



Государственное бюджетное образовательное учреждение высшего образования
Московской области

ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ

Колледж космического машиностроения и технологий

КУРСОВОЙ ПРОЕКТ

По МДК.01.02 «Прикладное программирование»

Тема: «Разработка приложения «Туристский клуб» на Python»

Выполнил студент

Суркова Валерия Андреевна

Группа П1-17

_____ (Подпись)

_____ (Дата сдачи работы)

Проверил преподаватель

Гусятинер Леонид Борисович

_____ (Подпись)

_____ (Оценка)

Королёв 2020 г.

Оглавление

Введение	3
1. Теоретическая часть.....	4
1.1. Описание предметной области.....	4
1.2. Описание существующих разработок.....	6
1.2.1. Туристский клуб "Aqua Natura"	7
1.2.2. Туристский клуб «Перевал».....	8
2. Проектная часть.	10
2.1. Диаграмма прецедентов.....	10
2.2. Выбор инструментов	12
2.3. Проектирование сценария.	13
2.4. Диаграмма классов.....	14
2.5. Описание главного модуля.....	15
2.6. Описание спецификаций к модулям.	17
2.7. Описание модулей.	18
2.8. Описание тестовых наборов модулей.....	21
2.9. Описание применения средств отладки.	22
3. Эксплуатационная часть.	23
3.1. Руководство оператора.	23
3.1.1. Назначение программы.	23
3.1.2. Условия выполнения программы.....	23
3.1.3. Выполнение программы.....	23
Заключение.....	27
Список литературы и интернет – источников.	28
Приложение 1. Главный модуль.	29
Приложение 2. Окно «Маршруты»	30
Приложение 3. Окно «Инструкторы»	33
Приложения 4. Окно «Типы»	36
Приложение 5. Окно «О нас»	39

Введение

Целью данного курсового проекта является написание программы «Туристский клуб». Эта тема является актуальной, так как большинство людей любят путешествовать. Данный курсовой проект поможет облегчить работу персоналу и туристам туристского клуба. Также в этой программе будет простой для понимания интерфейс.

В первой части записки будет рассмотрена предметная область данной темы, а также будут примеры уже существующих разработок по данной теме.

Во второй части будут рассмотрены инструменты и модули, которые были разработаны, структура программной части и листинги ключевых частей программных модулей.

В третьей части будет рассмотрено руководство оператора, а также изображение графического интерфейса программы.

1. Теоретическая часть.

1.1. Описание предметной области.

Туристский клуб - туристское самостоятельное объединение, широко представленная форма объединения туристов. Клуб туристов является основной организационной единицей активного и самостоятельного туризма.

Туризм является самым массовым общественным движением в мире.

Существует большое количество видов туризма:

1. Спортивный туризм – это вид спорта, в основе которого лежат соревнования на маршрутах, включающих преодоление категорированных препятствий в природной среде (перевалов, вершин (в горном туризме), порогов (в водном туризме), каньонов, пещер и пр.), и на дистанциях, проложенных в природной среде и на искусственном рельефе.
2. Событийный туризм – это вид туризма, поездки при котором приурочены к каким-либо событиям. События могут относиться к сфере культуры, спорта, бизнеса и т. д. Примеры событий, вызывающих всплеск событийного туризма: Олимпиады, чемпионаты мира по футболу, рок-фестивали, кинофестивали, карнавалы, экономические форумы, авиасалоны.
3. Лечебный туризм – самое массовое и популярное направление лечебного туризма
4. Культурно – исторический туризм – это форма туризма, цель которого состоит в знакомстве с культурой и культурной средой места посещения, включая ландшафт, знакомство с традициями жителей и их образом жизни, художественной культурой и искусством, различными формами проведения досуга местных жителей.
5. Экотуризм – это природный туризм, который включает изучение окружающей природной среды и служит для улучшения обстановки в этой среде.

6. Деловой туризм – область индустрии туризма (наряду с рекреационным туризмом, англ. leisure travel), обеспечивающая организацию и управление командировками (деловыми поездками, англ. business trips) сотрудников различных компаний.

1.2. Описание существующих разработок.

Как правило интерфейс таких программ чаще всего простой и понятный в использовании для людей.

Цели использования туристского клуба:

1. Запись и просмотр различных туров
2. Введения туризма
3. Расчет различных статистических показателей: какие туры чаще всего заказывают, в какое время года больше всего туристов.

Основные требования:

1. Введение отчетности в соответствии с законодательством страны, в которой расположена организация
2. Запись на определенное направление
3. Оплата при оформлении тура.

1.2.1. Туристский клуб "Aqua Natura"

Программа туристский клуб "Aqua Natura" разработана для вступления в туристский клуб и записи на различные туры.

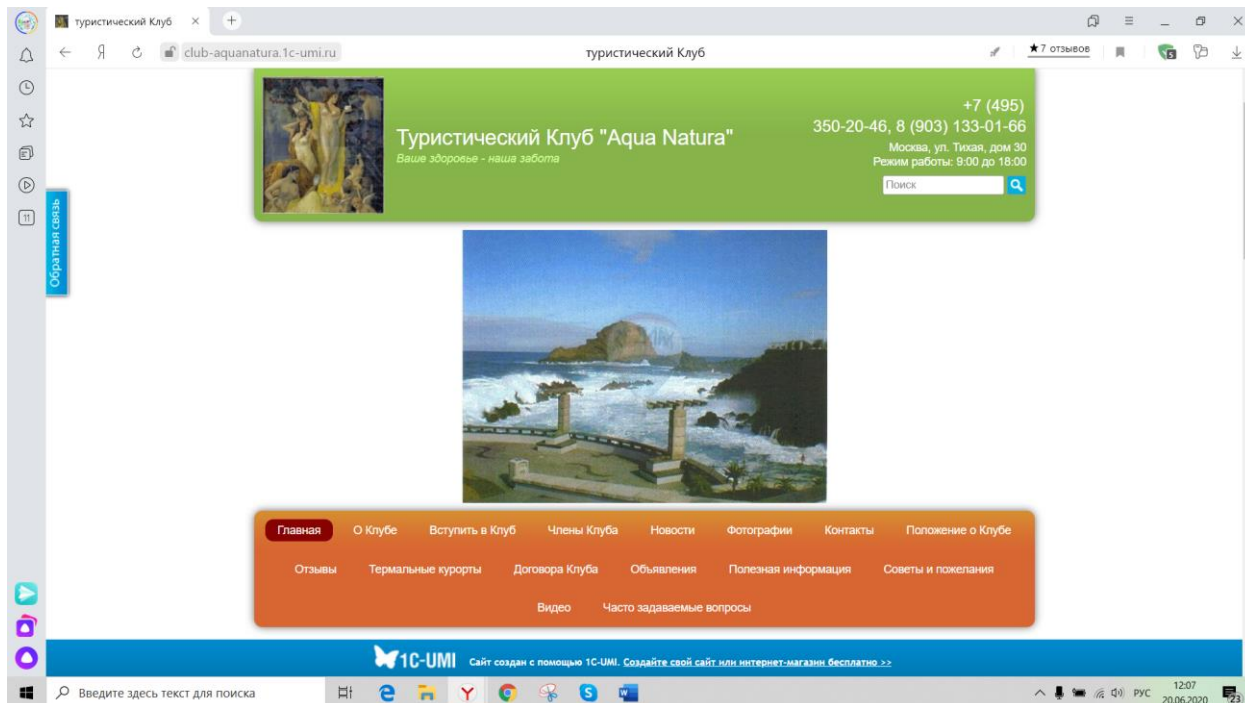


Рисунок 1. Интерфейс туристского клуба

Туристский клуб состоит из 3-х направлений: экологического, оздоровительного и лечебного.

Статус членства:

- а) коммерческий;
- б) общественно-социальный (льготный).

Членом Клуба может стать любой гражданин Российской Федерации.

Коммерческое направление предусматривает персональный подбор путевок на отдых и лечение для члена клуба на известных курортах Европы, Америки и Азии по рыночной стоимости.

Общественно-социальное направление предоставляет членам Клуба льготные путевки, согласно контрактам, на отдых и лечение на известных термальных курортах и клиниках с полным медицинским сопровождением.

1.2.2. Туристский клуб «Перевал»

Программа туристский клуб "Перевал" разработана для записи на различные туры.

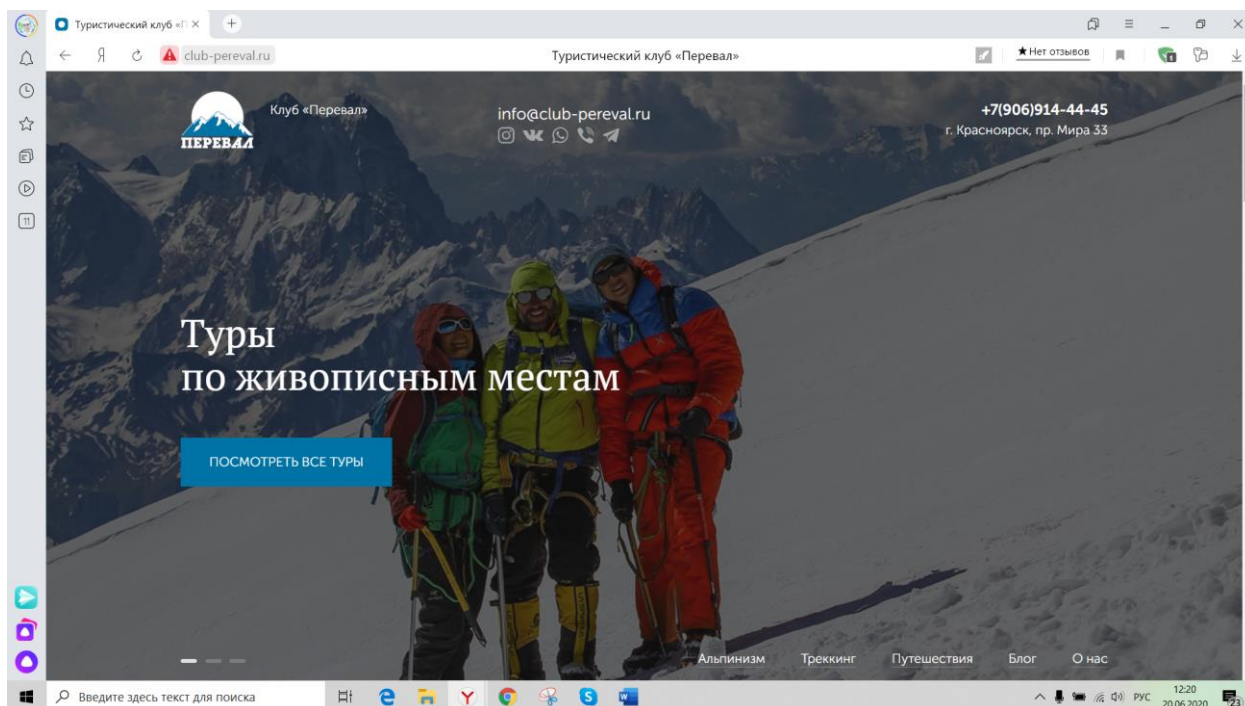


Рисунок 2. Интерфейс туристского клуба

Эта программа позволяет записаться на определенный тур и подробно просмотреть информацию о выбранном туре.

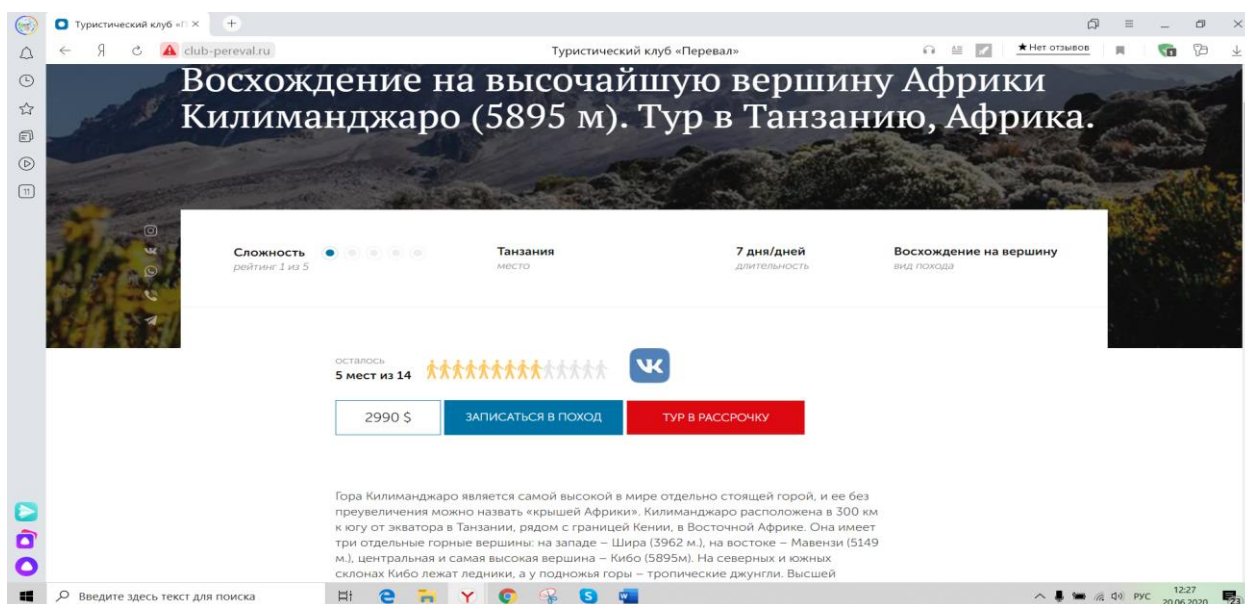


Рисунок 3. Запись на поход.

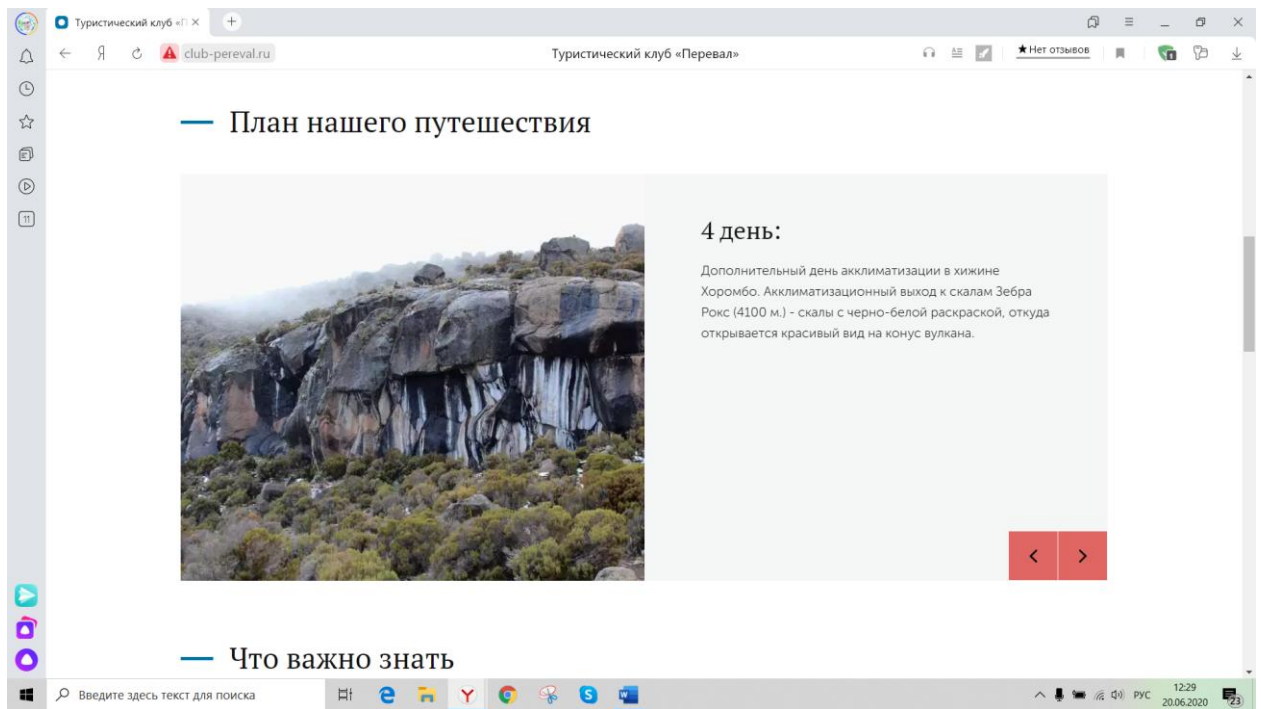


Рисунок 4. План путешествия.

2. Проектная часть.

2.1. Диаграмма прецедентов.

Были построены диаграммы прецедентов для «Туриста», «Директора клуба» и «Инструктора»

Диаграмма прецедентов «Турист» (Рисунок 5) показывает, что турист имеет следующие выборы, такие как: просмотр всех направлений, просмотр своих направлений своего, просмотр всех инструкторов, записаться на новое направление.

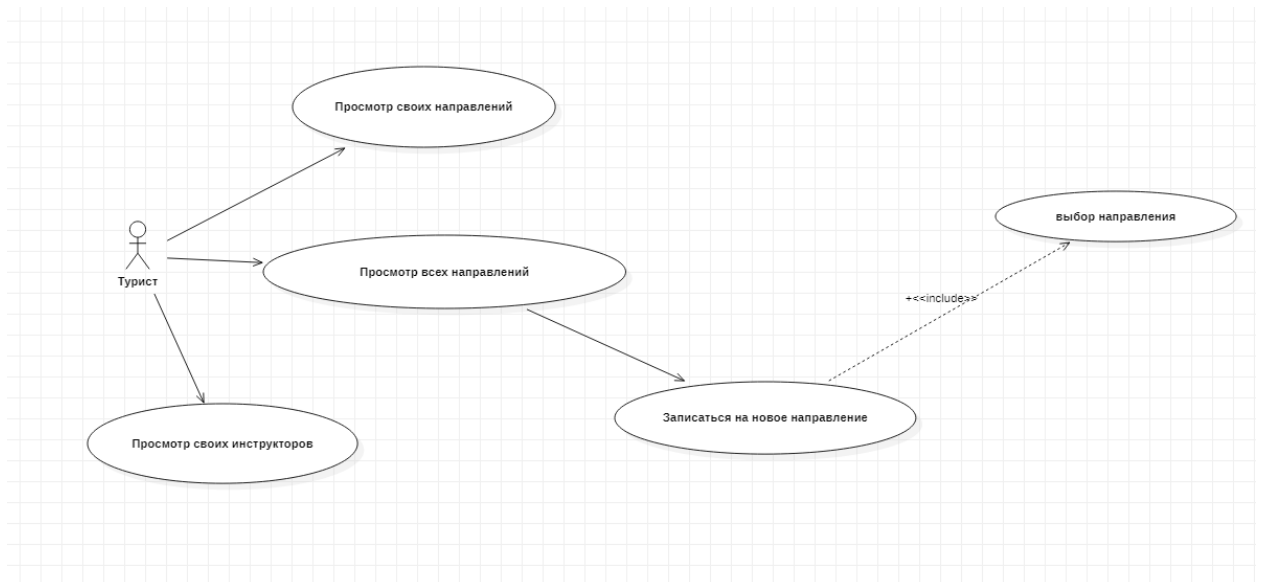


Рисунок 5. Диаграмма прецедентов «Турист»

Диаграмма прецедентов «Директор клуба» (Рисунок 6) показывает, что директор имеет такие выборы, как: просмотр типов отдыха, просмотр всех направлений и просмотр инструкторов.

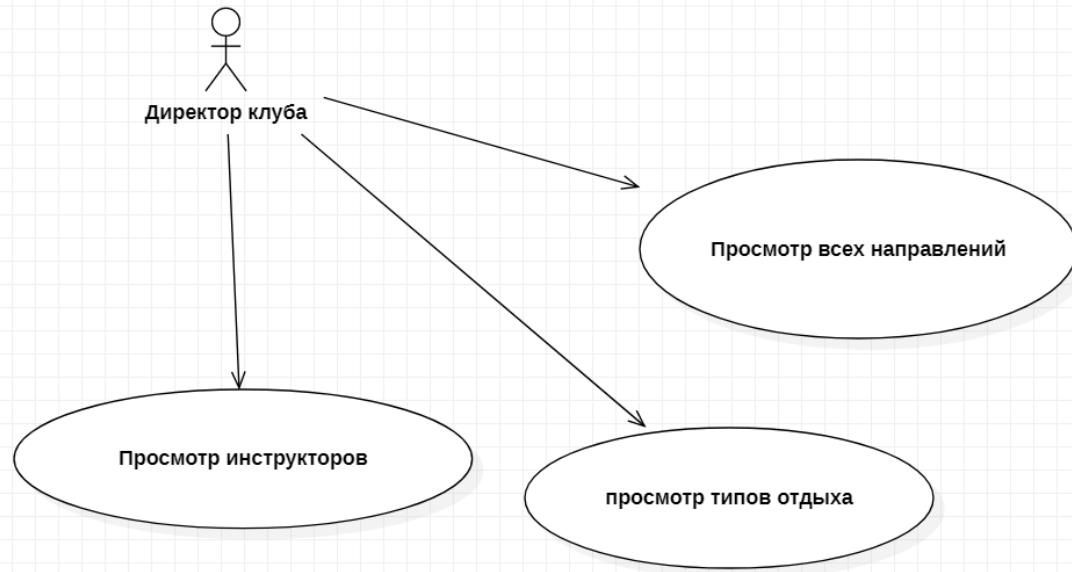


Рисунок 6. Диаграмма прецедентов "Директор клуба"

Диаграмма прецедентов «Инструктор» (Рисунок 7) показывает, что инструктор имеет следующие выборы, такие как: просмотр своего направления, просмотр своих туристов, просмотр типов отдыха.

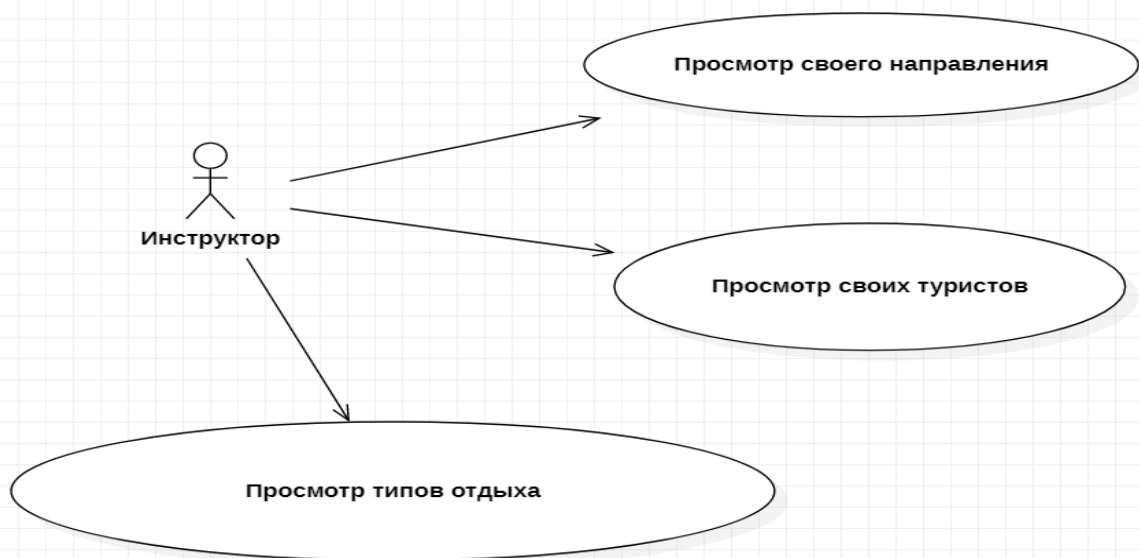


Рисунок 7. Диаграмма прецедентов «Инструктор»

2.2. Выбор инструментов

При выборе инструментов я исходила из некоторых критериев, приведенных в таблице ниже.

Важность критерия я выбирала из: низкая, средняя, высокая.

Таблица 1.

Важность критерия оценивания.

Критерий	Участие в корпоративном проекте	Понятный синтаксис	Наличие официальной документации на русском языке	Скорость разработки.	Кол-во библиотек
Важность критерия	Высокая	Средняя	Средняя	Средняя	Высокая

Исходя из этих критериев я сравнила 2 языка программирования от 0 до 10 баллов за критерий.

Таблица 2.

Сравнение двух языков программирования.

Критерий/ язык программирования	C++	Python
Участие в корпоративном проекте.	10	8
Понятный синтаксис.	9	10
Наличие официальной документации на русском языке.	8	7
Скорость разработки.	5	10
Кол-во библиотек.	6	10
Итого баллов:	38	45

По результатам сравнения для разработки данного курсового проекта был выбран язык программирования Python.

2.3. Проектирование сценария.

В данном разделе приведен сценарий использования программы туристский клуб (Рисунок 8).

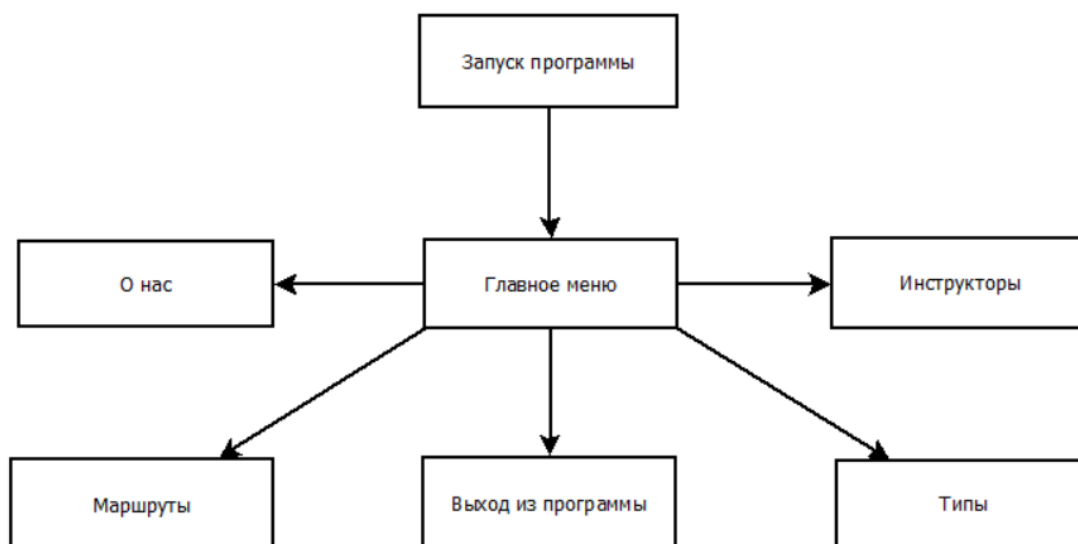


Рисунок 8. Сценарий

Главным окном является «Главное меню». Это первое окно при открытии приложения. Оно содержит в себе четыре кнопки: «Маршруты», «Инструкторы», «Типы» и «О нас». Нажимая на определенную кнопку, открывается соответствующее окно.

2.4. Диаграмма классов.

В данном разделе представлены все классы, использующиеся в проекте, а также их взаимосвязь друг с другом. Данная диаграмма является временной и в дальнейшем в ходе разработки приложения будет изменяться. (Рисунок 9).

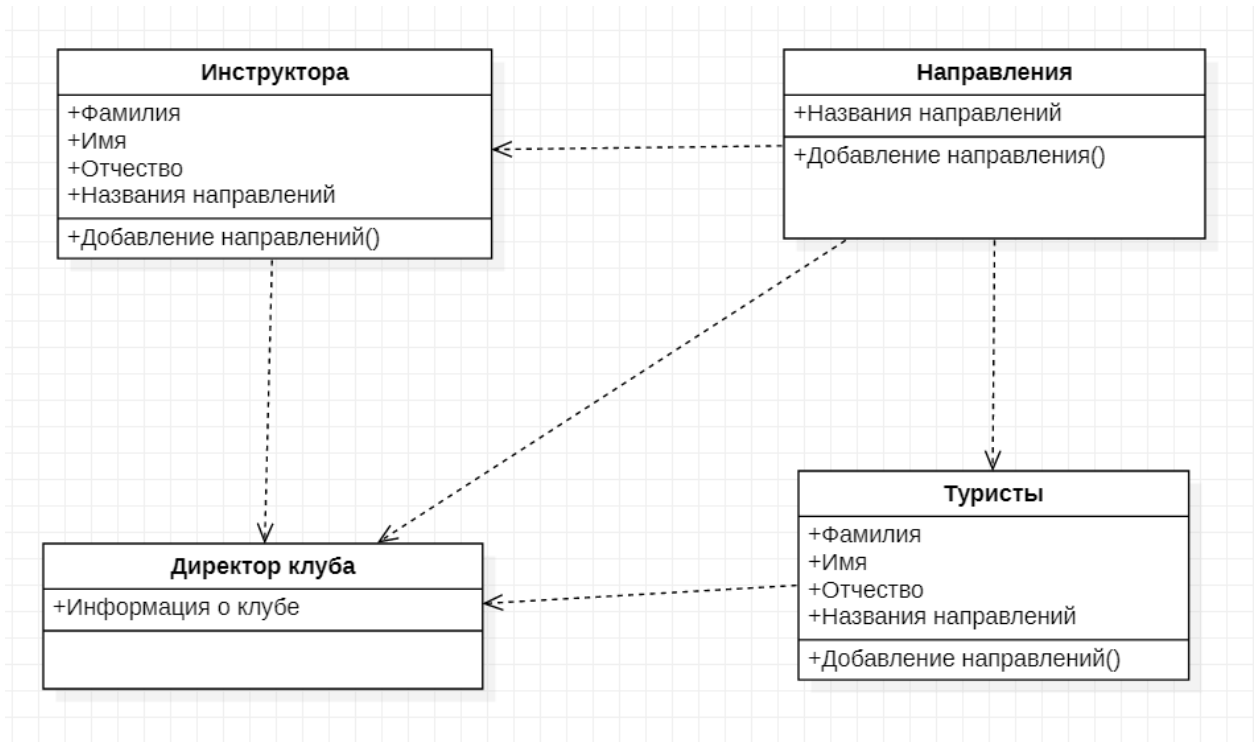


Рисунок 9. Диаграмма классов.

2.5. Описание главного модуля.

Главный модуль состоит из четырех на выбор кнопок и фона. Пользователь нажимает на выбранную им кнопку и в последствии на экран выводится выбранное им окно.

Листинг 1. Главный модуль

```
##Главный модуль программы
from tkinter import * ## импорт самого модуля
from tkinter import messagebox
from tkinter import scrolledtext
from tkinter import ttk
from datetime import datetime
import marshryt, Tipi, instr, Raz

root = Tk() ## корневое окно программы
root.geometry("1280x690")## размеры окна
root.title("Туристский Клуб")## заголовок окна
root.resizable(False, False)
c = Canvas(root, width = 1280, height = 730, bg = 'black')## адресовка
графических примитивов (родительское окно, ширина, высота и цвет фона)
c.pack()

main_menu = Menu()
fon = PhotoImage(file='fon.png')
fonl = c.create_image(640, 400,image = fon)
mar = PhotoImage(file = 'marsh.png')
tip = PhotoImage(file = 'tipi.png')
ins = PhotoImage(file = 'instr.png')
nas = PhotoImage(file = 'o_nas.png')

Button(root, width = 200, height = 100, font = "Batang 20", text =
"Маршрут", command = marshryt.marsh, image = mar).place(x = 750, y =
280)
Button(root, width = 200, height = 100, font = "Batang 20", text =
"Типы", command = Tipi.tt, image = tip).place(x = 750, y = 410)
Button(root, width = 200, height = 100, font = "Batang 20", text =
"Инструкторы", command = instr.inst, image = ins).place(x = 1000, y =
280)
Button(root, width = 200, height = 100, font = "Batang 20", text = "О
нас", command = Raz.razrab, image = nas).place(x = 1000, y = 410)
root.config(menu = main_menu)
root.mainloop()## отображение главного окна
```

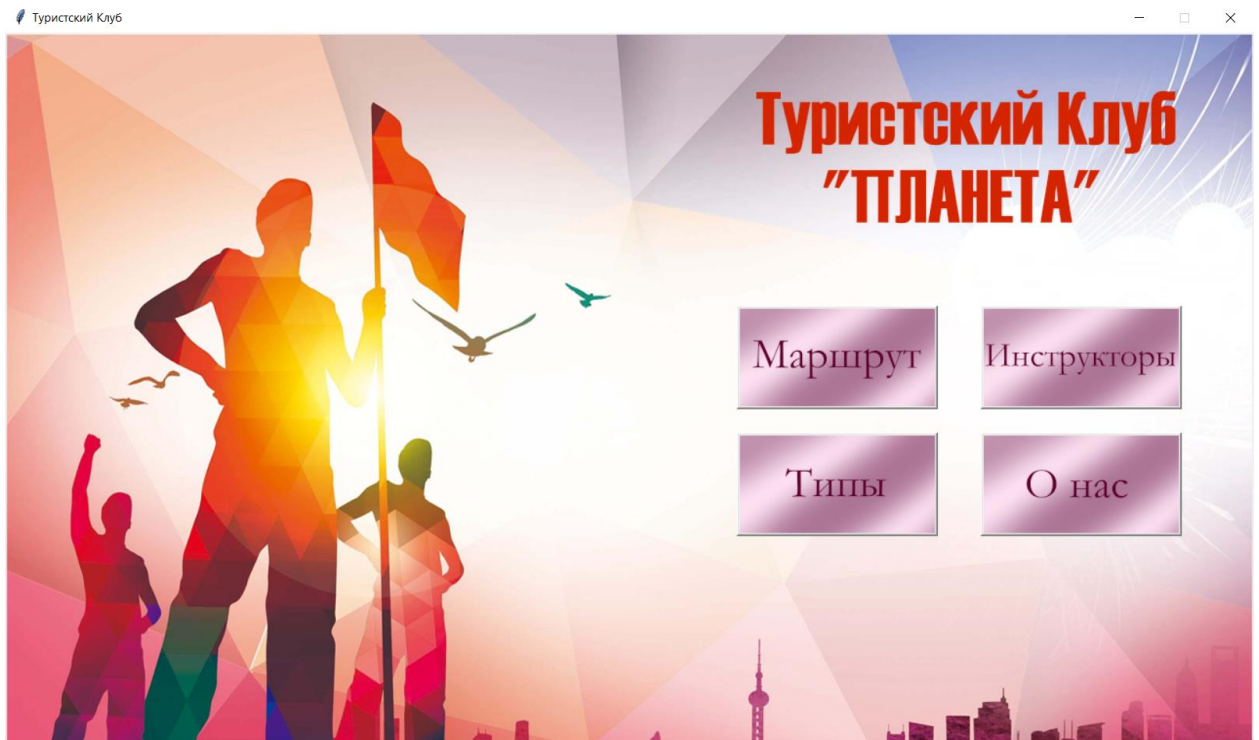


Рисунок 10. Окно главного меню.

2.6. Описание спецификаций к модулям.

В данном разделе описаны спецификации к модулям. Спецификации к модулям заключаются в следующем: использование методов библиотеки Tkinter.

Методы для работы с интерфейсом программы:

1. root.geometry – размер окна.
2. root.title – заголовок окна.
3. root.resizable – метод для изменения окна.
4. canvas – адресовка графических примитивов.
5. place – место объекта на экране.
6. command – переход к команде.
7. PhotoImage – импорт фотографий.
8. Mainloop – отображение главного окна.

2.7. Описание модулей.

В процессе работы над приложением были различные модули. Рассмотрим один из модулей программы.

В данном окне мы можем добавлять и удалять запись о маршрутах.

Листинг 2. Окно «Маршрут»

```
##Этот модуль запускает окно с информацией о маршрутах

from tkinter import *
from tkinter import ttk
import ctypes
import tkinter as tk
import random

def marsh(): ## функция, открывающая окно с информацией о маршрутах
    window = Tk()
    window.resizable(False, False)
    window.title("Маршруты")
    window.geometry('1100x700')

    user32 = ctypes.windll.user32
    widthButtonPlace = (user32.GetSystemMetrics(0) // 15)
    widthButton = widthButtonPlace // 7

    canvas = Canvas(window, width = user32.GetSystemMetrics(0), height
= user32.GetSystemMetrics(1))
    canvas.configure(bg = 'pink')
    canvas.pack()

    f = open('marsh.txt', 'r', encoding = 'utf-8') ## открытие и
чтение файла
    text = Text()
    text = scrolledtext.ScrolledText(window, height = '17', font =
'Batang 20', width = '40')
    text.insert(INSERT, f.read())
    text.place(x = 425, y = 80)
    f.close()

    def clicked_dobavlenie(): ## функция добавления информации
        z = open('marsh.txt', 'r', encoding = 'utf-8')
        y = z.read()
        x = y.split()
        marsh = []
        for i in range (0, len(x), 2):
            u = x[i] + ' ' + x[i+1]
            marsh.append(u)
        b = txt1.get()

        f = open('marsh.txt', 'a', encoding = 'utf-8') ## запись в
файл
        u11 = txt1.get() + ' ' + com.get() + '\n'
        f.write(u11)
```

```

f.close()

f = open('marsh.txt', 'r', encoding = 'utf-8')
text = Text()
text = scrolledtext.ScrolledText(window, height = '17', font =
'Batang 20', width = '40')
text.insert(INSERT, f.read())
text.place(x = 425, y = 80)
f.close()

def clicked_delete(): ## функция удаления информации
    z = open('marsh.txt', 'r', encoding = 'utf-8')
    y = z.read()
    x = y.split()
    marsh = []
    for i in range (len(x)):
        if i % 2 == 0:
            u = x[i] + ' ' + x[i+1]
            marsh.append(u)
    b = txt3.get()
    b1 = com.get()
    v = open('marsh.txt', 'w', encoding = 'utf-8')
    for i in range(len(x)):
        j = marsh[i].split()
        if i%2 == 0 and j[0] != b:
            v.write(j[0])
            v.write(' ')
            v.write(j[1])
            v.write('\n')
    v.close()
    f = open('marsh.txt', 'r', encoding = 'utf-8')
    text = Text()
    text = scrolledtext.ScrolledText(window, height = '17', font =
'Batang 20', width = '40')
    text.insert(INSERT, f.read())
    text.place(x = 425, y = 80)
    f.close()

    lbl1 = Label(window) ## метка
    lbl2 = Label(window)
    lbl1.configure(font = 'Batang 20', text = '    Дата', bg = 'pink',
fg = 'crimson')
    lbl1.place(x = 40, y = 81) ## положение метки
    lbl2.configure(font = 'Batang 20', text = '    Направление', bg =
'pink', fg = 'crimson')
    lbl2.place(x = 170, y = 80)
    txt1 = Entry(window, width = 10, font = 'Batang 20') ## поле ввода
    txt1.place(x = 20, y = 120)
    btn = Button(window, text = "Добавить", font = 'Batang 20',
command = clicked_dobavlenie, bg = 'hotpink') ## кнопка добавления
    btn.place(x = 105, y = 170)

    lbl3 = Label(window) ## метка
    lbl4 = Label(window)

```

```

    lbl3.configure(font = 'Batang 20', text = '    Дата', bg = 'pink',
fg = 'crimson')
    lbl3.place(x = 40, y = 281) ## расположение метки
    lbl4.configure(font = 'Batang 20', text = '    Направление', bg =
'pink', fg = 'crimson')
    lbl4.place(x = 170, y = 280)
    txt3 = Entry(window, width = 10, font = 'Batang 20')
    txt3.place(x = 20, y = 320)
    btn1 = Button(window, text = "Удалить", font = 'Batang 20',
command = clicked_delete, bg = 'hotpink') ## кнопка удаления
    btn1.place(x = 105, y = 370)
    com = ttk.Combobox(window) ## выпадающий список
    com.place(x = 190, y = 120, height = 37, width = 155)
    com['values'] = ['Байкал', 'Тайга', 'Горный калейдоскоп', 'Плато
Путорана', 'Камчатка', 'Восхождения на Эльбрус', 'Каньон реки Лава',
'Восхождение на вулкан Ключевая сопка', 'Велопоход вдоль Оки', 'Вокруг
Эльбруса за 9 дней']
    com.current(0)
    com2 = ttk.Combobox(window)
    com2.place(x = 190, y = 320, height = 37, width = 155)
    com2['values'] = ['Байкал', 'Тайга', 'Горный калейдоскоп', 'Плато
Путорана', 'Камчатка', 'Восхождения на Эльбрус', 'Каньон реки Лава',
'Восхождение на вулкан Ключевая сопка', 'Велопоход вдоль Оки', 'Вокруг
Эльбруса за 9 дней']
    com2.current(0)

```

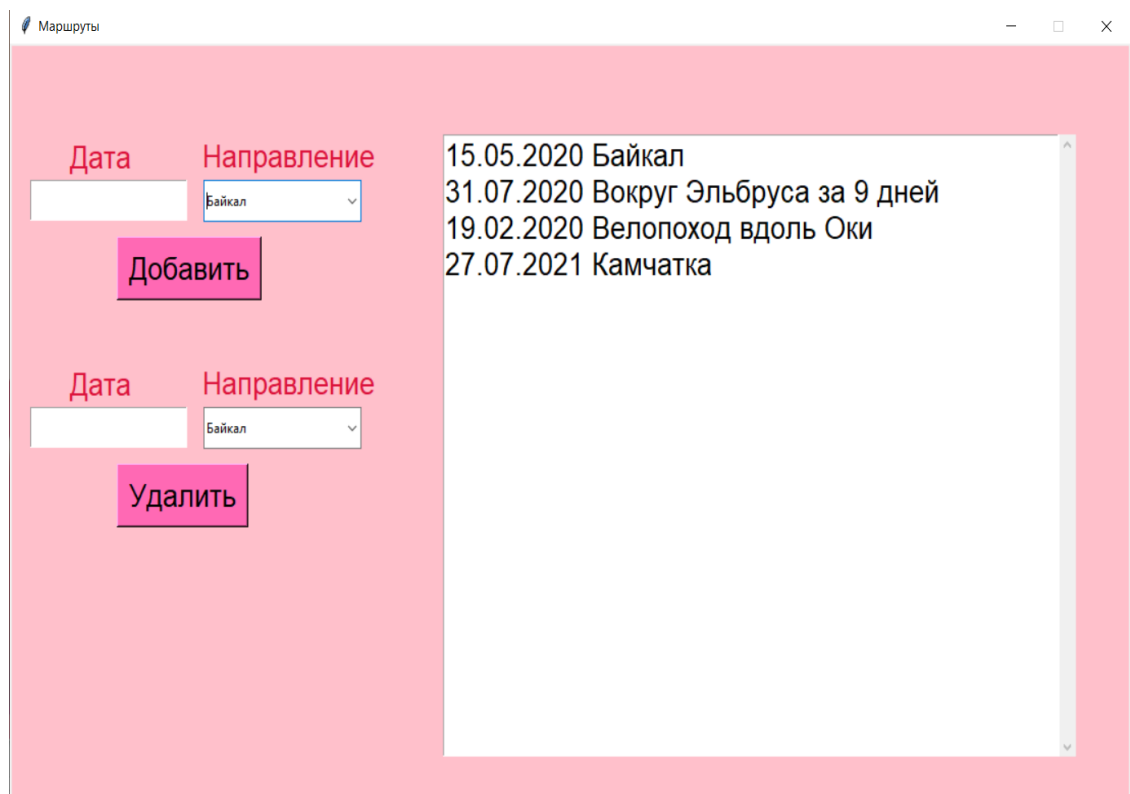


Рисунок 11. Окно «Маршруты»

2.8. Описание тестовых наборов модулей.

На рисунках ниже приведены результаты работы добавления записи.

Таблица до добавления новой записи показана на рисунке 12

The screenshot shows a window titled 'Инструкторы' (Instructors) with a pink background. On the left, there are two sets of input fields for 'Имя' (Name) and 'Фамилия' (Surname), each followed by a pink button labeled 'Добавить' (Add) and 'Удалить' (Delete) respectively. On the right, a large white text area contains a list of names: 'Инна Варнасова', 'Захар Кузнецов', and 'Анастасия Ивлеева'.

Рисунок 12. До добавления новой записи.

The screenshot shows the same 'Инструкторы' (Instructors) window, but now the list of names on the right includes an additional entry: 'Михаил Громов'. The input fields and buttons on the left remain the same.

Рисунок 13. После добавления записи.

2.9. Описание применения средств отладки.

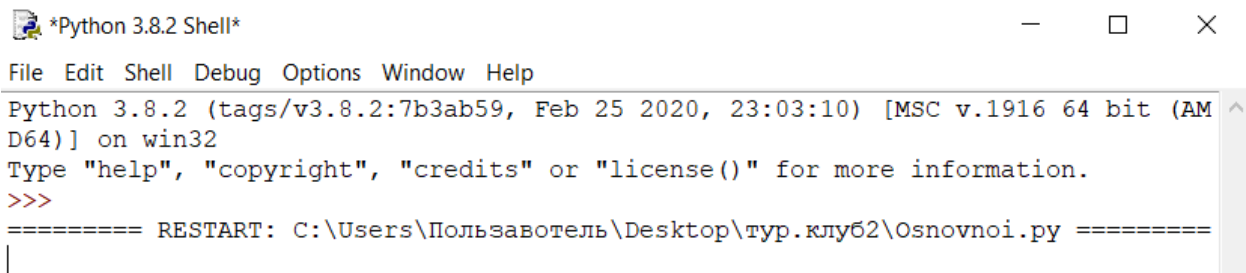
В этом разделе показано умение применять средства отладки.

В ходе написания курсового проекта при попытке запустить код, было получено сообщение с ошибкой:

```
Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 23:03:10) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\Users\Пользователь\Desktop\тур.клуб2\Osnovnoi.py =====
Traceback (most recent call last):
  File "C:\Users\Пользователь\Desktop\тур.клуб2\Osnovnoi.py", line 11, in <module>
    root.titli("Туристский Клуб")## заголовок окна
  File "C:\Program Files\Python38\lib\tkinter\__init__.py", line 2345, in __getattr__
    return getattr(self.tk, attr)
AttributeError: '_tkinter.tkapp' object has no attribute 'titli'
>>> |
```

Рисунок 14. До применения отладки.

После получения данного сообщения была обнаружена ошибка, которая впоследствии была устранена, а после попытки запуска скрипта получено данное сообщение:



```
*Python 3.8.2 Shell*
File Edit Shell Debug Options Window Help
Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 23:03:10) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\Users\Пользователь\Desktop\тур.клуб2\Osnovnoi.py =====
|
```

Рисунок 15. После применения средств отладки.

3. Эксплуатационная часть.

3.1. Руководство оператора.

3.1.1. Назначение программы.

Разработанное приложение «Туристский клуб» используется для управления маршрутом, редактирования инструкторов.

3.1.2. Условия выполнения программы.

- ОС: Windows 10
- Место на диске: 10 MB

3.1.3. Выполнение программы.

Перед тем как начать работать с программой, её следует запустить.

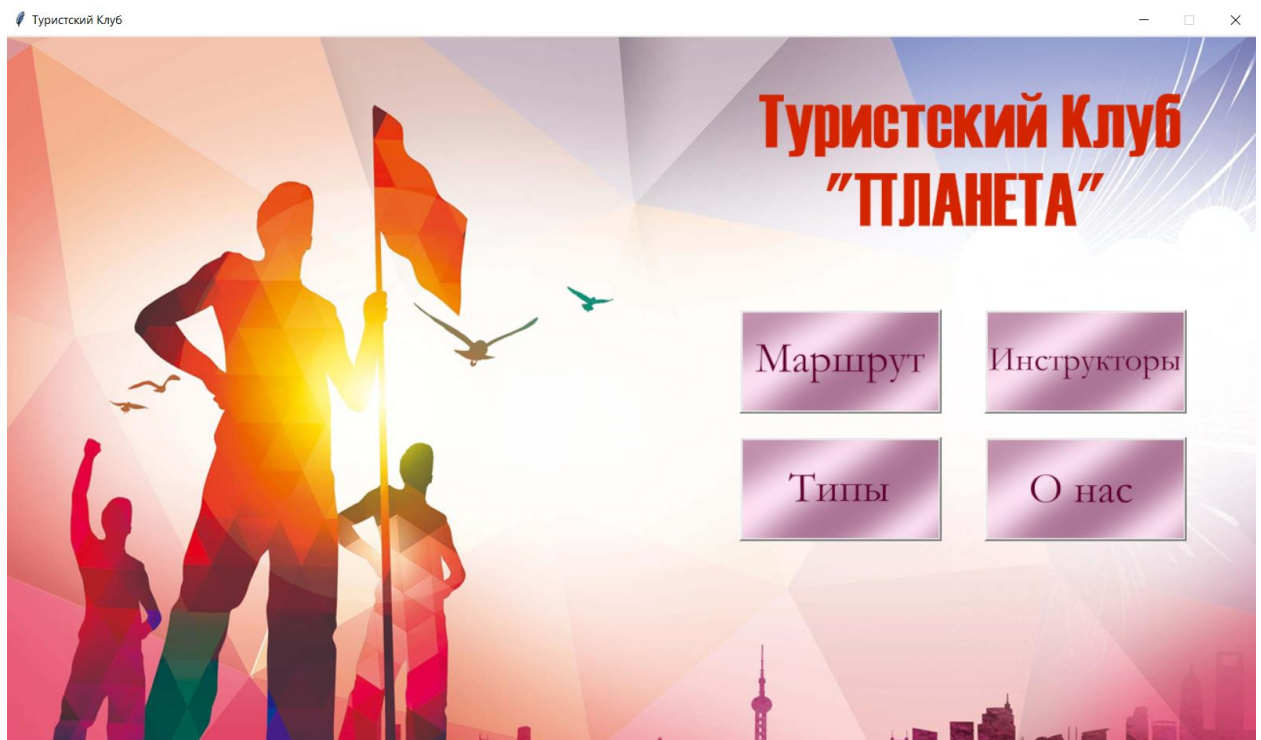


Рисунок 16. Главное меню программы.

После запуска программы открывается окно выбора (Рисунок 16), содержащее 4 кнопки: «Маршрут», «Типы», «Инструкторы», «О нас».

При нажатии кнопки «О нас» открывается окно, содержащее информацию о разработчике, а также о самой программе (Рисунок 17).

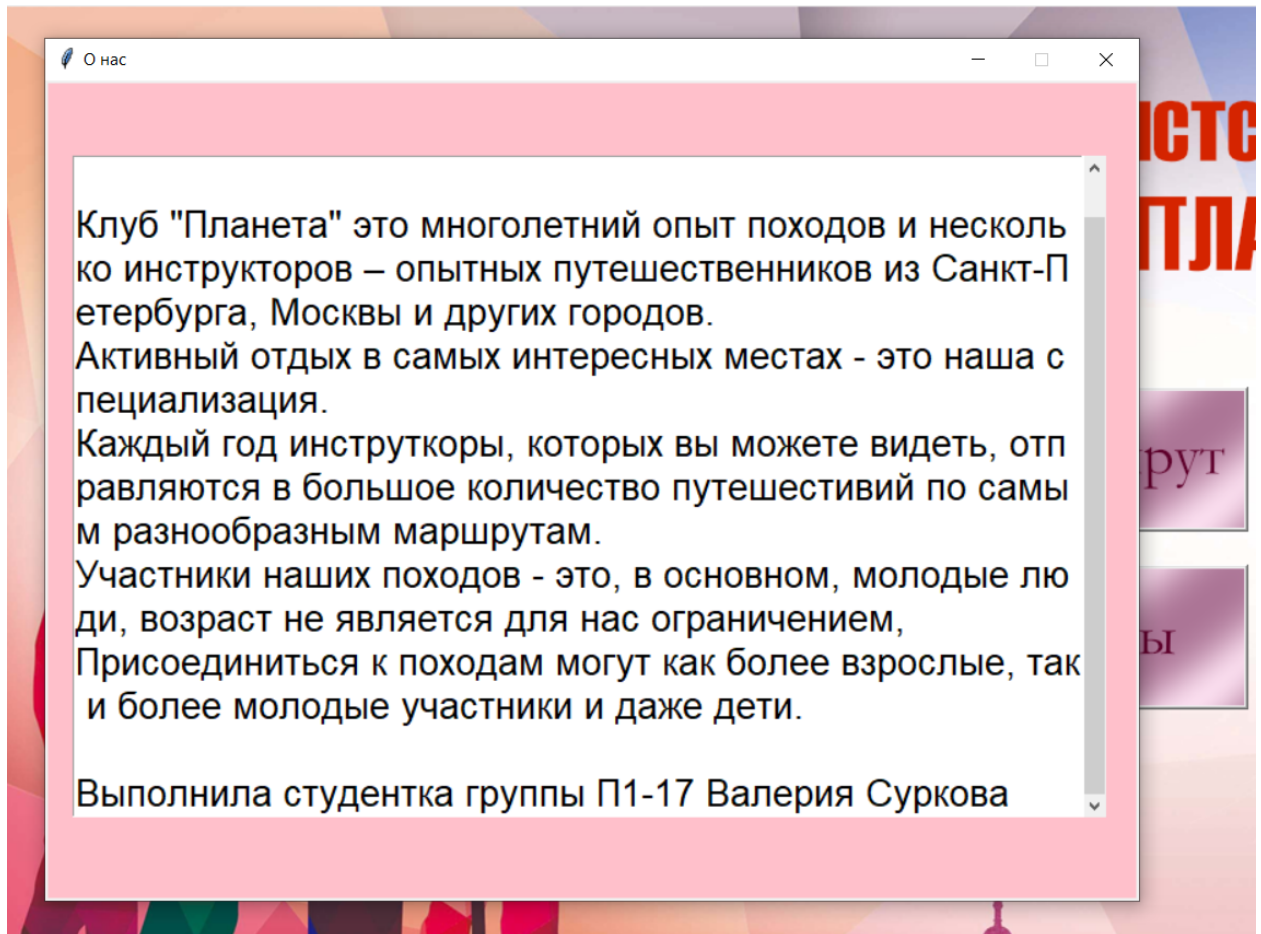


Рисунок 17. Окно «О нас»

При нажатии на одну из всех предложенных кнопок, открывается соответствующее окно.

Кнопка «Маршрут» откроет окно «Маршруты» с соответствующими данными о маршрутах (Рисунок 18).

Маршруты

Дата Направление

Добавить

Дата Направление

Удалить

15.05.2020 Байкал Эльбруса за 9 дней
 19.02.2020 Велопоход вдоль Оки
 27.07.2021 Камчатка
 26.11.2019 Каньон реки Лава

Рисунок 18. Окно «Маршруты»

Кнопка «Типы» откроет окно «Типы» с соответствующими данными.
 (Рисунок 19)

Типы

Стоимость Тип отдыха

Добавить

Стоимость Тип отдыха

Удалить

20000 Яхтинг
 7000 Восхождения
 10000 Скалолазание
 5000 Рыбалка

Рисунок 19. Окно «Типы»

Кнопка «Инструкторы» откроет окно «Инструкторы» с соответствующими данными (Рисунок 20).

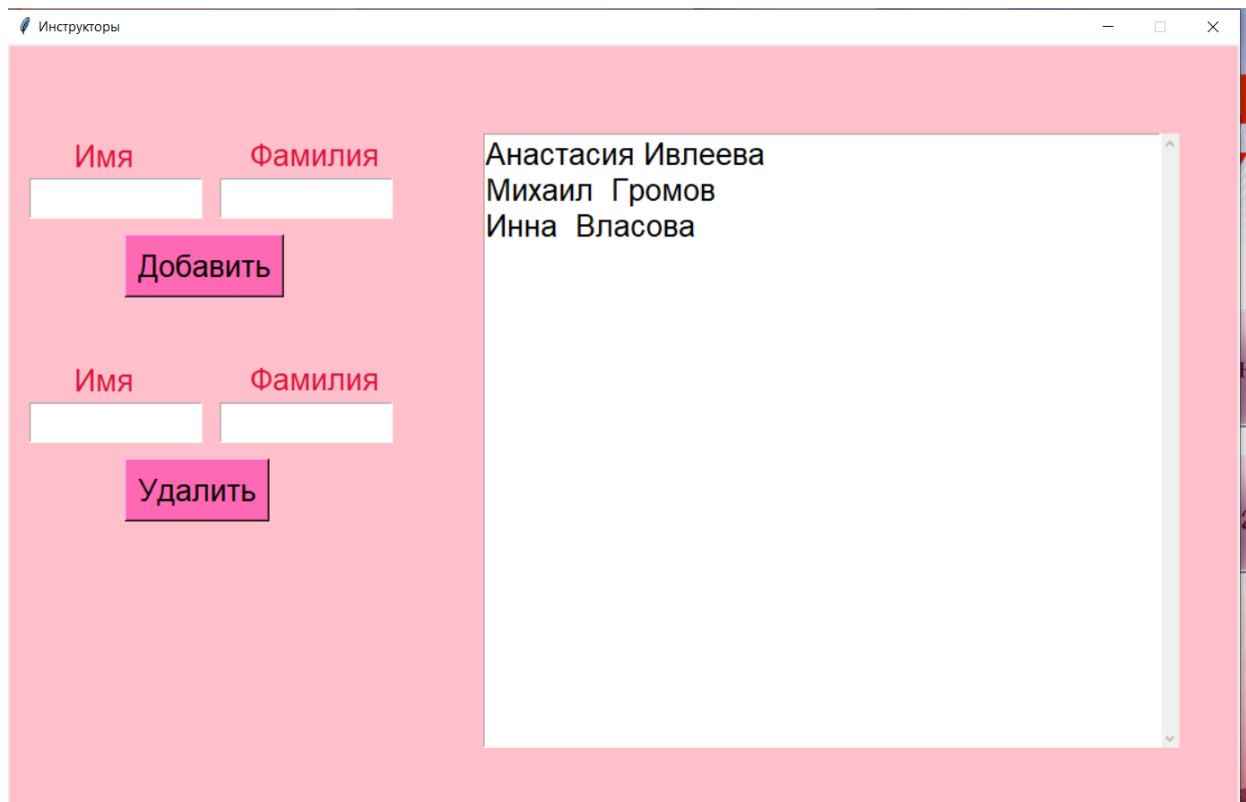


Рисунок 20. Окно «Инструкторы»

Каждое из окон имеет примерно одинаковый интерфейс. При нажатии на кнопку «Закреть окно» открытое окно закрывается и пользователю открывается окно главного экрана.

Заключение.

В результате выполнения курсового проекта была разработана программа «Туристский клуб».

Данная программа может быть использованная в учебных целях.

В качестве доработки программы будет осуществлено дальнейшее продолжение работы над проектом. Планируется изменить интерфейс программы и добавить окно с самыми популярными маршрутами.

Список литературы и интернет – источников.

1. Туристский Клуб "Aqua Natura":
<https://club-aquanatura.1c-umi.ru/otzyvy/>
2. Клуб «Перевал» Туры по живописным и интересным местам:
<http://club-pereval.ru/afisha>
3. Лаборатория линуксоида:
<https://younglinux.info/tkinter/widget.php>
4. YouTube канал IT Hobbies:
<https://www.youtube.com/channel/UCXE4vvggmw3gpu59D-QSLjA/playlists>
5. Python. Работа с изображениями:
<https://wiki.programstore.ru/python-rabota-s-izobrazheniyami/>
6. YouTube канал «Изучаем мир ИТ/ Олег Шпагин /WISEPLAT
https://www.youtube.com/watch?v=K_le5u6k8gM&list=PLxiU3nwEQ4PHtLT8LzGAdcuze-laotXps

Приложение 1. Главный модуль.

```

##Главный модуль программы
from tkinter import * ## импорт самого модуля
from tkinter import messagebox
from tkinter import scrolledtext
from tkinter import ttk
from datetime import datetime
import marshryt, Tipi, instr, Raz

root = Tk() ## корневое окно программы
root.geometry("1280x690")## размеры окна
root.title("Туристский Клуб")## заголовок окна
root.resizable(False, False)
c = Canvas(root, width = 1280, height = 730, bg = 'black')## адресовка
графических примитивов (родительское окно, ширина, высота и цвет фона)
c.pack()

main_menu = Menu()
fon = PhotoImage(file='fon.png')
fonl = c.create_image(640, 400,image = fon)
mar = PhotoImage(file = 'marsh.png')
tip = PhotoImage(file = 'tipi.png')
ins = PhotoImage(file = 'instr.png')
nas = PhotoImage(file = 'o_nas.png')

Button(root, width = 200, height = 100, font = "Batang 20", text =
"Маршрут", command = marshryt.marsh, image = mar).place(x = 750, y =
280)
Button(root, width = 200, height = 100, font = "Batang 20", text =
"Типы", command = Tipi.tt, image = tip).place(x = 750, y = 410)
Button(root, width = 200, height = 100, font = "Batang 20", text =
"Инструкторы", command = instr.inst, image = ins).place(x = 1000, y =
280)
Button(root, width = 200, height = 100, font = "Batang 20", text = "О
нас", command = Raz.razrab, image = nas).place(x = 1000, y = 410)
root.config(menu = main_menu)
root.mainloop()## отображение главного окна

```

Приложение 2. Окно «Маршруты»

```

##Этот модуль запускает окно с информацией о маршрутах

from tkinter import *
from tkinter import ttk
import ctypes
import tkinter as tk
import random

def marsh(): ## функция, открывающая окно с информацией о маршрутах
    window = Tk()
    window.resizable(False, False)
    window.title("Маршруты")
    window.geometry('1100x700')

    user32 = ctypes.windll.user32
    widthButtonPlace = (user32.GetSystemMetrics(0) // 15)
    widthButton = widthButtonPlace // 7

    canvas = Canvas(window, width = user32.GetSystemMetrics(0), height
= user32.GetSystemMetrics(1))
    canvas.configure(bg = 'pink')
    canvas.pack()

    f = open('marsh.txt', 'r', encoding = 'utf-8') ## открытие и
чтение файла
    text = Text()
    text = scrolledtext.ScrolledText(window, height = '17', font =
'Batang 20', width = '40')
    text.insert(INSERT, f.read())
    text.place(x = 425, y = 80)
    f.close()

def clicked_dobavlenie(): ## функция добавления информации
    z = open('marsh.txt', 'r', encoding = 'utf-8')
    y = z.read()
    x = y.split()
    marsh = []
    for i in range (0, len(x), 2):
        u = x[i] + ' ' + x[i+1]
        marsh.append(u)
    b = txt1.get()

    f = open('marsh.txt', 'a', encoding = 'utf-8') ## запись в
файл
    u11 = txt1.get() + ' ' + com.get() + '\n'
    f.write(u11)
    f.close()

```

```

f = open('marsh.txt', 'r', encoding = 'utf-8')
text = Text()
text = scrolledtext.ScrolledText(window, height = '17', font =
'Batang 20', width = '40')
text.insert(INSERT, f.read())
text.place(x = 425, y = 80)
f.close()

```

```

def clicked_delete(): ## функция удаления информации
    z = open('marsh.txt', 'r', encoding = 'utf-8')
    y = z.read()
    x = y.split()
    marsh = []
    for i in range (len(x)):
        if i % 2 == 0:
            u = x[i] + ' ' + x[i+1]
            marsh.append(u)
    b = txt3.get()
    b1 = com.get()
    v = open('marsh.txt','w', encoding = 'utf-8')
    for i in range(len(x)):
        j = marsh[i].split()
        if i%2 == 0 and j[0] != b:
            v.write(j[0])
            v.write(' ')
            v.write(j[1])
            v.write('\n')
    v.close()
    f = open('marsh.txt', 'r', encoding = 'utf-8')
    text = Text()
    text = scrolledtext.ScrolledText(window, height = '17', font =
'Batang 20', width = '40')
    text.insert(INSERT, f.read())
    text.place(x = 425, y = 80)
    f.close()

```

```

lbl1 = Label(window) ## метка
lbl2 = Label(window)
lbl1.configure(font = 'Batang 20', text = '    Дата', bg = 'pink',
fg = 'crimson')
lbl1.place(x = 40, y = 81) ## положение метки
lbl2.configure(font = 'Batang 20', text = '    Направление', bg =
'pink', fg = 'crimson')
lbl2.place(x = 170, y = 80)
txt1 = Entry(window, width = 10, font = 'Batang 20') ## поле ввода
txt1.place(x = 20, y = 120)
btn = Button(window, text = "Добавить", font = 'Batang 20',
command = clicked_dobavlenie, bg = 'hotpink') ## кнопка добавления
btn.place(x = 105, y = 170)

lbl3 = Label(window) ## метка

```

```

    lbl4 = Label(window)
    lbl3.configure(font = 'Batang 20', text = '    Дата', bg = 'pink',
fg = 'crimson')
    lbl3.place(x = 40, y = 281) ## расположение метки
    lbl4.configure(font = 'Batang 20', text = '    Направление', bg =
'pink', fg = 'crimson')
    lbl4.place(x = 170, y = 280)
    txt3 = Entry(window, width = 10, font = 'Batang 20')
    txt3.place(x = 20, y = 320)
    btn1 = Button(window, text = "Удалить", font = 'Batang 20',
command = clicked_delete, bg = 'hotpink') ## кнопка удаления
    btn1.place(x = 105, y = 370)
    com = ttk.Combobox(window) ## выпадающий список
    com.place(x = 190, y = 120, height = 37, width = 155)
    com['values'] = ['Байкал', 'Тайга', 'Горный калейдоскоп', 'Плато
Путорана', 'Камчатка', 'Восхождения на Эльбрус', 'Каньон реки Лава',
'Восхождение на вулкан Ключевая сопка', 'Велопоход вдоль Оки', 'Вокруг
Эльбруса за 9 дней']
    com.current(0)
    com2 = ttk.Combobox(window)
    com2.place(x = 190, y = 320, height = 37, width = 155)
    com2['values'] = ['Байкал', 'Тайга', 'Горный калейдоскоп', 'Плато
Путорана', 'Камчатка', 'Восхождения на Эльбрус', 'Каньон реки Лава',
'Восхождение на вулкан Ключевая сопка', 'Велопоход вдоль Оки', 'Вокруг
Эльбруса за 9 дней']
    com2.current(0)

```


Приложение 3. Окно «Инструкторы»

```

##Этот модуль запускает окно со списком инструкторов

from tkinter import *
from tkinter import scrolledtext
import ctypes

def inst(): ##функция, открывающая окно со списком инструкторов
    window = Tk()
    window.resizable(False, False)
    window.title("Инструкторы") ## название окна
    window.geometry('1100x700') ## размер окна

    user32 = ctypes.windll.user32
    widthButtonPlace = (user32.GetSystemMetrics(0) // 15)
    widthButton = widthButtonPlace // 7

    canvas = Canvas(window, width = user32.GetSystemMetrics(0), height
= user32.GetSystemMetrics(1))
    canvas.configure(bg = 'pink') ## фон окна
    canvas.pack()

    f = open('inst.txt', 'r', encoding = 'utf-8') ## открытие и чтение
файла
    text = Text()
    text = scrolledtext.ScrolledText(window, height = '17', font =
'Batang 20', width = '40')
    text.insert(INSERT, f.read())
    text.place(x = 425, y = 80)
    f.close() ## закрытие файла

def clicked(): ## функция добавления информации
    z = open('inst.txt', 'r', encoding = 'utf-8')
    y = z.read()
    x = y.split()
    t = []
    for i in range (len(x)):
        if i%2 == 0:
            u = x[i] + ' ' + x[i+1]
            t.append(u) ## добавление информации
    b = txt1.get()
    b1 = txt2.get()

    f = open('inst.txt', 'a', encoding = 'utf-8') ## запись в файл
    u11 = txt1.get() + ' ' + txt2.get() + '\n'
    f.write(u11)
    f.close()

    f = open('inst.txt', 'r', encoding = 'utf-8')
    text = Text()

```

```

        text = scrolledtext.ScrolledText(window, height = '17', font =
'Batang 20', width = '40')
        text.insert(INSERT, f.read())
        text.place(x = 425, y = 80)
        f.close()

```

```

def clicked1(): ## функция удаления записи
    z = open('inst.txt', 'r', encoding = 'utf-8')
    y = z.read()
    x = y.split()
    t = []
    for i in range (len(x)):
        if i%2 == 0:
            u = x[i] + ' ' + x[i+1]
            t.append(u)
    b = txt3.get()
    b1 = txt4.get()
    v = open('inst.txt','w', encoding = 'utf-8')
    for i in range(len(x)): ## удаления информации
        j = t[i].split()
        if i%2 == 0 and j[0] != b:
            v.write(j[0])
            v.write(' ')
            v.write(j[1])
            v.write('\n')
    v.close()
    f = open('inst.txt', 'r', encoding = 'utf-8')
    text = Text()
    text = scrolledtext.ScrolledText(window, height = '17', font =
'Batang 20', width = '40')
    text.insert(INSERT, f.read())
    text.place(x = 425, y = 80)
    f.close()

```

```

    lbl1 = Label(window) ## метка
    lbl2 = Label(window)
    lbl1.configure(font = 'Batang 20', text = '      Имя', bg = 'pink',
fg = 'crimson') ## шрифт, название и цвет метки
    lbl1.place(x = 25, y = 81) ## положение метки
    lbl2.configure(font = 'Batang 20', text = '      Фамилия', bg =
'pink', fg = 'crimson')
    lbl2.place(x = 190, y = 80)
    txt1 = Entry(window, width = 10, font = 'Batang 20') ## поле ввода
    txt1.place(x = 20, y = 120)
    txt2 = Entry(window, width = 10, font = 'Batang 20')
    txt2.place(x = 190, y = 120)
    btn = Button(window, text = "Добавить", font = 'Batang 20',
command=clicked, bg = 'hotpink') ## кнопка добавления
    btn.place(x = 105, y = 170)

    lbl3 = Label(window) ## метка
    lbl4 = Label(window)

```

```

    lbl3.configure(font = 'Batang 20', text = '    Имя', bg = 'pink',
fg = 'crimson')
    lbl3.place(x = 25, y = 281)
    lbl4.configure(font = 'Batang 20', text = '    Фамилия', bg =
'pink', fg = 'crimson')
    lbl4.place(x = 190, y = 280)
    txt3 = Entry(window, width = 10, font = 'Batang 20') ## поле ввода
    txt3.place(x = 20, y = 320)
    txt4 = Entry(window, width = 10, font = 'Batang 20')
    txt4.place(x = 190, y = 320)
    btn1 = Button(window, text = "Удалить", font = 'Batang 20',
command=clicked1, bg = 'hotpink') ##кнопка удаления
    btn1.place(x = 105, y = 370)

```

Приложения 4. Окно «Типы»

```

## модуль запускает окно с данными о типах отдыха

from tkinter import *
from tkinter import ttk
from tkinter import scrolledtext
import ctypes
import tkinter as tk
import random

def tt(): ##функция, открывающая окно с данными о типах отдыха
    window = Tk()
    window.resizable(False, False)
    window.title("Типы")
    window.geometry('1100x700')

    user32 = ctypes.windll.user32
    widthButtonPlace = (user32.GetSystemMetrics(0) // 15)
    widthButton = widthButtonPlace // 7

    canvas = Canvas(window, width = user32.GetSystemMetrics(0), height
= user32.GetSystemMetrics(1))
    canvas.configure(bg = 'pink') ## фон окна
    canvas.pack()

    f = open('tipi.txt', 'r', encoding = 'utf-8') ## открытие и чтение
файла
    text = Text()
    text = scrolledtext.ScrolledText(window, height = '17', font =
'Batang 20', width = '40')
    text.insert(INSERT, f.read())
    text.place(x = 425, y = 80)
    f.close()

def clicked_dobavlenie(): ## функция добавления информации
    z = open('tipi.txt', 'r', encoding = 'utf-8')
    y = z.read()
    x = y.split()
    tipi = []
    for i in range (0, len(x), 2):
        u = x[i] + ' ' + x[i+1]
        tipi.append(u) ## добавление информации
    b = txt1.get()

    f = open('tipi.txt', 'a', encoding = 'utf-8')##запись в файл
    u11 = txt1.get() + ' ' + com.get() + '\n'
    f.write(u11)
    f.close()

```

```

f = open('tipi.txt', 'r', encoding = 'utf-8')
text = Text()
text = scrolledtext.ScrolledText(window, height = '17', font =
'Batang 20', width = '40')
text.insert(INSERT, f.read())
text.place(x = 425, y = 80)
f.close()

```

```

def clicked_delete(): ## функция удаления информации
    z = open('tipi.txt', 'r', encoding = 'utf-8')
    y = z.read()
    x = y.split()
    tipi = []
    for i in range (len(x)):
        if i % 2 == 0:
            u = x[i] + ' ' + x[i+1]
            tipi.append(u)
    b = txt3.get()
    b1 = com.get()
    v = open('tipi.txt','w', encoding = 'utf-8')
    for i in range(len(x)): ## удаление информации
        j = tipi[i].split()
        if i%2 == 0 and j[0] != b:
            v.write(j[0])
            v.write(' ')
            v.write(j[1])
            v.write('\n')
    v.close()
    f = open('tipi.txt', 'r', encoding = 'utf-8')
    text = Text()
    text = scrolledtext.ScrolledText(window, height = '17', font =
'Batang 20', width = '40')
    text.insert(INSERT, f.read())
    text.place(x = 425, y = 80)
    f.close()

```

```

lbl1 = Label(window) ## метка
lbl2 = Label(window)
lbl1.configure(font = 'Batang 20', text = '    Стоимость', bg =
'pink', fg = 'crimson')
lbl1.place(x = 10, y = 81)
lbl2.configure(font = 'Batang 20', text = '    Тип отдыха', bg =
'pink', fg = 'crimson')
lbl2.place(x = 180, y = 80) ## положение метки
txt1 = Entry(window, width = 10, font = 'Batang 20') ## поле ввода
txt1.place(x = 20, y = 120)
btn = Button(window, text = "Добавить", font = 'Batang 20',
command = clicked_dobavlenie, bg = 'hotpink') ## кнопка добавления
btn.place(x = 105, y = 170)

lbl3 = Label(window)

```

```

    lbl4 = Label(window)
    lbl3.configure(font = 'Batang 20', text = '    Стоимость', bg =
'pink', fg = 'crimson')
    lbl3.place(x = 10, y = 281)
    lbl4.configure(font = 'Batang 20', text = '    Тип отдыха', bg =
'pink', fg = 'crimson')
    lbl4.place(x = 180, y = 280)
    txt3 = Entry(window, width = 10, font = 'Batang 20')
    txt3.place(x = 20, y = 320)
    btn1 = Button(window, text = "Удалить", font = 'Batang 20',
command = clicked_delete, bg = 'hotpink')
    btn1.place(x = 105, y = 370)
    com = ttk.Combobox(window) ## выпадающий список
    com.place(x = 190, y = 120, height = 37, width = 155)
    com['values'] = ['Рюкзаки', 'Скалолазание', 'Рыбалка', 'Каяки',
'Восхождения', 'Яхтинг', 'Охота']
    com.current(0)
    com2 = ttk.Combobox(window)
    com2.place(x = 190, y = 320, height = 37, width = 155)
    com2['values'] = ['Рюкзаки', 'Скалолазание', 'Рыбалка', 'Каяки',
'Восхождения', 'Яхтинг']
    com2.current(0)

```

Приложение 5. Окно «О нас»

```

## этот модуль запускает информацию о клубе

from tkinter import *
import ctypes

def razrab(): ## функция вывода информации на экран о клубе
    window = Tk()
    window.resizable(False, False)
    window.title("О нас") ## название окна
    user32 = ctypes.windll.user32
    widthButtonPlace = (user32.GetSystemMetrics(0) // 15)
    widthButton = widthButtonPlace // 7
    f = open('proga.txt', 'r', encoding='utf-8') ## открытие и чтение
    файла
    canvas = Canvas(window, width = user32.GetSystemMetrics(0), height
= user32.GetSystemMetrics(1))
    canvas.configure(bg = 'pink')
    canvas.pack()
    window.geometry('800x600')
    txt = scrolledtext.ScrolledText(window, height = '15', font =
'Batang 20', width = '49')
    txt.insert(INSERT, f.read())
    txt.place(x = 20, y = 55)
    window.mainloop()

```