Е.

_____ Штурмин Е.А.

08.04.2020

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ	4
1 РАЗРАБОТКА ПРИЛОЖЕНИЯ.....	5
1.1. Описание предметной области.....	5
1.2. Проектирование приложения	9
1.3. Инструментальные средства разработки	11
1.4. Описание методов.....	12
2 КОМПИЛЯЦИЯ ВНЕ ИНТЕГРИРОВАННОЙ СРЕДЫ РАЗРАБОТКИ	15
3 СИСТЕМА КОНТРОЛЯ ВЕРСИЙ.....	17
4 ТЕСТИРОВАНИЕ.....	21
4.1. Тест-требования.....	21
4.2. Тест- план.....	21
4.3. Результаты тестирования.....	25
ЗАКЛЮЧЕНИЕ.....	33
СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ.....	34
ПРИЛОЖЕНИЕ А.....	35

ВВЕДЕНИЕ

Учебная практика является одной из составляющих учебного процесса и эффективной формой подготовки специалистов к трудовой деятельности. Данный вид практики предусмотрен учебным планом, определяет ее цели, задачи и форму. Так же учебная практика является важным элементом в системе практической подготовки студентов, она теснейшим образом связана со всеми другими формами обучения.

Данная учебная практика по модулю ПМ.03 «Участие в интеграции программных модулей» проходила в соответствии с учебным планом в ВлГУ с 18-го апреля по 24-ое апреля 2020-го года.

Во время прохождения учебной практики студент должен получить теоретические знания и закрепить их на практике.

В ходе практики студент должен выполнить:

- изучение вопросов, предусмотренных программой по всем разделам.
- построение на языке UML необходимых диаграмм для выбранного приложения.
- разработку приложения «Калькулятор расчета суточной нормы калорий».
- изучение принципов работы системы контроля версий.
- изучение принципов использования компилятора вне интегрированной среды разработки.
- разработать тест-кейсы и на их основании протестировать приложение.
- оформить отчет по результатам практики согласно требованиям, к оформлению документации.

1.1. Описание предметной области

Считать калорийность рациона, чтобы похудеть, решили еще в 20-х годах прошлого века, и такая система питания быстро завоевала популярность. О том, что кусок торта калорийнее яблока, сегодня знает даже тот, кто в принципе не представляет, что же за зверь такой – "ккал". "Калории – это такие маленькие существа, которые залезают ночью в твой шкаф и ушивают одежду", – шутят "в интернетах". Но в каждой шутке, как известно, – лишь доля шутки.

На самом деле, калория – от латинского calor, то есть "тепло" — это единица измерения родом из физики, и показывает она количество энергии, выделяемой в результате сгорания топлива. Для нашего организма таким топливом является пища, энергетическая ценность которой измеряется "большими" или килокалориями – ккал. Так, 1 попавший внутрь грамм белка или углевода даст нам 4 ккал – на физическую и умственную активность и просто для поддержания тепла и нормального функционирования сложносочиненных процессов в нашем организме, 1 грамм чистого жира (при 100%-ном усвоении) добавит "в актив" 9 ккал, ну а алкоголь находится где-то посередине – 1 грамм обернется 7 калориями. Чрезвычайно важные в работе организма витамины, минеральные соли и вода в расчетах энергетической ценности не учитываются, так как не "весят" ничего.

Первое, от чего стоит отталкиваться, если вам приглянулась система питания, основанная на подсчете калорий – показатель вашего индивидуального БОВ (базовый обмен веществ), иногда именуемый также ВОО (величина основного обмена). БОВ – это то количество калорий в сутки, которое необходимо организму для поддержания жизнедеятельности в состоянии покоя.

То есть тот объем энергии, которое наше тело ежедневно тратит на, казалось бы, несложные занятия, вроде дыхания, циркуляции крови, поддержания температуры, регенерации клеток и пр. БОВ зависит от возраста (наверняка еще в детстве бабушка говорила вам, что растущему организму требуется больше калорий, и была права), пола (мужчинам требуется больше ккал, чем женщинам), роста и веса (чем выше эти показатели, тем больше энергии необходимо организму на обмен веществ), а также от мышечной массы (мышцы, как известно, сжигают больше калорий, именно поэтому заниматься спортом уже сегодня – все равно что инвестировать в завтрашний день, когда мускулатура начнет сама "работать на вас").

Формул расчета БОВ существует великое множество – версия ВОЗ (Всемирной Организации Здравоохранения), формула Харриса-Бенедикта, Кетч-МакКардл, Тома Венуто и пр. — и результаты большинства из них отличаются не более, чем на 50-100 ккал. Наиболее популярной и даже получившей признание Американской Диетической Ассоциации стала формула Маффина-Джеора (Миффлина-Сан Жеора). В своём приложении расчёт будет производиться по формулам Харриса-Бенедикта и Миффлина-Сан Жеора.

1.1.1. Формула Харриса-Бенедикта.

Формула Харриса-Бенедикта поможет понять, сколько калорий вам нужно ежедневно для похудения и поддержания веса.

Все люди отличаются друг от друга и у всех организмы обладают индивидуальными особенностями, поэтому для каждого человека нужно свое количество калорий в день для похудения или поддержания веса. Для определения своей нормы калорий в день рекомендуем использовать формулу Харриса-Бенедикта. [1]

Средние показатели суточной нормы калорий различаются в разных странах, но общие показатели таковы: для сильного пола 2500-2700 ккал в день, для женщин – 2000-2200 ккал в день. Всем известно, что иногда и 100 калорий сверх нормы могут стать причиной появления лишнего веса и жировых отложений, а в этих показателях разброс в 200 ккал. Именно поэтому лучше

всего рассчитывать индивидуальную норму калорий при помощи формулы Харриса-Бенедикта.

Первый раз о формуле Харриса-Бенедикта написали в 1919 году Джеймсом Артуром Харрисом и Фрэнсисом Гано Бенедиктом. Работа была опубликована Институтом Карнеги в Вашингтоне и называлась «Биометрические исследования основного обмена человека».

Харрис и Бенедикт показали, что число калорий, нужных человеку каждый день, изменяется от таких показателей: базального метаболизма (BMR) и активного метаболизма (AMR). [1]

В 1984 г была обнародована формула для определения базального метаболизма - это исправленные вычисления, более точно учитывающие возраст. И выглядят эти формулы так:

Базальный метаболизм BMR для женщин:

$$BMR = 447,593 + (9,247 * \text{вес в кг}) + (3,098 * \text{рост в см}) - (4,330 * \text{возраст в годах}).$$

Базальный метаболизм BMR для мужчин:

$$BMR = 88,362 + (13,397 * \text{вес в кг}) + (4,799 * \text{рост в см}) - (5,677 * \text{возраст в годах}).$$

1.1.2 Формула Миффлина-Сан Жеора.

Формула Миффлина-Сан Жеора – это одна из самых последних формул расчета калорий для оптимального похудения или сохранения нормального веса. Она была выведена в 2005 году и все чаще стала заменять классическую формулу Харриса-Бенедикта.

Формула Миффлина-Сан Жеора, разработанная группой американских врачей-диетологов под руководством докторов Миффлина и Сан Жеора, существует в двух вариантах – упрощенном и доработанном и выдает

необходимое количество килокалорий (ккал) в сутки для каждого конкретного человека. [2]

Упрощенный вариант формулы Миффлина-Сан Жеора:

– для мужчин:

$$\text{BMR} = 10 \times \text{вес (кг)} + 6,25 \times \text{рост (см)} - 5 \times \text{возраст (г)} + 5;$$

– для женщин:

$$\text{BMR} = 10 \times \text{вес (кг)} + 6,25 \times \text{рост (см)} - 5 \times \text{возраст (г)} - 161.$$

При расчете количества калорий на день большинство формул учитывают средний уровень активности. На основании выбранного уровня активности к базовому уровню обмена веществ применяется коэффициент. Таблица расчёта коэффициента представлена в таблице 1.1. [3]

Таблица 1.1 – Таблица коэффициентов

Тип	Коэффициент	Активность	Описание
Без физической нагрузки (сидячая работа)	1.2	Отсутствие нагрузки	Отсутствие физической нагрузки или небольшая легкая нагрузка
Физическая нагрузка 3 раза в неделю	1.375	Очень низкий уровень активности	Интенсивные упражнения не менее 20 минут от 1 до 3 раз в неделю. Например, езда на велосипеде, бег трусцой, баскетбол, и т.д. Или если часто приходится ходить в течение длительных периодов времени при отсутствии тренировок.
Физическая нагрузка 5 раз в неделю	1.46	Низкий уровень активности	Выполнение легких упражнений не менее 20 минут от 3 до 5 раз в неделю
Физическая нагрузка 5 раз в неделю (интенсивно)	1.55	Умеренный уровень активности	Интенсивные упражнения от 30 до 60 минут от 3 до 5 раз в неделю
Ежедневная физическая нагрузка	1.64	Средний уровень активности	Выполнение упражнений не менее 60 минут от 5 до 7 дней в неделю

Продолжение таблицы 1.1

Тип	Коэффициент	Активность	Описание
Ежедневная интенсивная физическая нагрузка	1.72	Высокий уровень активности	Выполнение интенсивных упражнений не менее 60 минут от 5 до 7 дней в неделю или занятия 2 раза в день. Также сюда относятся профессии, связанные с физической работой - строительные работы, сельское хозяйство, ландшафтные работы.
Ежедневная физическая нагрузка + физическая работа	1.9	Экстремальный уровень	7 или более часов в неделю очень интенсивных упражнений или интенсивные занятия 2 раза в день. Очень интенсивные и/или очень затратные виды деятельности. .

Для завершения подсчёта суточной нормы калорий умножаем 2 величины:

BMR (базальный метаболизм) * AMR (активный метаболизм).

1.2. Проектирование приложения

UML – это унифицированный графический язык моделирования для описания, визуализации, проектирования и документирования ОО систем. UML призван поддерживать процесс моделирования ПС на основе ОО подхода, организовывать взаимосвязь концептуальных и программных понятий, отражать проблемы масштабирования сложных систем. Модели на UML используются на всех этапах жизненного цикла ПС, начиная с бизнес-анализа и заканчивая сопровождением системы. Разные организации могут применять UML по своему усмотрению в зависимости от своих проблемных областей и используемых технологий. Любой язык состоит из словаря и правил комбинирования слов для получения осмысленных конструкций. Так, в частности, устроены языки программирования, таковым является и UML. Отличительной его особенностью

является то, что словарь языка образуют графические элементы. Каждому графическому символу соответствует конкретная семантика, поэтому модель, созданная одним разработчиком, может однозначно быть понята другим, а также программным средством, интерпретирующим UML. Отсюда, в частности, следует, что модель ПС, представленная на UML, может автоматически быть переведена на ОО язык программирования (такой, как Java, C++, VisualBasic), то есть, при наличии хорошего инструментального средства визуального моделирования, поддерживающего UML, построив модель, мы получим и заготовку программного кода, соответствующего этой модели. [4]

Диаграмма активностей (видов деятельности) - один из доступных видов диаграмм, поддерживаемых Flexberry Designer. Она, как и диаграмма состояний, отражает динамические аспекты поведения системы. По существу, эта диаграмма представляет собой блок-схему, которая наглядно показывает, как поток управления переходит от одной деятельности к другой. Диаграмма активностей, разработанная для приложения, представлена на рисунке 1.1.

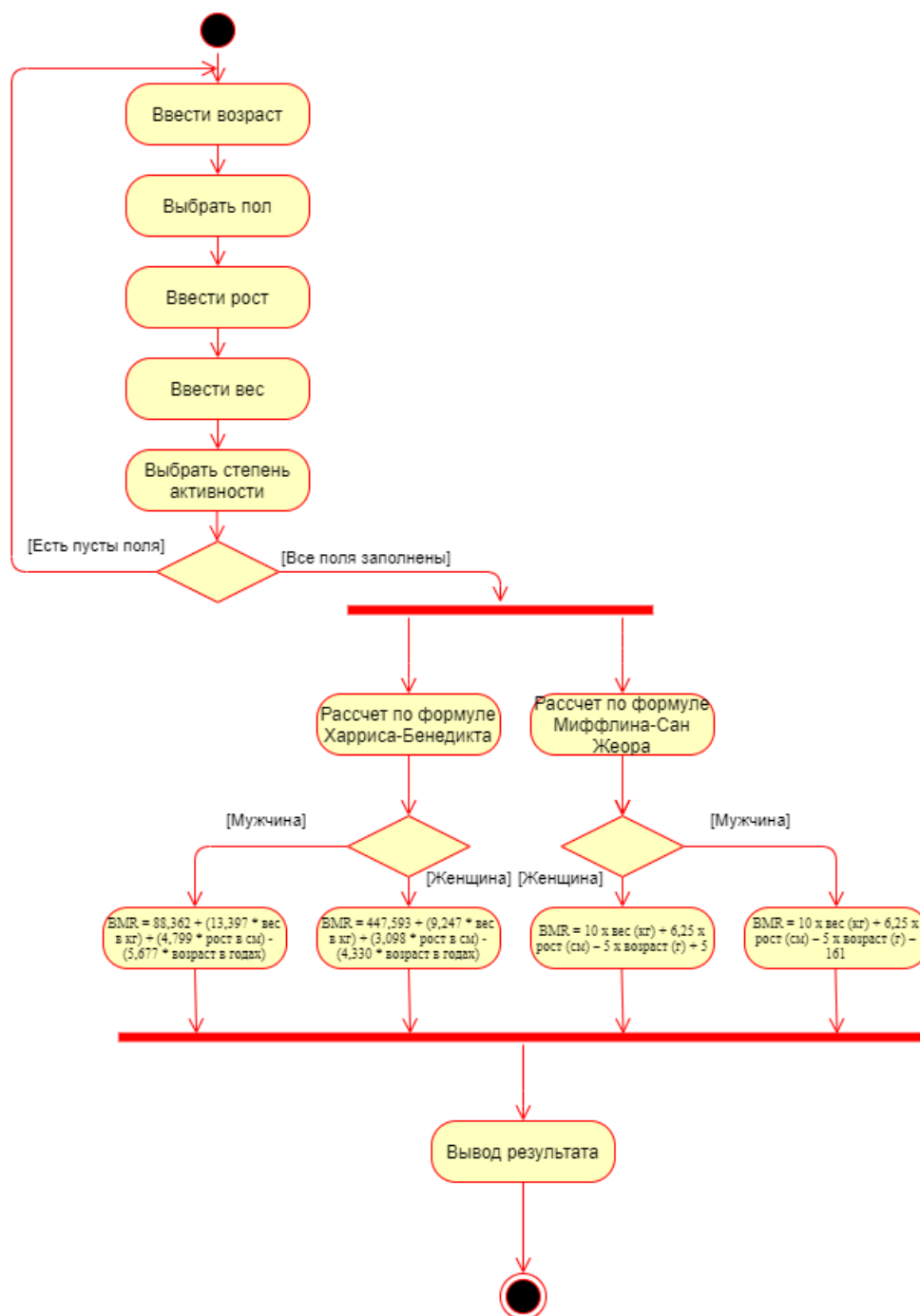


Рисунок 1.1 – Диаграмма активностей

1.3. Инструментальные средства разработки

В качестве интегрированной среды разработки выбрана Visual Studio. MS Visual Studio 2017 Community – многофункциональная интегрированная среда разработки, предназначенная для программирования современных приложений для платформы Windows. Доступны функции повышающие производительность, инструменты кроссплатформенной мобильной разработки и набор расширений из Visual Studio Gallery.

Community — редакция известной среды разработки Microsoft Visual Studio 2017, распространяемая бесплатно. Она отличается от платных вариантов Professional и Enterprise тем, что рассчитана исключительно на индивидуальных разработчиков.

Последняя версия Community предлагает большое количество улучшений, которые сильно расширили возможности программного продукта. Особенности программы:

- Упрощенная установка;
- Функциональные инструменты программирования, разрешающие обнаружить и успешно ликвидировать неточности в коде, проводить рефакторинг;
- Модернизированная отладка, во время которой проводится выявление проблем производительности;
- Веб-инструменты ASP.NET, Node.js, Python и JavaScript, необходимые для создания веб-приложений.;
- Несколько поддерживаемых языков программирования, среди которых C#, Visual Basic, F#, JavaScript, C++, TypeScript, Python, возможность включить поддержку новых языков;
- Доступ к бесплатным инструментам, а также обучающим программам Xamarin University, Pluralsight и прочим. [5]

1.4. Описание методов

Разрабатываемое приложение предназначено для подсчёта суточной нормы калорий. Для разработки приложения нам необходимы следующие методы:

```
buttonEnter_Click () { }
```

Назначение: метод, обрабатывающий клик по ячейке.

Входные значения: нет.

Выходные значения: нет.

Содержание: метод считывает все выбранные и введенные значения, далее передает их в методы FormulaMifflin и FormulaHarries, данные методы будут описаны ниже. Далее происходит заполнение полей, отвечающих за результат.

```
public int FormulaHarries (bool gender, double factorActive, int age, double weight, double height){ }
```

Назначение: расчёт нормы калорий по формуле Миффлина-Сан Жеора.

Входные значения: пол, коэффициент активности, возраст, вес, рост.

Выходные значения: суточная норма калорий.

Содержание: с помощью условного оператора if, происходит проверка пришедшего пола. Если пришло значение true, происходит расчет по формуле для мужчин, иначе по формуле для женщин.

```
public int Харриса-Бенедикта (bool gender, double factorActive, int age, double weight, double height){ }
```

Назначение: расчёт нормы калорий по формуле Харриса-Бенедикта.

Входные значения: пол, коэффициент активности, возраст, вес, рост.

Выходные значения: суточная норма калорий.

Содержание: с помощью условного оператора if, происходит проверка пришедшего пола. Если пришло значение true, происходит расчет по формуле для мужчин, иначе по формуле для женщин.

```
private double Active (int id){ }
```

Назначение: расчет коэффициента активности

Входные значения: индекс выбранного элемента в ComboBox.

Выходные значения: коэффициент активности.

Содержание: с помощью конструкции switch case, соответствующему индексу присваивается коэффициент.

```
private void textBoxAge_KeyPress(){} 
```

Назначение: контроль введенных значений.

Входные значения: нет.

Выходные значения: нет.

Содержание: ввод в TextBox только чисел.

Программный код приложения про расчёту суточной нормы калорий представлен в приложении А.

2 КОМПИЛЯЦИЯ ВНЕ ИНТЕГРИРОВАННОЙ СРЕДЫ РАЗРАБОТКИ

Команда `dotnet build` выполняет сборку проекта и его зависимостей в набор двоичных файлов. Эти двоичные файлы содержат код проекта в виде файлов на промежуточном языке (IL) с расширением `DLL`. В зависимости от типа и параметров проекта могут быть включены другие файлы, например:

Исполняемый файл, который может использоваться для запуска приложения, если тип проекта является исполняемым файлом, предназначенным для .NET Core 3.0 или более поздней версии.

Файлы символов, используемые для отладки с помощью расширения `PDB`.

Файл `.deps.json`, в котором перечислены зависимости приложения или библиотеки.

Файл `.runtimeconfig.json`, определяющий общую среду выполнения и ее версию для приложения.

Другие библиотеки, от которых зависит проект (через ссылки на проект или ссылки на пакеты NuGet).[6]

`dotnet build` использует MSBuild для сборки проекта, поддерживая при этом как параллельные, так и инкрементные сборки. Дополнительные сведения см. в разделе Инкрементные сборки.

Помимо своих параметров, команда `dotnet build` принимает и параметры MSBuild, например `-p` для задания свойств или `-l` для определения средства ведения журнала. Дополнительные сведения об этих параметрах см. в справочнике по командной строке MSBuild. Либо же вы можете использовать команду `dotnet msbuild`.

Выполнение `dotnet build` эквивалентно выполнению `dotnet msbuild -restore`; однако уровень детализации выходных данных по умолчанию отличается. [6]

Компиляция с помощью `dotnet build` представлена на рисунках 2.1 – 2.3.

```
C:\Users\User>cd /d E:\TrainingPractice\TrainingPractice
E:\TrainingPractice\TrainingPractice>
```

Рисунок 2.1 – Изменение пути к папке с проектом

```
E:\TrainingPractice\TrainingPractice>dir
Том в устройстве E имеет метку Диск
Серийный номер тома: 826A-E89F

Содержимое папки E:\TrainingPractice\TrainingPractice

23.04.2020  19:05    <DIR>      .
23.04.2020  19:05    <DIR>      ..
23.04.2020  17:34      184 App.config
23.04.2020  19:04    <DIR>      bin
20.04.2020  18:42        3 891 Form1.cs
20.04.2020  03:02       16 564 Form1.Designer.cs
20.04.2020  03:02      107 589 Form1.resx
19.04.2020  15:20       67 646 nutrition_icon_136342.ico
23.04.2020  19:05    <DIR>      obj
19.04.2020  15:14      563 Program.cs
23.04.2020  19:05    <DIR>      Properties
23.04.2020  18:15       4 660 TrainingPractice.csproj
                7 файлов          201 097 байт
                5 папок    289 891 573 760 байт свободно
```

Рисунок 2.2 – Отображение всех файлов и подкаталогов

```
E:\TrainingPractice\TrainingPractice>dotnet build
Microsoft (R) Build Engine версии 16.5.0+d4cbfca49 для .NET Core
(C) Корпорация Майкрософт (Microsoft Corporation). Все права защищены.

Работы нет. Ни в одном из указанных проектов нет пакетов, которые можно было бы восстановить.
TrainingPractice -> E:\TrainingPractice\TrainingPractice\bin\Debug\TrainingPractice.exe

Сборка успешно завершена.

Предупреждений: 0
Ошибок: 0

Прошло времени 00:00:03.26
```

Рисунок 2.3 – Выполнение сборки, с помощью dotnet build

3 СИСТЕМА КОНТРОЛЯ ВЕРСИЙ

Система контроля версий — это система, записывающая изменения в файл или набор файлов в течение времени и позволяющая вернуться позже к определённой версии. Для контроля версий файлов в этой книге в качестве примера будет использоваться исходный код программного обеспечения, хотя на самом деле вы можете использовать контроль версий практически для любых типов файлов.

Если вы графический или web-дизайнер и хотите сохранить каждую версию изображения или макета (скорее всего, захотите), система контроля версий (далее СКВ) — как раз то, что нужно. Она позволяет вернуть файлы к состоянию, в котором они были до изменений, вернуть проект к исходному состоянию, увидеть изменения, увидеть, кто последний менял что-то и вызвал проблему, кто поставил задачу и когда и многое другое. Использование СКВ также значит в целом, что, если вы сломали что-то или потеряли файлы, вы спокойно можете всё исправить. В дополнение ко всему вы получите всё это без каких-либо дополнительных усилий. [7]

Для работы с СКВ GIT, необходимо создать учётную запись на github.com. Пример созданной учётной записи представлен на рисунке 3.1.

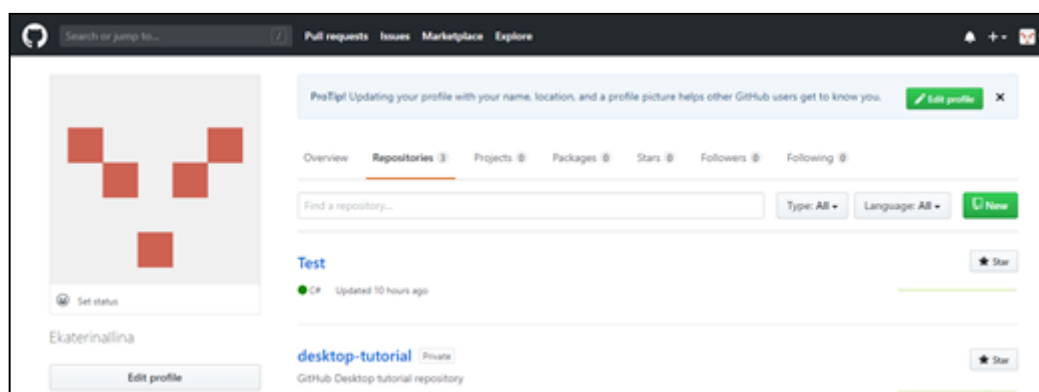
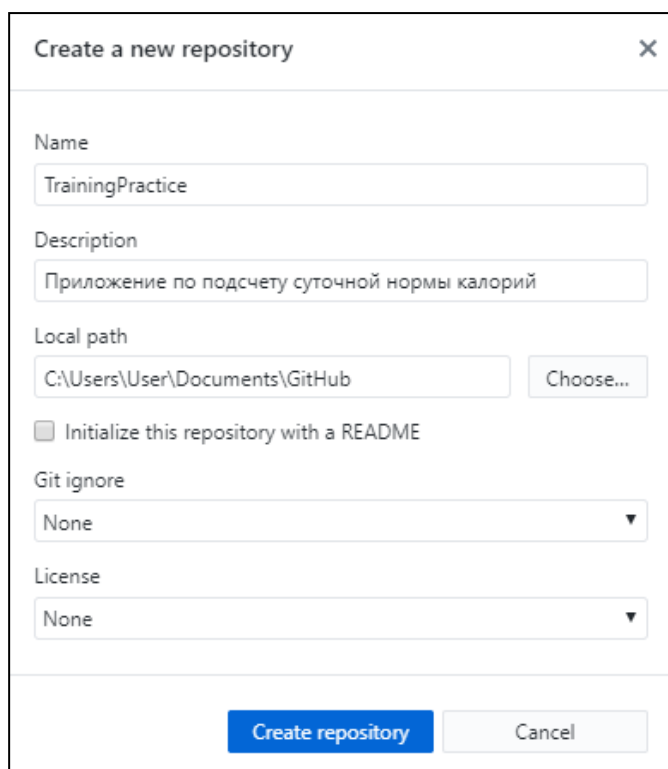


Рисунок 3.1 – Учетная запись на github.com

Для более удобной работы с репозиторием было установлено клиентское приложение GitHubDesktop. Создание репозитория представлено на рисунке 3.2.



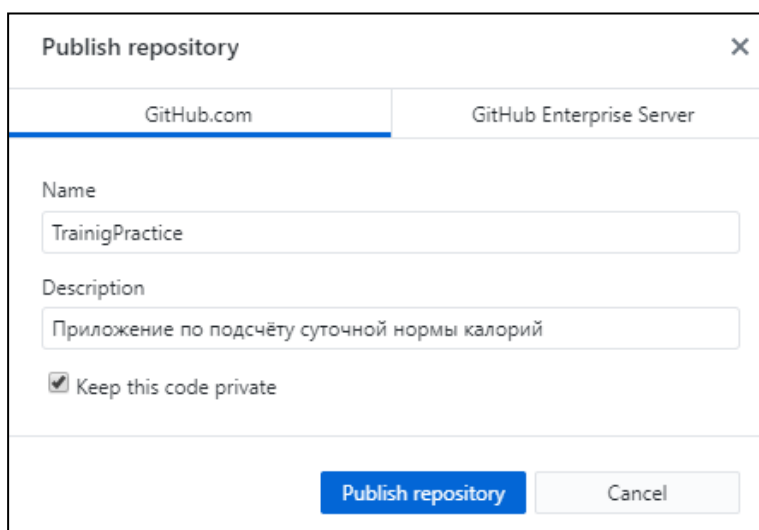
The screenshot shows a dialog box titled "Create a new repository" with a close button (X) in the top right corner. The dialog contains several input fields and checkboxes:

- Name:** A text input field containing "TrainingPractice".
- Description:** A text input field containing "Приложение по подсчету суточной нормы калорий".
- Local path:** A text input field containing "C:\Users\User\Documents\GitHub" and a "Choose..." button to the right.
- Initialize this repository with a README:** An unchecked checkbox.
- Git ignore:** A dropdown menu with "None" selected.
- License:** A dropdown menu with "None" selected.

At the bottom of the dialog, there are two buttons: "Create repository" (highlighted in blue) and "Cancel".

Рисунок 3.2 – Создание нового репозитория

Что бы другие пользователи могли видеть содержимое репозитория, его необходимо опубликовать. Процесс публикации представлен на рисунке 3.3.



The screenshot shows a dialog box titled "Publish repository" with a close button (X) in the top right corner. The dialog has two tabs: "GitHub.com" (selected) and "GitHub Enterprise Server". The "Name" and "Description" fields are identical to the previous dialog. There is an additional checkbox:

- Keep this code private:** A checked checkbox.

At the bottom of the dialog, there are two buttons: "Publish repository" (highlighted in blue) and "Cancel".

Рисунок 3.3 – Публикация репозитория

Так как репозиторий после создания пустой, в него необходимо добавить файлы проекта. Занесение данных в репозиторий представлено на рисунке 3.4. Процесс записи изменений в репозитории, выполняется с помощью команды Commit, изображенной на рисунке 3.5.

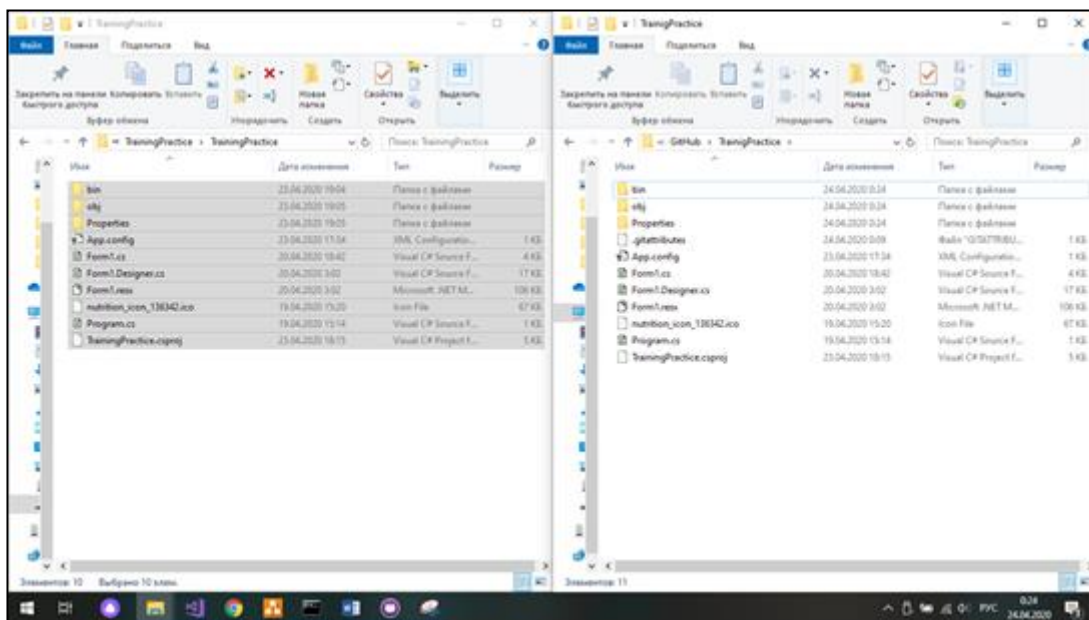


Рисунок 3.4 – Копирование данных из проекта в репозиторий

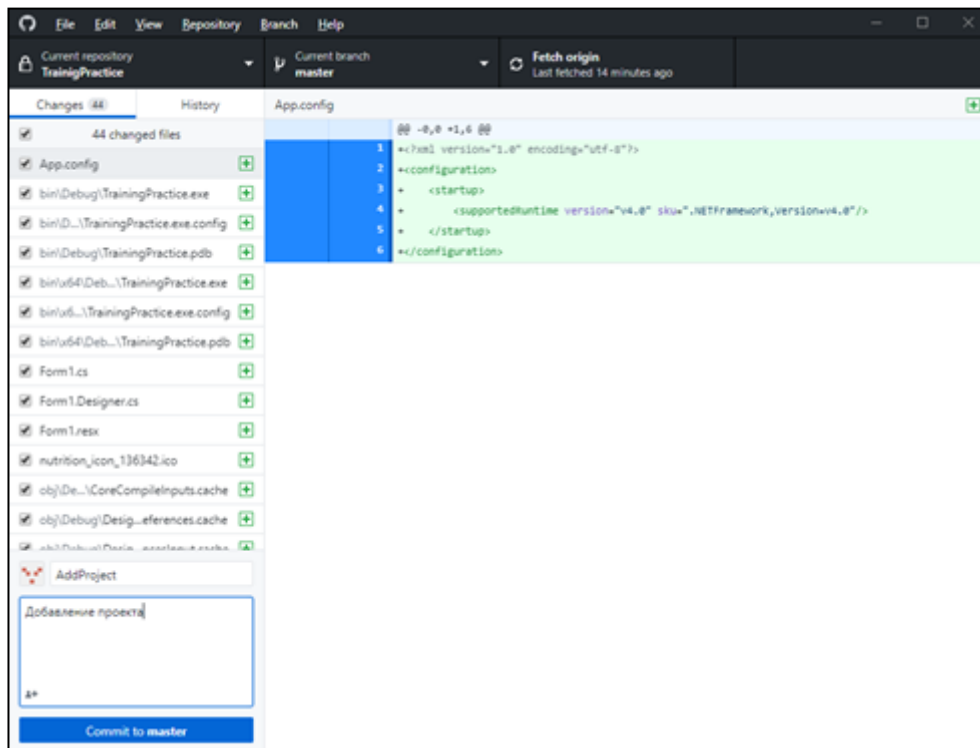


Рисунок 3.5 – Запись изменений в репозитории

Сохранение всех изменений, которые были сделаны в локальном репозитории на GitHub, производится при помощи Push commit, данный процесс изображен на рисунке 3.6. Содержимое репозитория после Push commit, представлено на рисунке 3.7.

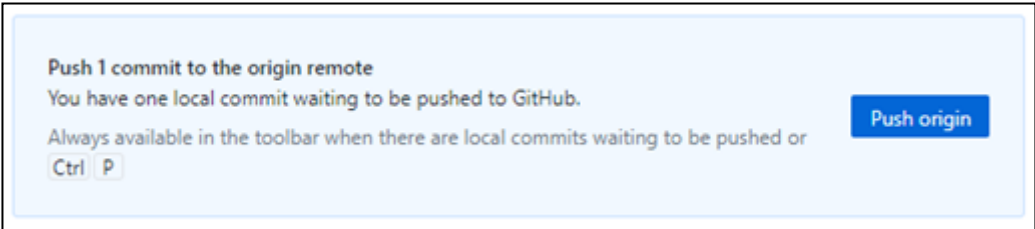


Рисунок 3.6 – Push commit

Ekaterinalina AddProject		Latest commit 5335ebf 11 minutes ago
Properties	AddProject	11 minutes ago
bin	AddProject	11 minutes ago
obj	AddProject	11 minutes ago
.gitattributes	Initial commit	33 minutes ago
App.config	AddProject	11 minutes ago
Form1.Designer.cs	AddProject	11 minutes ago
Form1.cs	AddProject	11 minutes ago
Form1.resx	AddProject	11 minutes ago
Program.cs	AddProject	11 minutes ago
TrainingPractice.csproj	AddProject	11 minutes ago
nutrition_icon_136342.ico	AddProject	11 minutes ago

Рисунок 3.7 – Успешное добавление файлов

4 ТЕСТИРОВАНИЕ

4.1. Тест-требования

- 1) Проверить результат работы программы при введении разных параметров возраста.
- 2) Проверить результат работы программы при изменении пола.
- 3) Проверить результат работы программы при введении разных параметров роста.
- 4) Проверить результат работы программы при введении разных параметров веса.
- 5) Проверить результат работы программы при выборе разного показателя активности.
- 6) Проверить результат работы программы с незаполненными полями

4.2. Тест- план

На основе данных тест-требований был разработан тест-план, состоящий из 6 тестов.

Тестовый пример 1.

Требования: 1.

Описание теста: Проверить результат работы программы при введении разных параметров возраста.

Входные данные:

1) возраст – 20, пол – мужской, вес – 80, рост – 170, активность – занятия фитнесом 3 раза в неделю.

2) возраст – 30, пол – мужской, вес – 80, рост – 170, активность – занятия фитнесом 3 раза в неделю.

Ожидаемые выходные данные: изменение суточной нормы калорий.

Сценарий:

1) Запускаем программу.

2) Заполняем поля данными: возраст – 20, пол – мужской, вес – 80, рост – 170, активность – занятия фитнесом 3 раза в неделю.

3) Нажимаем кнопку «Рассчитать».

4) Меняем данные: возраст – 30, пол – мужской, вес – 80, рост – 170, активность – занятия фитнесом 3 раза в неделю.

5) Нажимаем кнопку «Рассчитать».

Тестовый пример 2.

Требования: 2.

Описание теста: Проверить результат работы программы при изменении пола.

Входные данные:

1) возраст – 20, пол – женский, вес – 80, рост – 170, активность – занятия фитнесом 3 раза в неделю.

2) возраст – 20, пол – мужской, вес – 80, рост – 170, активность – занятия фитнесом 3 раза в неделю.

Ожидаемые выходные данные: изменение суточной нормы калорий.

Сценарий:

1) Запускаем программу.

2) Заполняем поля данными: возраст – 20, пол – женский, вес – 80, рост – 170, активность – занятия фитнесом 3 раза в неделю.

3) Нажимаем кнопку «Рассчитать».

4) Меняем данные: возраст – 20, пол – мужской, вес – 80, рост – 170, активность – занятия фитнесом 3 раза в неделю.

5) Нажимаем кнопку «Рассчитать».

Тестовый пример 3.

Требования: 3.

Описание теста: Проверить результат работы программы при введении разных параметров роста.

Входные данные:

1) возраст – 20, пол – мужской, вес – 80, рост – 130, активность – занятия фитнесом 3 раза в неделю.

2) возраст – 20, пол – мужской, вес – 80, рост – 170, активность – занятия фитнесом 3 раза в неделю.

Ожидаемые выходные данные: изменение суточной нормы калорий.

Сценарий:

1) Запускаем программу.

2) Заполняем поля данными: возраст – 20, пол – мужской, вес – 80, рост – 130, активность – занятия фитнесом 3 раза в неделю.

3) Нажимаем кнопку «Рассчитать».

4) Меняем данные: возраст – 20, пол – мужской, вес – 80, рост – 170, активность – занятия фитнесом 3 раза в неделю.

5) Нажимаем кнопку «Рассчитать».

Тестовый пример 4.

Требования: 4.

Описание теста: Проверить результат работы программы при введении разных параметров веса.

Входные данные:

1) возраст – 20, пол – женский, вес – 45, рост – 165, активность – занятия фитнесом 3 раза в неделю.

2) возраст – 20, пол – женский, вес – 60, рост – 165, активность – занятия фитнесом 3 раза в неделю.

Ожидаемые выходные данные: изменение суточной нормы калорий.

Сценарий:

1) Запускаем программу.

2) Заполняем поля данными: возраст – 20, пол – женский, вес – 45, рост – 165, активность – занятия фитнесом 3 раза в неделю.

3) Нажимаем кнопку «Рассчитать».

4) Меняем данные: возраст – 20, пол – женский, вес – 60, рост – 165, активность – занятия фитнесом 3 раза в неделю.

5) Нажимаем кнопку «Рассчитать».

Тестовый пример 5.

Требования: 5.

Описание теста: Проверить результат работы программы при выборе разного показателя активности.

Входные данные:

1) возраст – 30, пол – мужской, вес – 80, рост – 170, активность – занятия фитнесом 3 раза в неделю.

2) возраст – 30, пол – мужской, вес – 80, рост – 170, активность – ежедневная физическая нагрузка плюс физическая работа.

Ожидаемые выходные данные: изменение суточной нормы калорий.

Сценарий:

1) Запускаем программу.

2) Заполняем поля данными: возраст – 30, пол – мужской, вес – 80, рост – 170, активность – занятия фитнесом 3 раза в неделю.

3) Нажимаем кнопку «Рассчитать».

4) Меняем данные: возраст – 30, пол – мужской, вес – 80, рост – 170, активность – ежедневная физическая нагрузка плюс физическая работа.

5) Нажимаем кнопку «Рассчитать».

Тестовый пример 6.

Требования: 6.

Описание теста: Проверить результат работы программы с незаполненными полями.

Входные данные: пол – мужской, вес – 80, рост– 170, активность – занятия фитнесом 3 раза в неделю.

Ожидаемые выходные данные: Выводиться сообщение «Заполните все поля»

Сценарий:

- 1) Запускаем программу.
- 2) Заполняем поля данными: пол – мужской, вес – 80, рост– 170, активность – занятия фитнесом 3 раза в неделю.
- 3) Нажимаем кнопку «Рассчитать».

4.3. Результаты тестирования

Тестовый пример 1.

Специалист по тестированию: Штурмин Евгений Андреевич.

Дата и время тестирования: 21 апреля 2020, 20:30.

Входные данные:

- 1) Запускаем программу.
- 2) Заполняем поля данными: возраст – 20, пол – мужской, вес – 80, рост– 170, активность – занятия фитнесом 3 раза в неделю.
- 3) Нажимаем кнопку «Рассчитать».
- 4) Меняем данные: возраст – 30, пол – мужской, вес – 80, рост – 170, активность – занятия фитнесом 3 раза в неделю.
- 5) Нажимаем кнопку «Рассчитать».

Полученные выходные данные:

Рассчёт суточной нормы калорий

Возраст: 20
Пол: ☒ М ☐ Ж
Вес: 80
Рост: 170

Активность: занятия фитнесом 3 раза в неделю

Рассчитать

Суточная норма калорий:

	Для похудения	Не изменяя вес	Для набора массы
по формуле Миффлина-Сан Жеора	2070	2430	2790
по формуле Харриса-Бенедикта	2186	2561	2936

Рисунок 4.1 – Результат работы программы при вводе первого варианта заполнения

Рассчёт суточной нормы калорий

Возраст: 30
Пол: ☒ М ☐ Ж
Вес: 80
Рост: 170

Активность: занятия фитнесом 3 раза в неделю

Рассчитать

Суточная норма калорий:

	Для похудения	Не изменяя вес	Для набора массы
по формуле Миффлина-Сан Жеора	2017	2362	2707
по формуле Харриса-Бенедикта	2123	2483	2843

Рисунок 4.2 – Результат работы программы при вводе второго варианта заполнения

Тестовый пример 2.

Специалист по тестированию: Штурмин Евгений Андреевич.

Дата и время тестирования: 21 апреля 2020, 20:35.

Входные данные:

1) Запускаем программу.

2) Заполняем поля данными: возраст – 20, пол – женский, вес – 80, рост – 170, активность – занятия фитнесом 3 раза в неделю.

3) Нажимаем кнопку «Рассчитать».

4) Меняем данные: возраст – 20, пол – мужской, вес – 80, рост – 170, активность – занятия фитнесом 3 раза в неделю.

5) Нажимаем кнопку «Рассчитать».

Полученные выходные данные:

Рассчёт суточной нормы калорий

Возраст: 20, Пол: ☐ М ☒ Ж, Активность: занятия фитнесом 3 раза в неделю, Вес: 80, Рост: 170

Рассчитать

Суточная норма калорий:

	Для похудения	Не изменяя вес	Для набора массы
по формуле Миффлина-Сан Жеора	1872	2202	2532
по формуле Харриса-Бенедикта	1908	2238	2568

Рисунок 4.3 – Результат работы программы при вводе первого варианта заполнения

Рассчёт суточной нормы калорий

Возраст: 20, Пол: ☒ М ☐ Ж, Активность: занятия фитнесом 3 раза в неделю, Вес: 80, Рост: 170

Рассчитать

Суточная норма калорий:

	Для похудения	Не изменяя вес	Для набора массы
по формуле Миффлина-Сан Жеора	2070	2430	2790
по формуле Харриса-Бенедикта	2186	2561	2936

Рисунок 4.4 – Результат работы программы при вводе второго варианта заполнения

Тестовый пример 3.

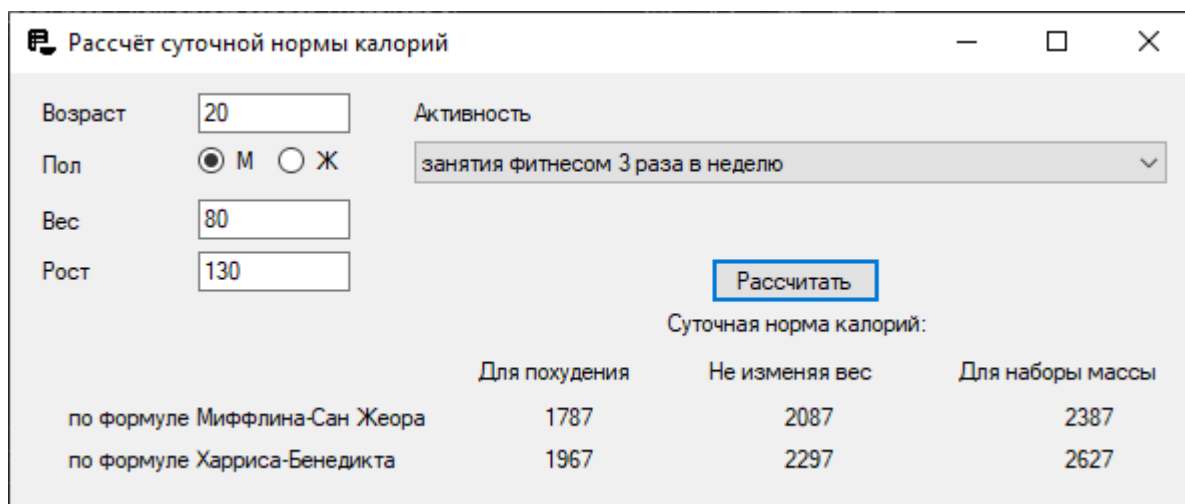
Специалист по тестированию: Штурмин Евгений Андреевич.

Дата и время тестирования: 21 апреля 2020, 20:40.

Входные данные:

- 1) Запускаем программу.
- 2) Заполняем поля данными: возраст – 20, пол – мужской, вес – 80, рост – 130, активность – занятия фитнесом 3 раза в неделю.
- 3) Нажимаем кнопку «Рассчитать».
- 4) Меняем данные: возраст – 20, пол – мужской, вес – 80, рост – 170, активность – занятия фитнесом 3 раза в неделю.
- 5) Нажимаем кнопку «Рассчитать».

Полученные выходные данные:



	Для похудения	Не изменяя вес	Для набора массы
по формуле Миффлина-Сан Жеора	1787	2087	2387
по формуле Харриса-Бенедикта	1967	2297	2627

Рисунок 4.5 – Результат работы программы при вводе первого варианта заполнения

	Для похудения	Не изменяя вес	Для набора массы
по формуле Миффлина-Сан Жеора	2070	2430	2790
по формуле Харриса-Бенедикта	2186	2561	2936

Рисунок 4.6 – Результат работы программы при вводе второго варианта заполнения

Тестовый пример 4.

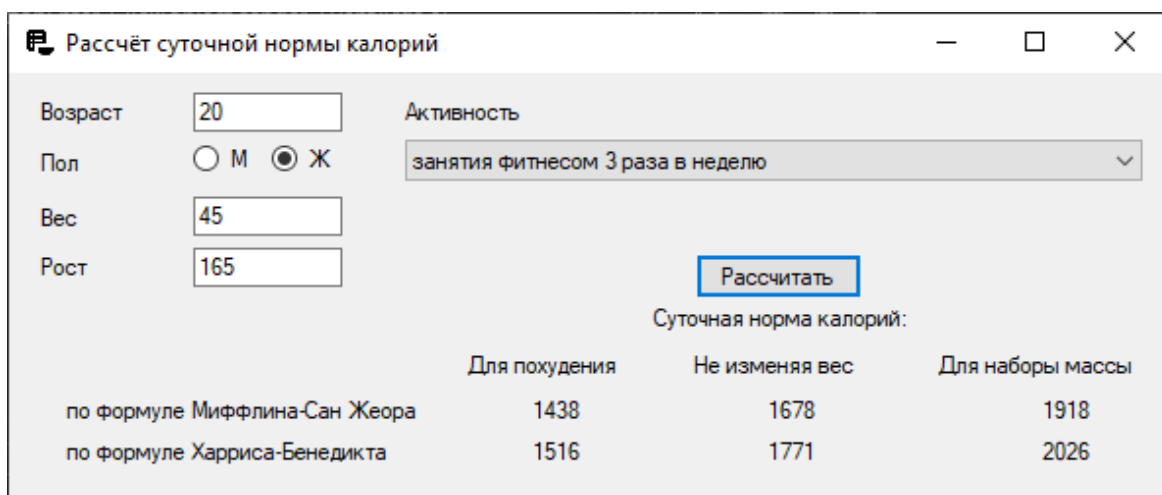
Специалист по тестированию: Штурмин Евгений Андреевич.

Дата и время тестирования: 21 апреля 2020, 20:45.

Входные данные:

- 1) Запускаем программу.
- 2) Заполняем поля данными: возраст – 20, пол – женский, вес – 45, рост – 165, активность – занятия фитнесом 3 раза в неделю.
- 3) Нажимаем кнопку «Рассчитать».
- 4) Меняем данные: возраст – 20, пол – женский, вес – 60, рост – 165, активность – занятия фитнесом 3 раза в неделю.
- 5) Нажимаем кнопку «Рассчитать».

Полученные выходные данные:



Рассчёт суточной нормы калорий

Возраст: 20 Активность: занятия фитнесом 3 раза в неделю

Пол: ☐ М ☒ Ж

Вес: 45

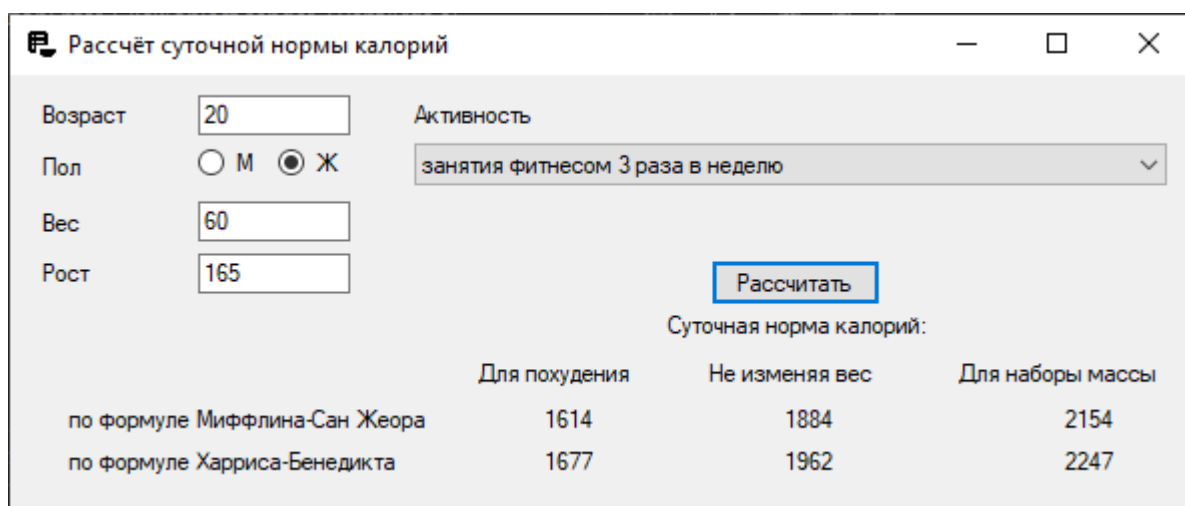
Рост: 165

Рассчитать

Суточная норма калорий:

	Для похудения	Не изменяя вес	Для набора массы
по формуле Миффлина-Сан Жеора	1438	1678	1918
по формуле Харриса-Бенедикта	1516	1771	2026

Рисунок 4.7 – Результат работы программы при вводе первого варианта заполнения



Рассчёт суточной нормы калорий

Возраст: 20 Активность: занятия фитнесом 3 раза в неделю

Пол: ☐ М ☒ Ж

Вес: 60

Рост: 165

Рассчитать

Суточная норма калорий:

	Для похудения	Не изменяя вес	Для набора массы
по формуле Миффлина-Сан Жеора	1614	1884	2154
по формуле Харриса-Бенедикта	1677	1962	2247

Рисунок 4.8 – Результат работы программы при вводе второго варианта заполнения

Тестовый пример 5.

Специалист по тестированию: Штурмин Евгений Андреевич.

Дата и время тестирования: 21 апреля 2020, 20:50.

Входные данные:

- 1) Запускаем программу.
- 2) Заполняем поля данными: возраст – 30, пол – мужской, вес – 80, рост – 170, активность – занятия фитнесом 3 раза в неделю.
- 3) Нажимаем кнопку «Рассчитать».

4) Меняем данные: возраст – 30, пол – мужской, вес – 80, рост – 170, активность – ежедневная физическая нагрузка плюс физическая работа.

5) Нажимаем кнопку «Рассчитать».

Полученные выходные данные:

Рассчёт суточной нормы калорий

Возраст: 30, Пол: ☒ М ☐ Ж, Вес: 80, Рост: 170, Активность: занятия фитнесом 3 раза в неделю

Рассчитать

Суточная норма калорий:

	Для похудения	Не изменяя вес	Для набора массы
по формуле Миффлина-Сан Жеора	2017	2362	2707
по формуле Харриса-Бенедикта	2123	2483	2843

Рисунок 4.9 – Результат работы программы при вводе первого варианта заполнения

Рассчёт суточной нормы калорий

Возраст: 30, Пол: ☒ М ☐ Ж, Вес: 80, Рост: 170, Активность: ежедневная физическая нагрузка плюс физическая работа

Рассчитать

Суточная норма калорий:

	Для похудения	Не изменяя вес	Для набора массы
по формуле Миффлина-Сан Жеора	2783	3263	3743
по формуле Харриса-Бенедикта	2921	3431	3941

Рисунок 4.10 – Результат работы программы при вводе второго варианта заполнения

Тестовый пример 6.

Специалист по тестированию: Штурмин Евгений Андреевич.

Дата и время тестирования: 21 апреля 2020, 20:55.

Входные данные:

- 1) Запускаем программу.
- 2) Заполняем поля данными: пол – мужской, вес – 80, рост– 170, активность – занятия фитнесом 3 раза в неделю.

- 3) Нажимаем кнопку «Рассчитать».

Полученные выходные данные:

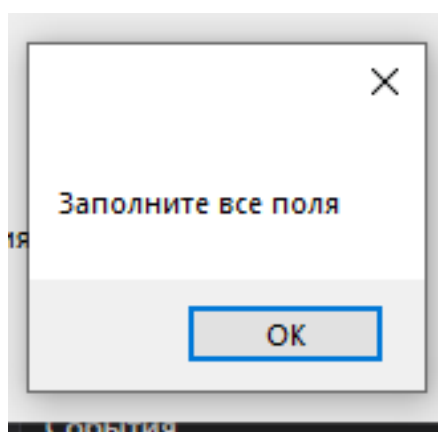


Рисунок 4.10 – Сообщение об ошибке

ЗАКЛЮЧЕНИЕ

В ходе прохождения практики по модулю ПМ.03 «Участие в интеграции программных модулей» в ВлГУ с 18-го апреля по 24-ое апреля 2020-го года, были получены теоретические и практические знания.

Также были успешно выполнены следующие задания:

- построены на языке UML диаграммы для приложения;
- разработка приложения «Калькулятор расчета суточной нормы калорий»;
- изучены принципы системы контроля версий;
- изучены принципы компиляторов вне интегрированной среды разработки;
- разработаны тест-кейсы и протестировано приложение;
- оформлен отчет по результатам практики.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

- 1) Формула Харриса-Бенедикта [Электронный ресурс] – Режим доступа: <https://www.calc.ru/Formula-Kharrisabenedikta.html> (Дата обращения: 22.04.2020)
- 2) Формула Миффлина-Сан Жеора [Электронный ресурс] – Режим доступа: <https://www.calc.ru/Formula-Mifflinasan-Zheora.html> (Дата обращения: 22.04.2020)
- 3) Определение уровня активности [Электронный ресурс] – Режим доступа: <https://beregifiguru.ru/Статьи/Как-определить-активность> (Дата обращения: 22.04.2020)
- 4) Общая характеристика языка UML [Электронный ресурс] – Режим доступа: <https://www.informicus.ru/Default.aspx?SECTION=6&id=73&subdivisionid=2> (Дата обращения: 22.04.2020)
- 5) Microsoft Visual Studio Community [Электронный ресурс] – Режим доступа: https://freesoft.ru/windows/microsoft_visual_studio_community_2017 (Дата обращения: 22.04.2020)
- 6) Команда dotnet build [Электронный ресурс] – Режим доступа: <https://docs.microsoft.com/ru-ru/dotnet/core/tools/dotnet-build> (Дата обращения: 22.04.2020)
- 7) Система контроля версий [Электронный ресурс] – Режим доступа: <https://git-scm.com/book/ru/v2/Введение-О-системе-контроля-версий> (Дата обращения: 22.04.2020)

ПРИЛОЖЕНИЕ А

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace TrainingPractice
{
    public partial class Form1 : Form
    {
        private int rezKkalHarries;
        private int rezKkalMifflin;
        private int age;
        private double height;
        private double weight;
        private bool gender;
        private double factorActive;
        public Form1()
        {
            InitializeComponent();
            comboBoxActive.SelectedIndex = 0;
        }
        private int FormulaMifflin(bool gender, double factorActive, int age, double weight,
double height)
        {
            if (gender == true)
            {
                return Convert.ToInt32(((10 * weight) + (6.25 * height) - (5 * age) +
5)*factorActive);
            }
            else
                return Convert.ToInt32(((10 * weight) + (6.25 * height) - (5 * age) - 161) *
factorActive);
        }
        private int FormulaHarries(bool gender, double factorActive, int age, double weight,
double height)
        {
            if (gender == true)
            {
                return Convert.ToInt32((88.362 + (13.397 * weight)+(4.799*height)-(5.677*age))
* factorActive);
            }
            else
                return Convert.ToInt32((447.593 + (9.247 * weight) + (3.098 * height) - (4.330
* age)) * factorActive);
        }
        private double Active (int id)
        {
            switch (id)
            {
                case 0:
                    return 1.2;
                case 1:
```

```

        return 1.375;
    case 2:
        return 1.46;
    case 3:
        return 1.55;
    case 4:
        return 1.64;
    case 5:
        return 1.72;
    case 6:
        return 1.9;
    default:
        return 0;
    }
}
private void buttonEnter_Click(object sender, EventArgs e)
{
    try
    {
        if (radioButtonM.Checked)
            gender = true;
        else
            gender = false;
        factorActive = Active(comboBoxActive.SelectedIndex);
        age = Convert.ToInt32(textBoxAge.Text);
        weight = Convert.ToDouble(textBoxWeight.Text);
        height = Convert.ToDouble(textBoxHeiht.Text);
        rezKkalMifflin = FormulaMifflin(gender, factorActive, age, weight, height);
        rezKkalHarries = FormulaHarries(gender, factorActive, age, weight, height);
        MifDef.Text = rezKkalMifflin.ToString();
        HarDef.Text = rezKkalHarries.ToString();
        MifDeficit.Text = (rezKkalMifflin - (rezKkalMifflin) / 100 * 15).ToString();
        HarDeficit.Text = (rezKkalHarries - (rezKkalHarries) / 100 * 15).ToString();
        MifProficit.Text = (rezKkalMifflin + (rezKkalMifflin) / 100 * 15).ToString();
        HarProficit.Text = (rezKkalHarries + (rezKkalHarries) / 100 * 15).ToString();
    }
    catch
    {
        MessageBox.Show("Заполните все поля");
    }
}

private void textBoxAge_KeyPress(object sender, KeyPressEventArgs e)
{
    if ((e.KeyChar <= 47 || e.KeyChar >= 58 ) && e.KeyChar != 8)
        e.Handled = true;
}

private void textBoxWeight_KeyPress(object sender, KeyPressEventArgs e)
{
    if ((e.KeyChar <= 45 || e.KeyChar >= 58 ) && e.KeyChar != 8)
        e.Handled = true;
}

private void textBoxHeiht_KeyPress(object sender, KeyPressEventArgs e)
{
    if ((e.KeyChar <= 45 || e.KeyChar >= 58) && e.KeyChar != 8)
        e.Handled = true;
}
}
}

```