

Министерство образования Республики Беларусь
Учреждение образования
«Белорусский государственный университет информатики и радиоэлектроники»
Факультет компьютерных систем и сетей
Кафедра информатики

Курсовой проект по дисциплине:
«Программирование»

Пояснительная записка к курсовой работе

Тема работы:
«Погодный информер»

Исполнитель
студент гр. 653502

(подпись дата)

Павлюков С.Е.

Руководитель

(подпись дата)

Козуб В.Н.

(оценка)

Минск
2017 год
Содержание

Введение	3
1. С#	
1.1 Описание языка	3
2. Работа программы	
2.1 Взаимодействие с API	4
2.2 Структура программы.	5
2.3 Описание работы программы	6
2.4 Алгоритм программы	7
Выводы	10
Список использованных источников	10

Введение

Каждый раз, выходя из дома, нас интересует какая погода за окном. Чтобы найти эту информацию, мы обычно используем поисковики в интернете. Эта программа предназначена для того, чтобы сократить время поиска и показывать только самое необходимое. С этой целью в программе используется весьма минималистичный интерфейс.

Вся информация о погоде берется с сайта <https://openweathermap.org/>, предоставляющего API.

В качестве языка программирования используется C#. Так же используется фреймворк Json.NET для дисериализации информации в формате Json.

1. C#

1.1 Описание языка

C# — объектно-ориентированный язык программирования. Разработан в 1998—2001 годах в компании Microsoft как язык разработки приложений для платформы Microsoft .NET Framework.

C# относится к семье языков с C-подобным синтаксисом, из них его синтаксис наиболее близок к C++ и Java. Язык имеет статическую типизацию, поддерживает полиморфизм, перегрузку операторов, делегаты, атрибуты, события, свойства, обобщённые типы и методы, итераторы, анонимные функции с поддержкой замыканий, LINQ, исключения, комментарии в формате XML.

Переняв многое от своих предшественников — языков C++, Pascal, Модула, Smalltalk и, в особенности, Java — C#, опираясь на практику их использования, исключает некоторые модели, зарекомендовавшие себя как проблематичные при разработке программных систем, например, C# в отличие от C++ не поддерживает множественное наследование классов.

2. Работа программы

2.1 Взаимодействие с API

Сайт <https://openweathermap.org/> предоставляет несколько вариантов API. В своей программе я использовал только данные о погоде на текущий момент.

Чтобы получить данные, нужно отправить запрос:

```
WebReque request =  
WebRequest.Create($"http://api.openweathermap.org/data/2.5/weather?q={city}&units=metric&  
APPID=de8d63d5a267652c85f3fe2614a45ff6");
```

q - название города.

APPID - уникальный ключ пользователя, который выдается при регистрации.

units - система мер.

```
response = await request.GetResponseAsync();  
using (Stream s = response.GetResponseStream())  
{  
    using (StreamReader reader = new StreamReader(s))  
    {  
        answer = await reader.ReadToEndAsync();  
    }  
}  
response.Close();
```

В качестве ответа мы получим данные в формате Json, преобразуем их в объекты классов с помощью фреймворка Json.NET. После чего информацию можно удобно использовать, обращаясь к полям объекта:

```
OpenWeather.OpenWeather open_w =  
JsonConvert.DeserializeObject<OpenWeather.OpenWeather>(answer);
```

2.2 Структура программы

Весь код разбит на классы. В каждом файле один класс. Структура:

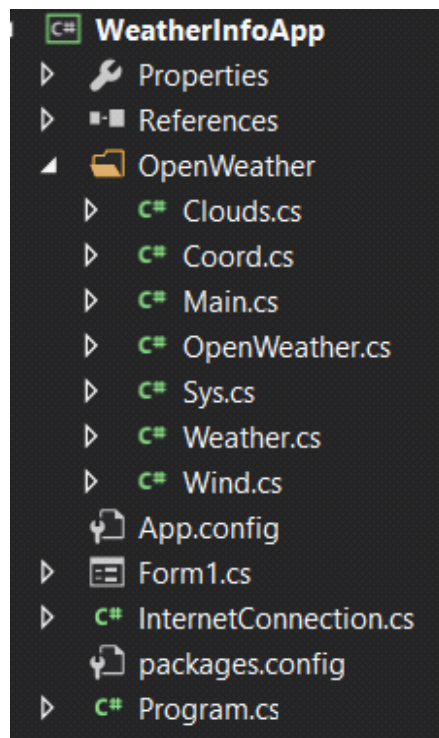


Рисунок 1. Структура программы

- *OpenWeather* — папка, в которой содержатся все классы для десериализации (преобразования Json в объекты) данных о погоде.
- *Form1.cs* — система файлов, в которой хранится пользовательский интерфейс приложения.
- *InternetConnection.cs* — файл, в котором описан класс *InternetConnection*. Проверяет наличие подключения к интернету.
- Все остальные файлы создаются вместе с проектом и никак не были мной изменены.

2.3 Описание работы программы

При запуске программы пользователя встречает стартовое окно:

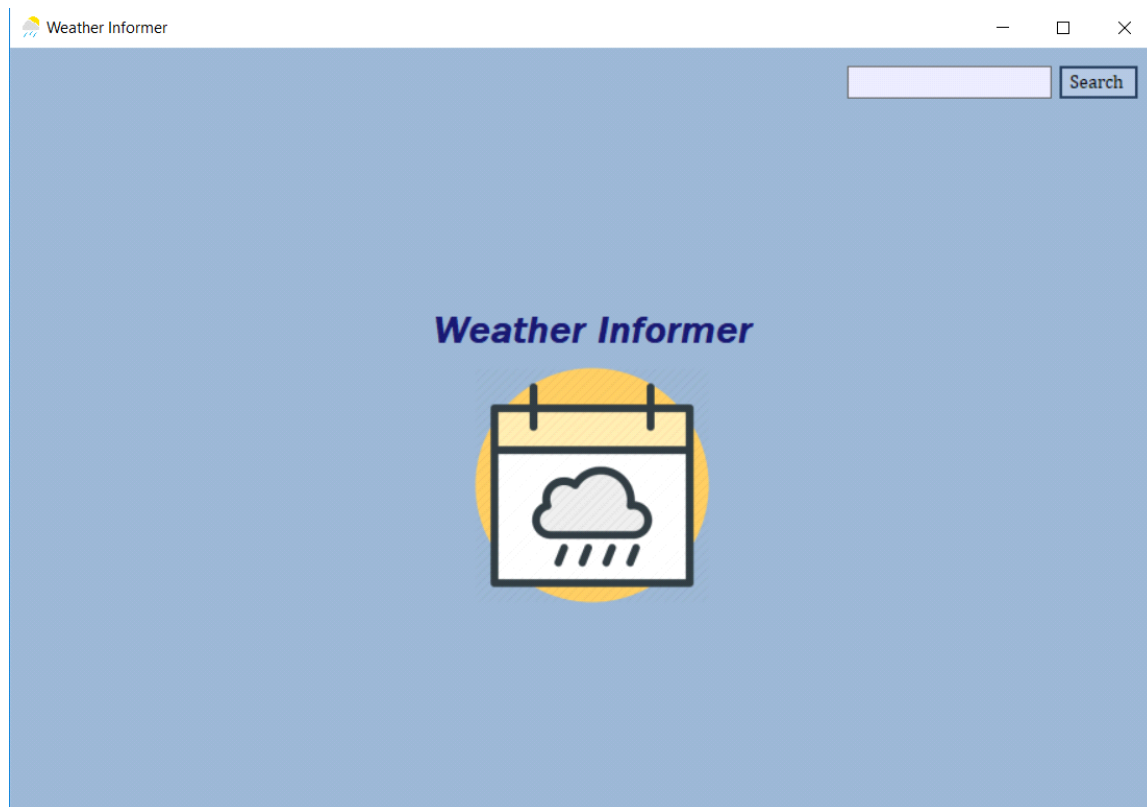


Рисунок 2. Стартовое окно

В строку для поиска вводим название города и нажимаем кнопку *Search* или клавишу *Enter*. После чего пользователь увидит основную информацию о погоде в введенном городе.

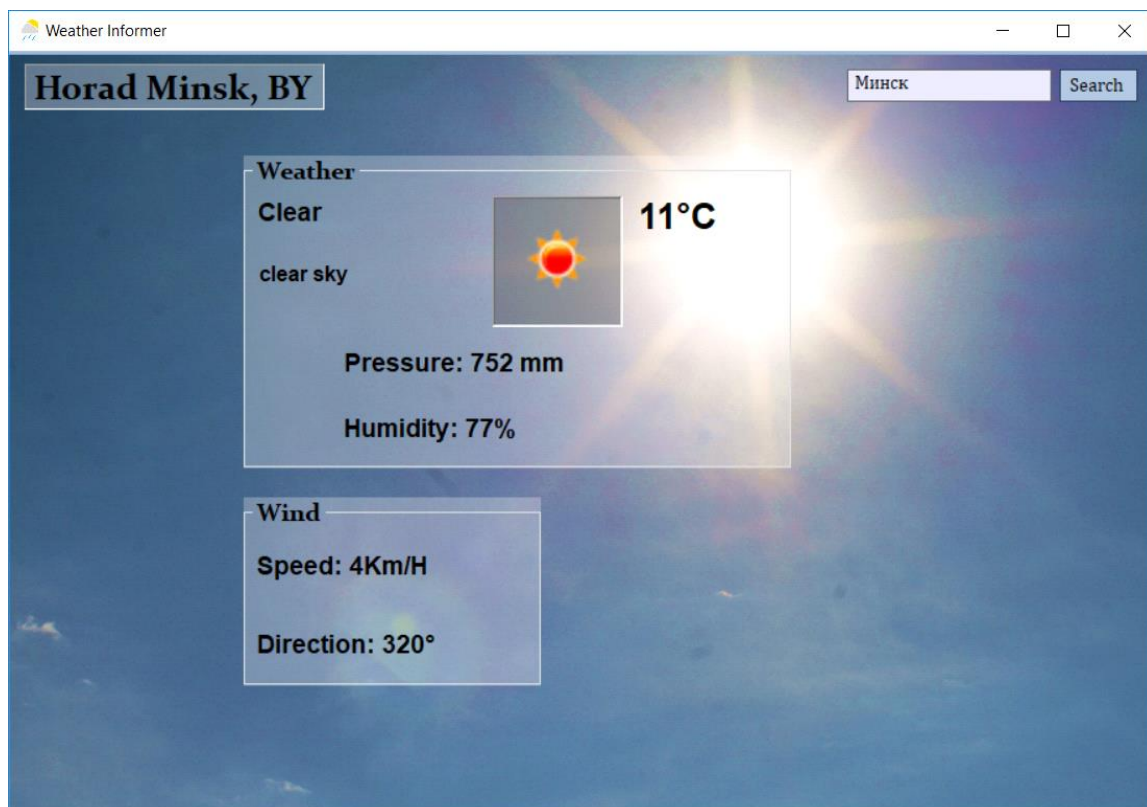


Рисунок 3. Окно с информацией о погоде

2.4 Алгоритм программы

При нажатии на кнопку поиска или клавишу *Enter* программа проверяет наличие подключения к интернету. Если подключение отсутствует, то программа выдает ошибку:

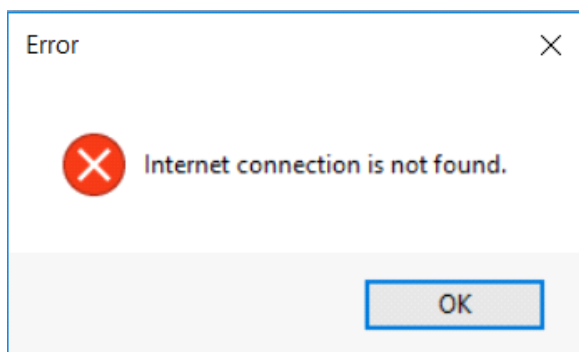


Рисунок 4. Ошибка подключения к интернету

Если же подключение присутствует, то отправляется запрос на сайт и, если название города находится в базе данных сайта, то приходит информация о погоде в этом городе в формате Json. В противном случае выдается ошибка и программа закрывается.

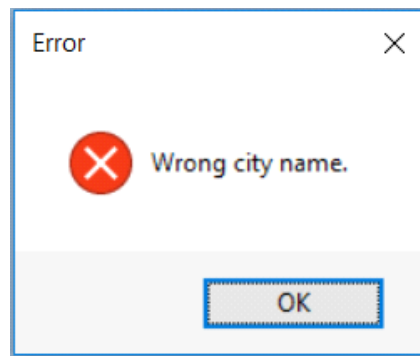


Рисунок 5. Окно ошибки поиска

После получения информации о погоде, она преобразуется в объекты класса с помощью функции:

```
OpenWeather.OpenWeather open_w =  
JsonConvert.DeserializeObject<OpenWeather.OpenWeather>(answer);
```

Затем выводятся иконки погоды (в зависимости от текущей погоды):



















Day icon	Night icon	Description
01d.png 	01n.png 	clear sky
02d.png 	02n.png 	few clouds
03d.png 	03n.png 	scattered clouds
04d.png 	04n.png 	broken clouds
09d.png 	09n.png 	shower rain
10d.png 	10n.png 	rain
11d.png 	11n.png 	thunderstorm
13d.png 	13n.png 	snow
50d.png 	50n.png 	mist

Рисунок 6. Иконки погоды

Картинки для заднего фона (в зависимости от текущей погоды):



ClearMoon



CloudMoon



CloudsDay



CloudsNight



CloudSun



Mist



Rain



SnowFall



SunShine



ThunderStorm

Рисунок 7. Картинки для заднего фона

И вся информация:

- Название города.
- Код страны.
- Сокращенное описание погоды.
- Полное описание погоды.
- Текущая температура.
- Давление.
- Влажность.
- Направление и скорость ветра.

Выводы

В данной курсовой работе была разработана программа "Погодный информер для Windows" на языке программирования C#, которая позволяет выводить всю необходимую информацию о погоде на текущий момент.

В ходе курсового проектирования была изучена работа с API на ресурсе <https://openweathermap.org/>. Рассмотрены основные характеристики используемого языка программирования C# и фреймворка Json.NET.

Программа была протестирована многими пользователями с использованием различных входных данных. Ошибок при этом выявлено не было.

В дальнейшем в программу можно добавить прогноз погоды на ближайшие дни, что не вызовет больших сложностей, так как ресурс предоставляет различные API.

Список использованных источников

1. OpenWeatherMap [электронный ресурс] - Режим доступа: <https://openweathermap.org/>.
2. Wikipedia [электронный ресурс] - Режим доступа: https://ru.wikipedia.org/wiki/C_Sharp.
3. Newton soft: Json.Net [электронный ресурс] - Режим доступа: <http://www.newtonsoft.com/json>.
4. Habrahabr [электронный ресурс] - Режим доступа: <https://habrahabr.ru/post/176087/>.