

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение

высшего образования

**«Кубанский государственный университет»**

**(ФГБОУ ВО «КубГУ»)**

**Институт среднего профессионального образования**

**(ИНСПО)**

**РАСЧЕТНО-ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

**к курсовой работе**

**«МДК 03.01 Технология разработки программного обеспечения» н а тему:**

**Разработка программы конвертора размеров женской одежды в разных странах**

Выполнил студент гр. ПКС-4

Талалуев Сергей Андреевич

Руководитель преподаватель

Трубников Юрий.Юрьевич.

Краснодар, 2022

**Содержание**

[**ВВЕДЕНИЕ** 3](#_Toc122150844)

[**ГЛАВА 1: Анализ задания и выбор технологии, языка и среды разработки** 4](#_Toc122150845)

[1.1 Анализ темы. 4](#_Toc122150846)

[1.2 Выбор таблиц для расчёта размеров женской одежды 5](#_Toc122150847)

[1.3 Выбор технологии, среды разработки и языка программирования 6](#_Toc122150848)

[**ГЛАВА 2: Разработка диаграммы вариантов использования** 7](#_Toc122150849)

[2.1 Определение способа взаимодействия пользователей с системой 7](#_Toc122150850)

[2.2 Создание диаграммы вариантов использования 7](#_Toc122150851)

[**ГЛАВА 3: Определение структуры программного продукта** 9](#_Toc122150852)

[3.1 Создание структурной и функциональной схем ПП 9](#_Toc122150853)

[**ГЛАВА 4: Реализация программного продукта** 11](#_Toc122150854)

[4.1 Разработка интерфейса 11](#_Toc122150855)

[**4.1.1 Создание наименования программы.** 11](#_Toc122150856)

[**4.1.2 Создание основного интерфейса.** 12](#_Toc122150857)

[**ГЛАВА 5: Тестирование программы.** 18](#_Toc122150858)

[5.1 Выбор стратегии тестирования программы. 18](#_Toc122150859)

[5.2 Тестирование программы. 18](#_Toc122150860)

[ВЫВОД 19](#_Toc122150861)

[ЛИТЕРАТУРА 20](#_Toc122150862)

[ПРИЛОЖЕНИЕ А 21](#_Toc122150863)

# **ВВЕДЕНИЕ**

Многие из нас сегодня заказывают наряды по интернету — действительно, это удобно и быстро, к тому же значительно дешевле, чем покупать их в магазинах. При индивидуальном пошиве мастер делает раскрой согласно снятым меркам. А вот при выборе готовой одежды очень важно знать, что именно будет соответствовать вашей фигуре. К сожалению, системы определения размеров в разных странах существенно отличаются друг от друга. Хотя в настоящее время и намечается тенденция к переходу на единый международный буквенный стандарт, но многие производители продолжают маркировать одежду по схемам, принятым в их стране.

Знание точных параметров своей фигуры может сэкономить покупательнице массу времени и при покупке в обычном магазине, т.к. только очень опытные продавцы-консультанты способны «на глаз» оценить предполагаемый размер и предложить соответствующие образцы одежды для примерки. Последовательный же перебор всех близких к нужному размеру вещей без предварительного их отбора по характеристикам, указанным на навесных этикетках и вшивных бирках, может растянуться на многие часы.

Конвертор размеров женской одежды поможет вам разобраться в достаточно большом количестве различных систем обозначения размеров одежды существующих в мире.

Аналоги программ подбора размеров – Allcalc, Good Sizer 3D, Sizer и Savanasoft.

Главная их особенность – все они мобильные приложения. Их легко найти, скачать и сразу использовать. У них современный и удобный интерфейс и большой функционал. Их недостаток – отсутствие версии на ПК.

**Цель работы** – разработать программу-конвертор размеров женской одежды для компьютеров, работающих на IBM.

**Задачи исследования**:

1) Изучить предметную область таблиц размеров и их расчёт;

2) Выбрать технологию, язык программирования и среду разработки программы;

3) Разработать программу, описывая ход работы;

4) Провести тестирование готового продукта.

# **ГЛАВА 1: Анализ задания и выбор технологии, языка и среды разработки**

## 1.1 Анализ темы.

Одежду стали шить по фигуре сравнительно недавно. Вначале одежда была более свободная и ее подвязывали поясом, чтобы она плотнее прилегала к фигуре. Так были устроены тоги в Древнем Риме. Во многих странах такая одежда была популярна вплоть до девятнадцатого века, в то время как женская одежда стала более прилегающей, особенно в области лифа и рукавов. Так как мужская одежда была широкой, размеры не имели такого большого значения, как в женской одежде.

Выбор размеров штанов, юбок и рубашек зависит от страны, и от средних показателей размеров жителей этих стран. Например, в странах, где большинство женщин носят маленькие размеры, приезжим из СНГ и других стран Европы и Северной Америки трудно найти свой размер не только из-за слишком коротких или узких в талии штаны. Даже если длина и обхват талии подходят, штаны могут быть слишком узкими в объеме бедра, то есть по ширине штанины. Например, приезжие в Японию жалуются, что штаны плохо сидят или слишком короткие.

Поэтому для облегчения данной задачи, рекомендуется пользоваться специальными разными таблицами, в которых одежда всегда соответствует своим размерам.

Например, российские размеры начинаются с 38 и заканчиваются 64, а европейские — с 32 по 58. Чтобы упростить для покупателей выбор изделий из разных стран, была принята международная буквенная размерная сетка. Ниже смотрите соответствие международного стандарта российскому:

|  |  |  |
| --- | --- | --- |
| № | Стандарт | Размер |
| 1 | XS (eXtra Small) | 40 |
| 2 | S (Small) | 42 |
| 3 | M (Medium) | 44 |

Продолжение таблицы

|  |  |  |
| --- | --- | --- |
| 4 | L (Large) | 46 |
| 5 | XL (eXtra Large) | 48 |
| 6 | XXL (eXtra eXtra Large) | 50 |

Таблица 1 Российский международный стандарт

Порой чтобы вещь идеально села на нас, обобщенных параметров не достаточно. Особенностью программы конвертора является то, что вписывая физические параметры он "составляет" наиболее подходящий пользователю стандарт.

Для того чтобы определить размер одежды, нужно вооружиться сантиметром или сантиметровой лентой, чтобы измерить длины тела.

* Для роста — встаньте спиной к стене так, чтобы пятки, ягодицы, плечи и голова к ней прикасались. Затем измерьте расстояние от нее до пола.
* Обхват талии — в самом узком месте.
* Обхват бедер — в самой выступающей точке ягодиц.
* Длину рукава (для рубашек) измеряют от середины спины вдоль плеча и до основания большого пальца (запястья). При этом рука должна быть в полусогнутом положении.

Полученное значение нужно сверить с таблицей соответствия размеров.

## 1.2 Выбор таблиц для расчёта размеров женской одежды

«В Европе антропологические исследования проводятся каждые 15 лет. Считается, что именно за такой период времени у людей заметно изменяются рост и вес. В соответствии с этими изменениями швейных дел мастера периодически разрабатывают новые лекала, а то и вообще придумывают совершенно новые методики конструирования одежды.  
Что касается размерной типологии женщин, то, действительно, в средне-возрастных группах отмечается увеличение объёма бедер, а также изменение соотношения балансовых измерение Длина переда до талии Длина спины до талии в сторону увеличения Длины переда.  
Отмечается увеличение роста в группах от 18 до 40 лет.  
Значительно увеличилась численность в полнотных группах с так называемой отрицательной полнотой, года объём бёдер меньше объёма груди.». Московская правда | 2004. Как видим приблизительно каждое поколение большинство таблиц пересматриваются и устаревают.

На сегодняшний день обозначают стандарт женской одежды ISO 8559, которая очень напоминает европейский стандарт [EN 13402](https://en.wikipedia.org/wiki/EN_13402). Я выбрал его, т.к. она наиболее точна из тех, что я исследовал и была введена 03.01.2021. Согласно этому стандарту далее приведены основные таблицы, которые я добавил в программу:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Соединенные Штаты** | 0 | 2 | 4 | 6 | 8 | 10 | 12 | 14 | 16 | 18 | 20 | 22 | 24 |
| **Великобритания** | 4 | 6 | 8 | 10 | 12 | 14 | 16 | 18 | 20 | 22 | 24 | 26 | 28 |
| **Германия** | 30 | 32 | 34 | 36 | 38 | 40 | 42 | 44 | 46 | 48 | 50 | 52 | 54 |
| **Франция** | 32 | 34 | 36 | 38 | 40 | 42 | 44 | 46 | 48 | 50 | 52 | 54 | 56 |
| **Италия** | 36 | 38 | 40 | 42 | 44 | 46 | 48 | 50 | 52 | 54 | 56 | 58 | 60 |
| **Корея** | 44 | 44 | 55 | 55 | 66 | 66 | 77 | 77 | 88 | 88 |  |  |  |

Таблица 2 Женские платья и костюмы

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **США** | 4 | 6 | 8 | 10 | 12 |
| **Англия** | 8 | 10 | 12 | 14 | 16 |
| **Россия** | 46 | 48 | 50 | 52 | 54 |
| **Франция** | 36 | 38 | 40 | 42 | 44 |
| **Япония** | 7 | 11 | 15 | 17 | 21 |
|  |  |  |  |  |  |
| **Бюст** | 81 см | 86 см | 91 см | 97 см | 102 см |
| **Талия** | 61 см | 67 см | 74 см | 79 см | 84 см |
| **Бёдра** | 89 см | 94 см | 99 см | 104 см | 109 см |

Таблица 3 Размеры женской одежды.

## 1.3 Выбор технологии, среды разработки и языка программирования

Для своей работы я выбрал IDE Visual Studio Code, так как для моего языка программирования там есть весь функционал от инструментов отладки до встроенных нагрузочных тестов.

При разработке программного обеспечения было отдано предпочтение структурному подходу к программированию. Цель создаваемой программы достигается составлением композиции из отдельных модулей, разработанных применительно к поставленной задаче

Преимущества этой технологии заключается в следующем:

* отдельные блоки могут разрабатываться и проходить тестирование независимо от остальной части программы;
* модули, написанные для одной программы, могут быть использованы в другой;
* существенно упрощается построение программы и её отладка.

Языком программирования стал Python. Python является одним из наиболее распространенных современных языков программирования. Язык Python хорошо зарекомендовал себя эффективностью, лаконичностью записи алгоритмов, логической стойкостью программ. Также на нём очень легко сделать компиляцию проекта. Python регулярно получает обновления и имеет широкую базу фреймворков.

В ходе вёрстки я познакомился с одним из них. Tkinker представляет собой стандартный интерфейс Python для набора инструментов Tcl/Tk GUI. Tkinker доступен на большинстве платформ Unix, включая macOS, а также в системах Windows. О нем я расскажу позже..

# **ГЛАВА 2: Разработка диаграммы вариантов использования**

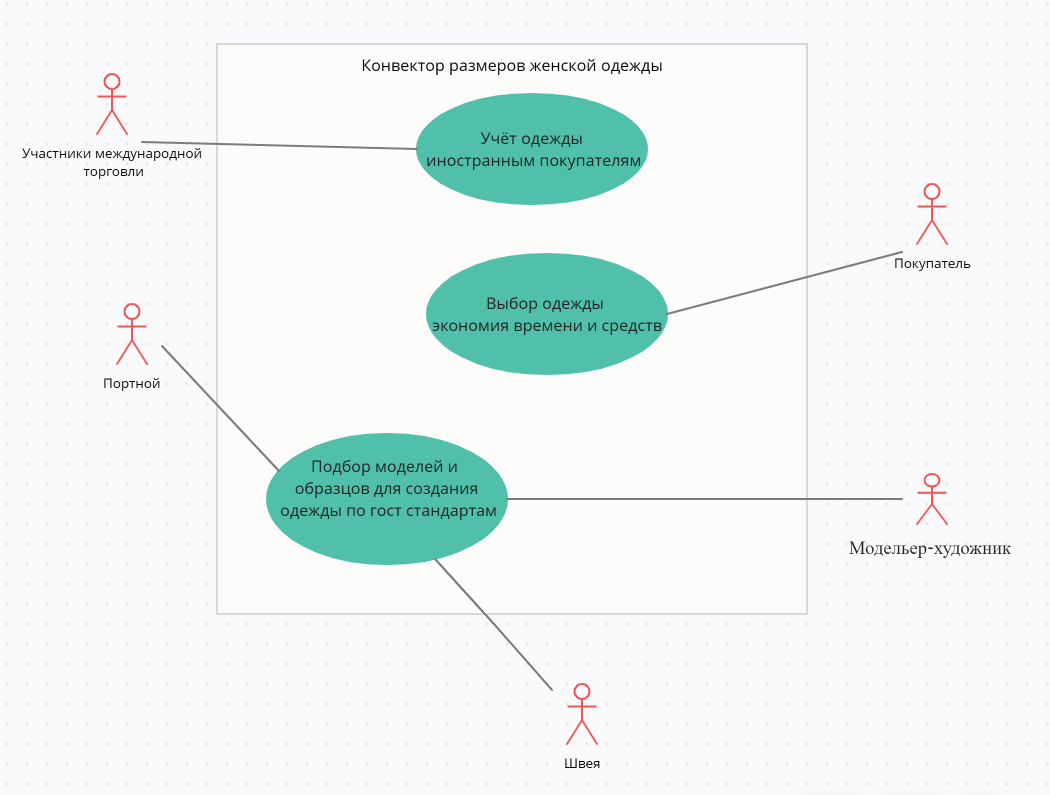
## 2.1 Определение способа взаимодействия пользователей с системой

Самым распространённым способом взаимодействия пользователей с программой является её интерфейс. Он должен быть интуитивным и понятным для пользователя. В нём должны быть все элементы, нужные для ввода и расчёта размеров. (*ПРИЛОЖЕНИЕ А*)

## 2.2 Создание диаграммы вариантов использования

Диаграмма вариантов использования (UseCase Diagram) – диаграмма, показывающая, в каких целях используется тот или иной продукт. В моём случае, нужно создать диаграмму того, в каких целях может использоваться программа размеров женской одежды.

Использованные диаграммы были созданы на сайте Creately.com.

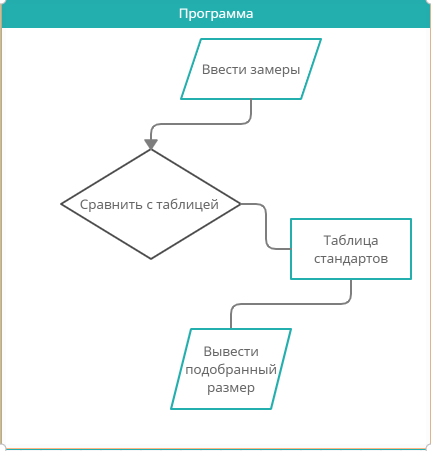


*Рис. 1: UseCase-диаграмма для программы размеров женской одежды.*

# **ГЛАВА 3: Определение структуры программного продукта**

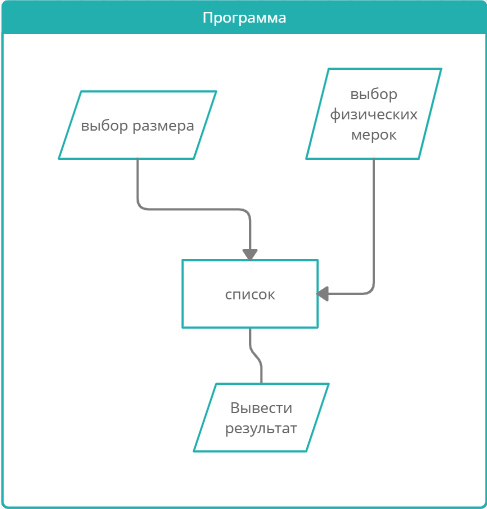
## 3.1 Создание структурной и функциональной схем ПП

Структурная схема – схема, отражающая состав и взаимодействие частей разрабатываемой программы.



*Рис. 2: Структурная схема программы размеров женской одежды.*

Функциональная схема – схема взаимодействия частей системы с описанием информационных потоков, состава данных в потоках и указанием используемых файлов и устройств.



*Рис. 3: Функциональная схема программы размеров женской одежды.*

# **ГЛАВА 4: Реализация программного продукта**

## 4.1 Разработка интерфейса

В качестве софта для этой задачи я, как уже упоминалось в пункте 1.2, пользуюсь фраймворком Tkinker.

Для удобства в вёрстке были созданы отдельные блоки данных, которые будут описаны в последующих подразделах.

### **4.1.1 Создание наименования программы.**

Для начала, следует импортировать Tkinter и создать окно, в котором мы зададим его название:

from tkinter import \*

from tkinter.ttk import Combobox

class Window:

    def \_\_init\_\_(self, width, height, title='MyWindow', resizable=(False, False)):

        self.root = Tk()

        self.root.title(title)

if \_\_name\_\_ == "\_\_main\_\_":

    window = Window(800, 720, "КАЛЬКУЛЯТОР ЖЕНСКОЙ ОДЕЖДЫ")

    window.run()

В переменную window записывается значение title – название нашей программы.



*Рис. 4: Готовое название программы;*

Функция run() запускает обработчик окон и цикл программы mainloop.

def run(self):

        self.draw\_widgets()

        self.root.mainloop()

В случае, если не вызвать функцию mainloop, для пользователя ничего не отобразится.

### **4.1.2 Создание основного интерфейса.**

Далее я буду прописывать параметры основного окна программы.

Устанавливаю размер окна по умолчанию, используя функцию geometry следующим образом:

self.root.geometry('450x700')

В скобках указаны координаты отступа от края по X и Y.

Также добавляю прозрачность окна бэкграунд через ['bg'] и wm\_attributes:

self.root['bg'] = '#fafafa'

self.root.wm\_attributes('-alpha', 0.9)

Здесь я отключаю возможность расширения окна пользователем по ширине и высоте через параметр resizable.

self.root.resizable(resizable[0], resizable[1])

Ключевым строительным блоком в графическом приложении являются различные элементов управления, с которыми взаимодействует пользователь, как кнопки, метки, поля ввода. В Tkinter имеется богатая палитра различных элементов управления, которые называются виджетами.

Для разметки виджетов требуются блоки (frame). Frame - фрейм, который организует виджеты в группы.

Tkinter предоставляет виджеты в двух вариантах: виджеты, которые располагаются непосредственно в пакете tkinter, и виджеты из пакета tkinter.ttk. С одной стороны, оба пакета предоставляют практически одни и те же виджеты, например, виджет Button есть в обоих пакетах. Но с другой стороны, ttk предоставляет чуть больше функциональности по настройке виджетов, в частности, по их стилизации. И считается, что виджеты из ttk несколько современнее, чем стандартные, в то же время с ttk, возможно, чуть сложнее работать.

Я создал пять фреймов. Два из них будут изображать стандарты размеров вверху. В третьем разположен текст, а два остальных будут расчитывать физические замеры внизу окна:

frame1 = Frame(self.root)

        frame1.grid(column=0, row=0, ipadx=6, ipady=6, padx=14, pady=14)

        frame2 = Frame(self.root)

        frame2.grid(column=1, row=0, ipadx=6, ipady=6, padx=14, pady=14)

        frame3 = Frame(self.root)

        frame3.grid(columnspan=2, row=1, ipadx=6, ipady=6, padx=14, pady=14)

        frame4 = Frame(self.root)

        frame4.grid(column=0, row=2, ipadx=6, ipady=6, padx=14, pady=14)

        frame5 = Frame(self.root)

        frame5.grid(column=1, row=2, ipadx=6, ipady=6, padx=14, pady=14)

Позиционирование grid() позволяет поместить виджет в определенную ячейку условной сетки или грида.

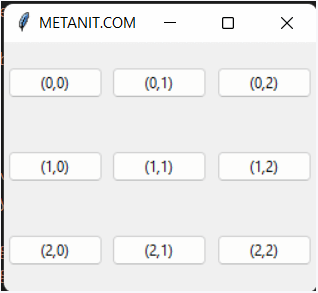


Таблица 4 представление фреймов с помощью grid.

column=0, row=0, ipadx=6, ipady=6, padx=14, pady=14

Здесь отступ между блоками равен 6, а внешний отступ 14 единицам.

frame3.grid(columnspan=2, row=1, ipadx=6, ipady=6, padx=14, pady=14)

С помощью параметра columnspan мы можем объединить ячейки.

В фреймы помещаем надписи стран, тоже с позиционированием grid, чтобы не смешались блоки.

title1 = Label(frame1, padx=20, pady=10, text='Россия и Украина', font=80).grid()

  title2 = Label(frame1, padx=20, pady=10, text='США', font=80).grid()

  title3 = Label(frame1, padx=20, pady=10, text='Германия', font=80).grid()

  title4 = Label(frame1, padx=20, pady=10, text='Англия', font=80).grid()

  title5 = Label(frame1, padx=20, pady=10, text='Франция', font=80).grid()

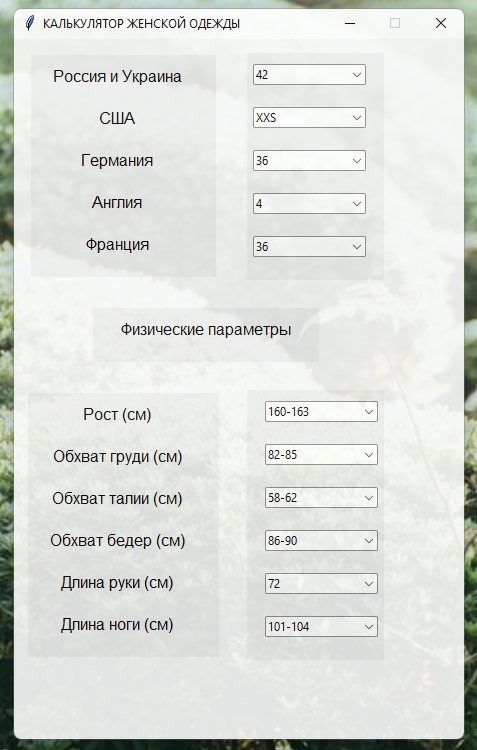


Таблица 5 формы и надписи программы.

Для ввода пользователем данных я выбрал выпадающий список. Для виджета поля с выпадающем списком, использую класс Combobox из ttk.

from tkinter.ttk import Combobox

В пять переменных записывается метод Combobox:

self.sngsizes = Combobox(frame2, values=(42,44,46,48,50,52,54,56,58), width=15, height=25)

self.ussizes = Combobox(frame2, values=('XXS','XS','S','M','L','XL','XXL','2XL','3XL'),

width=15, height=25)

self.grsizes = Combobox(frame2, values=(36,38,40,42,44,46,48,50,52), width=15, height=25)

self.uksizes = Combobox(frame2, values=(4,6,8,10,12,14,16,18,20), width=15, height=25)

self.frsizes = Combobox(frame2, values=(36,38,40,42,44,46,48,50,52), width=15, height=25)

Каждый список содержит проиндексированный список значений одежды.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **№ страны** | **Размер женской одежды** | | | | | | | | |
| **1** | 42 | 44 | 46 | 48 | 50 | 52 | 54 | 56 | 58 |
| **2** | XXL | XS | S | M | L | XL | XXL | 2XL | 3XL |
| **3** | 36 | 38 | 40 | 42 | 44 | 46 | 48 | 50 | 52 |
| **4** | 4 | 6 | 8 | 10 | 12 | 14 | 16 | 18 | 20 |
| **5** | 36 | 38 | 40 | 42 | 44 | 46 | 48 | 50 | 52 |

Размер женской одежды

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **№** | **Физические параметры** | | | | | | | | |
| **1** | 160-163 | 162-165 | 164-167 | 166-169 | 168-171 | 170-173 | 172-175 | 174-177 | 176-179 |
| **2** | 82-85 | 86-89 | 90-93 | 94-97 | 98-101 | 102-105 | 106-109 | 110-113 | 114-117 |
| **3** | 58-62 | 61-65 | 64-68 | 67-71 | 70-74 | 73-77 | 76-80 | 79-83 | 82-86 |
| **4** | 86-90 | 89-93 | 92-96 | 95-99 | 98-102 | 101-105 | 104-108 | 107-111 | 110-114 |
| **5** | 72 | 75 | 77 | 80 | 82 | 85 | 87 | 90 | 92 |
| **6** | 101-104 | 102-105 | 103-106 | 104-107 | 105-108 | 106-109 | 107-110 | 108-111 | 109-112 |

Физические параметры

Чтобы их вывести использую .pack(), а метод current(0) указывает элемент списка по умолчанию. В данном примере индекс 0 – потому что отсчет в python начинается с нуля:

self.sngsizes.current(0)

        self.sngsizes.pack(anchor=NW, padx=6, pady=11)

        self.ussizes.current(0)

        self.ussizes.pack(anchor=NW, padx=6, pady=11)

Внешние отступы padx и pady.означают по-вертикали 11, по-горизонтали 6 единиц.

Далее будет показан основная функция программы. В **tkinter** с помощью метода **bind** между собой связываются виджет, событие и действие. Например, виджет – кнопка, событие – клик по ней левой кнопкой мыши, действие – отправка сообщения. Другой пример: виджет – текстовое поле, событие – нажатие Enter, действие – получение текста из поля методом **get** для последующей обработки программой. Действие оформляют как функцию или метод, которые вызываются при наступлении события.

self.sngsizes.bind("<<ComboboxSelected>>", sngsizes)

        self.ussizes.bind("<<ComboboxSelected>>", ussizes)

        self.grsizes.bind("<<ComboboxSelected>>", grsizes)

        self.uksizes.bind("<<ComboboxSelected>>", uksizes)

        self.frsizes.bind("<<ComboboxSelected>>", frsizes)

Событие в данном случае – изменение значения в выпадающем списке пользователем.

Итак, когда пользователь меняет значение происходит событие я передаю индекс поля которое выбрано.

def sngsizes(event):

            sngindex = self.sngsizes.current()

            self.sngsizes.current(sngindex)

            self.ussizes.current(sngindex)

            self.grsizes.current(sngindex)

            self.uksizes.current(sngindex)

            self.frsizes.current(sngindex)

            self.asizes.current(sngindex)

            self.bsizes.current(sngindex)

            self.csizes.current(sngindex)

            self.dsizes.current(sngindex)

            self.esizes.current(sngindex)

            self.fsizes.current(sngindex)

Этот индекс содержится и в других полях и имеет значения разных размеров одежды.

Все списки меняют свое значение согласно индексу (0….0, 4…4 и т.д.).

def sngsizes(event):

            sngindex = self.sngsizes.current()

            self.sngsizes.current(sngindex)

            self.ussizes.current(sngindex)

            self.grsizes.current(sngindex)

            self.uksizes.current(sngindex)

            self.frsizes.current(sngindex)

            self.asizes.current(sngindex)

            self.bsizes.current(sngindex)

            self.csizes.current(sngindex)

            self.dsizes.current(sngindex)

            self.esizes.current(sngindex)

            self.fsizes.current(sngindex)

# **ГЛАВА 5: Тестирование программы.**

## 5.1 Выбор стратегии тестирования программы.

Для тестирования программы была выбрана стратегия Smoke Testing.

Это короткий цикл тестов, подтверждающий (отрицающий) факт того, что приложение стартует и выполняет свои основные функции. Данный тип тестирования позволяет на начальном этапе выявить основные быстро находимые критические дефекты. Я выбрал его потому что моя программа небольшая и можно сравнить её с начальным этапом разработки. Также в этом случае дымовое тестирование позволяет избежать затрат ограниченных ресурсов на неоправданное проведение глубоких проверок.

## 5.2 Тестирование программы.

Задача котроую нужно достичь это требования надежности и работоспособности проограммы как в целом так и отдельных её частей.

Тестирование будет на двух персональных ЭВМ: Acer 5270G и Asus n53f.

Данный тип тестирования будет включать в себя проверку следующих функциональностей:

* Menu (клик кнопок окна);
* Выпадающий список (работоспособность всех списков);
* Результаты (Вывод верных значений в специальных полях);

Можно разделить результаты на пройдено и провалено:

1. **Пройдено: все отлично работает.**
2. **Провалено: не работает.**

|  |  |  |
| --- | --- | --- |
| **Menu** | **Выпадающий список** | **Результаты** |
| **Пройдено** | **Пройдено** | **Пройдено** |
| **Пройдено** | **Пройдено** | **Пройдено** |
| **Пройдено** | **Пройдено** | **Пройдено** |
| **Пройдено** | **Пройдено** | **Пройдено** |

Таблица 6 тестирование Smoke Testing.

# *ВЫВОД*

В заключение следует отметить, что программное обеспечение, разработанное для конвертации размеров женской одежды, оказалось ценным инструментом для понимания потенциальной необходимости перевода размеров. Вводя информациюо **объёме бюста, талии и бедер** или выбирая наиболее подходящие размеры своей страны программное обеспечение способно рассчитать тип размера, который используется иностранными производителями.

В ходе этой работы удалось успешно разработать и внедрить удобный интерфейс, который позволяет легко вводить мерки и рассчитывать размер. Также были внедрены различные алгоритмы и формулы для точного расчета определения стандартов одежды на основе введённых данных. Кроме того, было произведено тестирование программы, используя стратегию Smoke testing.  
Кроме того согласно истории женская одежда будет претерпевать свои изменения и стандарты. Подобные вычисления сильно облегчат жизнь девушкам и женщинам, которые любят ходить по магазинам и очень часто обновляют свой гардероб или мастерам, готовые создавать этот продукт.

# *ЛИТЕРАТУРА*

Основная литература

1. https://docs.cntd.ru/document/1200175372 - национальный стандарт российской одежды.
2. https://ecalc.ru/sizes/ - статья о размерах женской одежды.
3. https://app.creately.com/d/7slUEHHcpfU/edit/s/tPmWtebJprx - UML диаграммы.

Дополнительная литература

1. https://pythonru.com/uroki/obuchenie-python-gui-uroki-po-tkinter - Разработка программы на python и tkinter.

2. https://metanit.com/python/tkinter/1.1.php - Обучающая документация по использованию фреймворка Tkinter.

# *ПРИЛОЖЕНИЕ А*

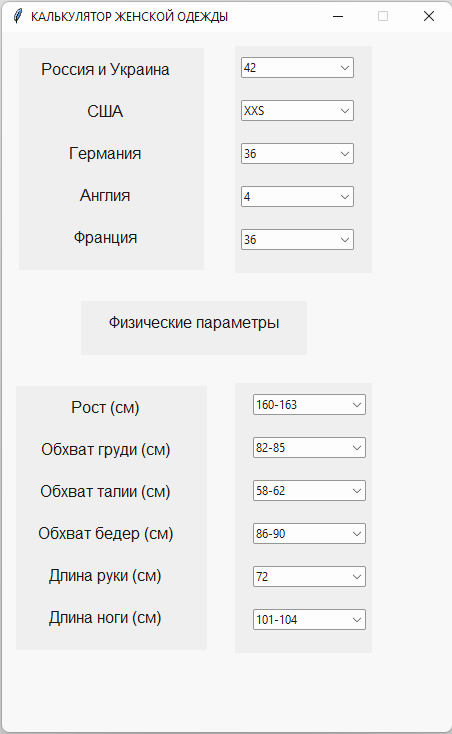


Рис. 10: Полный интерфейс программы конвертора размеров женской одежды в разных странах.