

Создание веб-приложения «Погода» с помощью API онлайн сервиса OpenWeatherMap

Стрельцова Марина Николаевна

Приамурский государственный университет им. Шолом-Алейхема

Студент

Аннотация

Погодные условия играют одну из важных ролей в жизни современного общества. Существуют огромное количество погодозависимых профессий: летчики, моряки, спасатели и так далее. Для таких людей важно знать текущую обстановку на улице, чтобы безопасно выполнять свои обязанности. Именно поэтому в настоящее время достаточно легко узнать не только погоду в любой точке мира, но и даже долгосрочный прогноз с помощью средств массовой информации. В данной статье описано создание веб-приложения «Погода» с помощью API онлайн сервиса OpenWeatherMap.

Ключевые слова: веб-приложение, php, погода, api, bootstrap.

Creating a web application "Weather" using the API of the online service OpenWeatherMap

Streltsova Marina Nikolaevna

Sholom-Aleichem Priamursky State University

Student

Abstract

Weather conditions play one of the important roles in the life of modern society. There are a huge number of weather-dependent professions: pilots, sailors, rescuers, and so on. It is important for such people to know the current situation on the street in order to safely carry out their duties. That is why it is now quite easy to find out not only the weather anywhere in the world but even a long-term forecast using the media. This article describes how to create a web application "Weather" using the API of the online service OpenWeatherMap.

Keywords: web application, PHP, weather, API, bootstrap.

1. Введение

1.1 Актуальность исследования

Погодные условия играют одну из важных ролей в жизни современного общества. Существуют огромное количество погодозависимых профессий: летчики, моряки, спасатели и так далее. Для таких людей важно знать текущую обстановку на улице, чтобы безопасно выполнять свои обязанности. Именно поэтому в настоящее время достаточно легко узнать не только погоду в любой точке мира, но и даже

долгосрочный прогноз с помощью средств массовой информации. В данной статье будет описан процесс создания веб-приложения «Погода» с помощью API онлайн сервиса OpenWeatherMap.

1.2 Обзор исследований

В своем исследовании И. Д. Премыслов описал web-приложение, собирающее и отображающее погодные показатели городов мира, и использующее JavaScript-библиотеку ReactJS [1]. Е. С. Татаринцев, К. С. Кондауров, А. И. Сукачев и Е. А. Сукачева предложили программно-аппаратный комплекс мониторинга и анализа фермерского хозяйства с возможностью аутентификации, добавления новых полей и области мониторинга погоды [2]. В своей статье В. И. Шевченко и Т. А. Бродовский разработали мобильное приложение для обработки и визуализации метеорологических данных [3]. Разработка системы мониторинга погоды в реальном времени на основе мобильного приложения с использованием автоматической метеостанции описывается в научном исследовании А. Munandar и др. [4]. В статье X. Sun и его соавторов описана разработка и реализация веб-визуальной, совместимой и масштабируемой платформы, которая способна получать доступ к климатологическим полям с моделей, спутников и наземных станций из ряда источников данных, используя Google Планета Земля в качестве общего графического интерфейса [5]. В научной статье К. Purwandari и др. рассматривается автоматическая классификация погоды на основе Twitter с использованием интеллектуального анализа текста с использованием машины опорных векторов (SVM), многомерного Байеса (MNB) и метода логистической регрессии (LR) [6].

1.3 Цель исследования

Целью данного исследования является разработка веб-приложения «Погода» с помощью API онлайн сервиса OpenWeatherMap.

1.4 Описание приложения «Погода»

Приложение отвечает за показ информации о текущей погоде в городе, введенном пользователем. В приложении также предусмотрена обработка ошибок при неверном или пустом вводе в поле «Название города». Температура в градусах Цельсия, погодные условия, атмосферное давление, скорость ветра, облачность, время восхода солнца, текущая дата и время – информация, выводимая приложением при успешном поиске города по названию, введенном пользователем.

2. Методы исследования

Для реализации веб-приложения были выбраны следующие программные средства, языки, сервисы и технологии:

- PHP - скриптовый язык общего назначения, интенсивно применяемый для разработки веб-приложений. В настоящее время поддерживается подавляющим большинством хостинг-

провайдеров и является одним из лидеров среди языков, применяющихся для создания динамических веб-сайтов [7].

- CSS - формальный язык описания внешнего вида документа (веб-страницы), написанного с использованием языка разметки (чаще всего HTML или XHTML). Также может применяться к любым XML-документам, например, к SVG или XUL [8].
- HTML - стандартизированный язык разметки документов для просмотра веб-страниц в браузере. Веб-браузеры получают HTML документ от сервера по протоколам HTTP/HTTPS или открывают с локального диска, далее интерпретируют код в интерфейс, который будет отображаться на экране монитора [9].
- PhpStorm - интегрированная среда разработки на PHP с интеллектуальным редактором, которая глубоко понимает код, поддерживает PHP 5.3-8.1 для современных и классических проектов, обеспечивает лучшее в индустрии автодополнение кода, рефакторинги, предотвращение ошибок налету и поддерживает смешивание языков [10].
- OpenWeatherMap - онлайн сервис, который предоставляет платный (есть функционально ограниченная бесплатная версия) API для доступа к данным о текущей погоде, прогнозам и историческим данным. В качестве источника данных используются официальные метеорологические службы, данные из метеостанций аэропортов, и данные с частных метеостанций [11].
- Bootstrap - свободный набор инструментов для создания сайтов и веб-приложений. Включает в себя HTML- и CSS-шаблоны оформления для типографики, веб-форм, кнопок, меток, блоков навигации и прочих компонентов веб-интерфейса, включая JavaScript-расширения [12].
- API - описание способов (набор классов, процедур, функций, структур или констант), которыми одна компьютерная программа может взаимодействовать с другой программой. Часто реализуется отдельной программной библиотекой или сервисом операционной системы. Используется программистами при написании всевозможных приложений [13].
- Open Server Panel - портативный локальный WAMP/WNMP сервер, имеющий многофункциональную управляющую программу и большой выбор подключаемых компонентов [14].

3. Результаты исследования

Для начала следует открыть PhpStorm, создать проект и файл weather.php внутри него. Далее перейти на сайт фреймворка Bootstrap и скопировать стандартную структуру html, которая станет визуальной основой (Рис. 1).

Starter template

Be sure to have your pages set up with the latest design and development standards. That means using an HTML5 doctype and including a viewport meta tag for proper responsive behaviors. Put it all together and your pages should look like this:

```
<!doctype html>
<html lang="en">
  <head>
    <!-- Required meta tags -->
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">

    <!-- Bootstrap CSS -->
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="styl

  <title>Hello, world!</title>
</head>
<body>
  <h1>Hello, world!</h1>

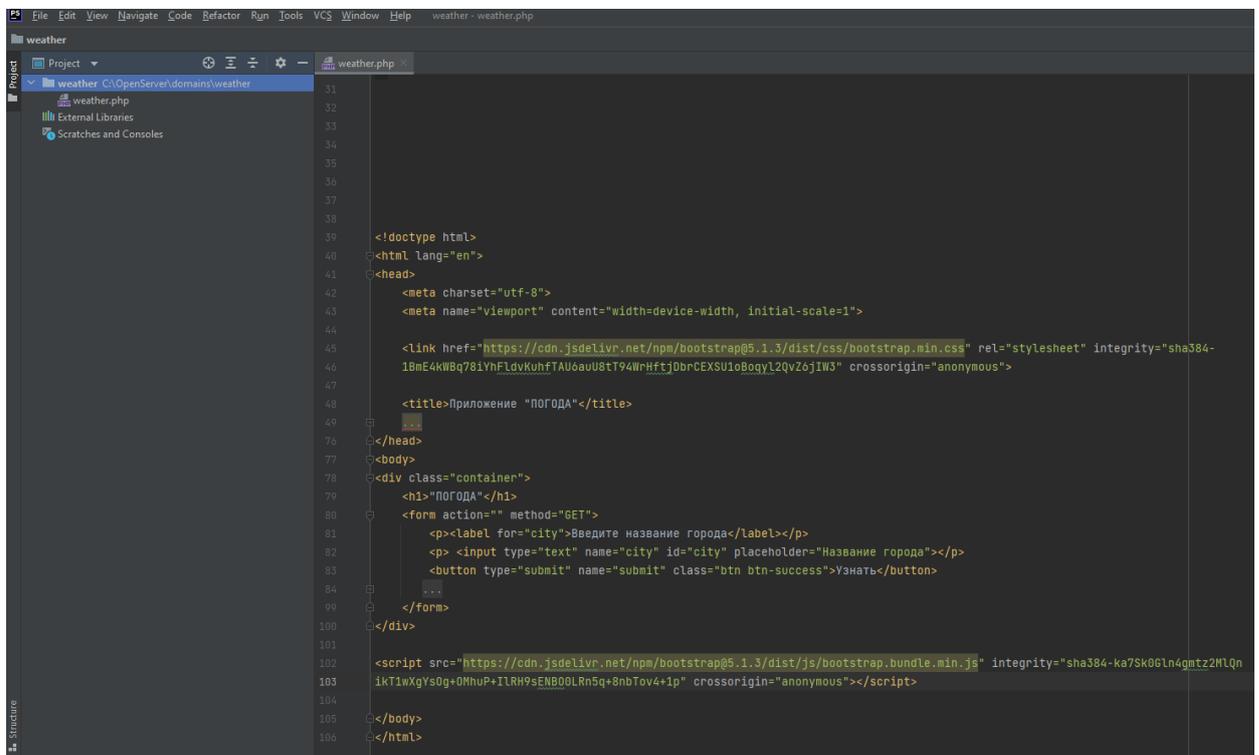
  <!-- Optional JavaScript; choose one of the two! -->

  <!-- Option 1: Bootstrap Bundle with Popper -->
  <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js" int

  <!-- Option 2: Separate Popper and Bootstrap JS -->
  <!--
  <script src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.10.2/dist/umd/popper.min.js" integr
  <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.min.js" integrity=
  -->
</body>
</html>
```

Рисунок 1 – Html структура для проекта

Далее следует вставить скопированную структуру и добавить заголовок с названием, поле для ввода названия города и кнопку отправки формы (Рис. 2).



```
31
32
33
34
35
36
37
38
39 <!doctype html>
40 <html lang="en">
41 <head>
42 <meta charset="utf-8">
43 <meta name="viewport" content="width=device-width, initial-scale=1">
44
45 <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-
46 1BmE4kWBq781YhFtLdVhF04q1U46au0tT94WVhfgtDbrCEXSU1oBoqyl2QvZ6jIW3" crossorigin="anonymous">
47
48 <title>Приложение "ПОГОДА"</title>
49
50 </head>
51 <body>
52 <div class="container">
53 <h1>ПОГОДА</h1>
54 <form action="" method="GET">
55 <p><label for="city">Введите название города</label></p>
56 <p><input type="text" name="city" id="city" placeholder="Название города"></p>
57 <p><button type="submit" name="submit" class="btn btn-success">Узнать</button>
58 </form>
59 </div>
60
61 <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js" integrity="sha384-ka7Sk0Gln4gmtz2MlQn
62 ikT1wYs0g+OMhUP+ILrH9sENBDOLRn5q+8nbTov4+1p" crossorigin="anonymous"></script>
63
64 </body>
65 </html>
```

Рисунок 2 – Добавление заголовка, поля для ввода и кнопки

Теперь необходимо прописать стили блоков, отрегулировав расположение, цвет, фон, шрифт и так далее (Рис. 3).

```
<style>
  body{
    margin: 0px;
    padding: 0px;
    box-sizing: border-box;
    background-image: url(weather.jpg);
    color: white;
    font-family: poppin, 'Times New Roman', Times, Serif;
    font-size: large;
    background-size: cover;
    background-attachment: fixed;
  }
  .container{
    text-align: center;
    justify-content: center;
    align-items: center;
    width: 440px;
  }
  h1{
    font-weight: 700;
    margin-top: 150px;
  }
  input{
    width: 350px;
    padding: 5px;
  }
</style>
```

Рисунок 3 – Свойства блоков приложения

Так будет выглядеть приложение в браузере (Рис. 4).

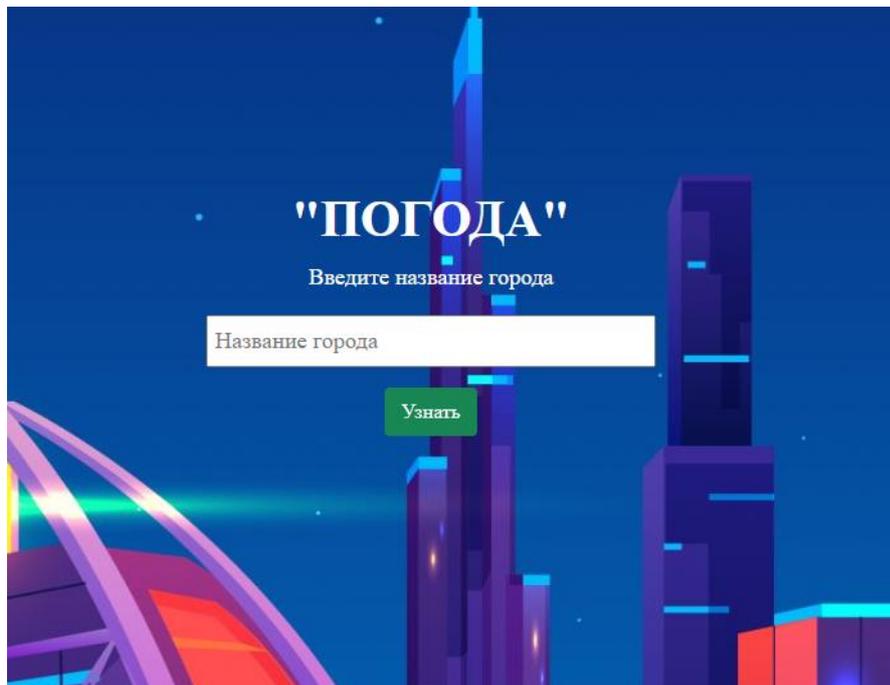


Рисунок 4 – Внешний вид приложения

Далее следует перейти на сайт сервиса OpenWeatherMap, выбрать раздел API, нажать на бесплатную подписку и зарегистрироваться для получения API ключа (Рис. 5-6).

Pricing Home / Pricing

Get weather data for any location on the globe immediately with our superb APIs! Just [sign up](#) with your email and start using minute forecasts, hourly forecasts, history and other weather data in your applications for free. For more functionality, please consider our generous subscriptions.

Current weather and forecasts collection

Free	Startup	Developer	Professional	Enterprise
Get API key	Subscribe	Subscribe	Subscribe	Subscribe
60 calls/minute 1,000,000 calls/month	600 calls/minute 10,000,000 calls/month	3,000 calls/minute 100,000,000 calls/month	30,000 calls/minute 1,000,000,000 calls/month	200,000 calls/minute 5,000,000,000 calls/month
Current Weather Minute Forecast 1 hour* Hourly Forecast 2 days* Daily Forecast 7 days*	Current Weather Minute Forecast 1 hour** Hourly Forecast 2 days** Daily Forecast 16 days	Current Weather Minute Forecast 1 hour Hourly Forecast 4 days Daily Forecast 16 days	Current Weather Minute Forecast 1 hour Hourly Forecast 4 days Daily Forecast 16 days	Current Weather Minute forecast 1 hour Hourly Forecast 4 days Daily Forecast 16 days

Рисунок 5 – Выбор бесплатной подписки

Create New Account

We will use information you provided for management and administration purposes, and for keeping you informed by mail, telephone, email and SMS of other products and services from us and our partners. You can proactively manage your preferences or opt-out of communications with us at any time using Privacy Centre. You have the right to access your data held by us or to request your data to be deleted. For full details please see the OpenWeather [Privacy Policy](#).

I am 16 years old and over

I agree with [Privacy Policy](#), [Terms and conditions of sale](#) and [Websites terms and conditions of use](#)

Рисунок 6 – Регистрация

Затем необходимо перейти в раздел API keys и скопировать полученный ключ (Рис. 7).

New Products Services **API keys** Billing plans Payments Block logs My orders My profile Ask a question

You can generate as many API keys as needed for your subscription. We accumulate the total load from all of them.

Key	Name	Status	Actions	Create key
fd6b0ea073f26f69cdb07198312fd7a2	1	Active	 	<input type="text" value="API key name"/> <input type="button" value="Generate"/>

Рисунок 7 – API ключ

Теперь, перейдя в раздел документации, нужно скопировать ссылку для получения информации об одном городе (Рис. 8).

Вызов текущих данных о погоде для одного местоположения

По названию города

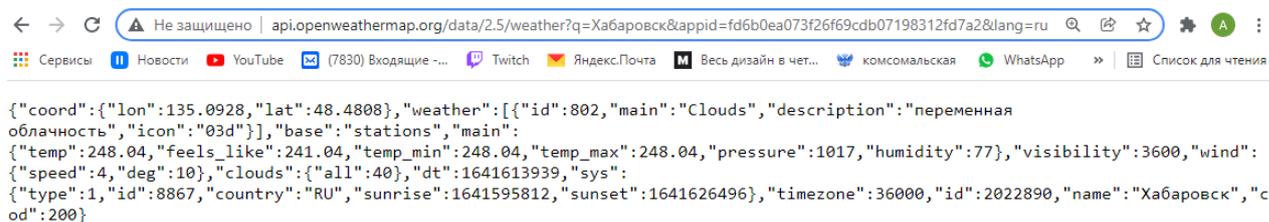
Вы можете звонить по названию города или городу, коду штата и коду страны. Обратите внимание, что поиск по штатам доступен только для локаций в США.

Вызов API

```
api.openweathermap.org/data/2.5/weather?q={city name}&appid={API key}
```

Рисунок 8 – Ссылка для получения информации об одном городе

Вставим ссылку в браузер, подставив название города, API ключ и дописав параметр lang для вывода информации на русском языке (Рис. 9).



The screenshot shows a browser window with the address bar containing the URL: `api.openweathermap.org/data/2.5/weather?q=Хабаровск&appid=fd6b0ea073f26f69cdb07198312fd7a2&lang=ru`. Below the address bar, there is a list of services and a search bar. The main content area displays the JSON response from the API, including coordinates, weather conditions, and city information.

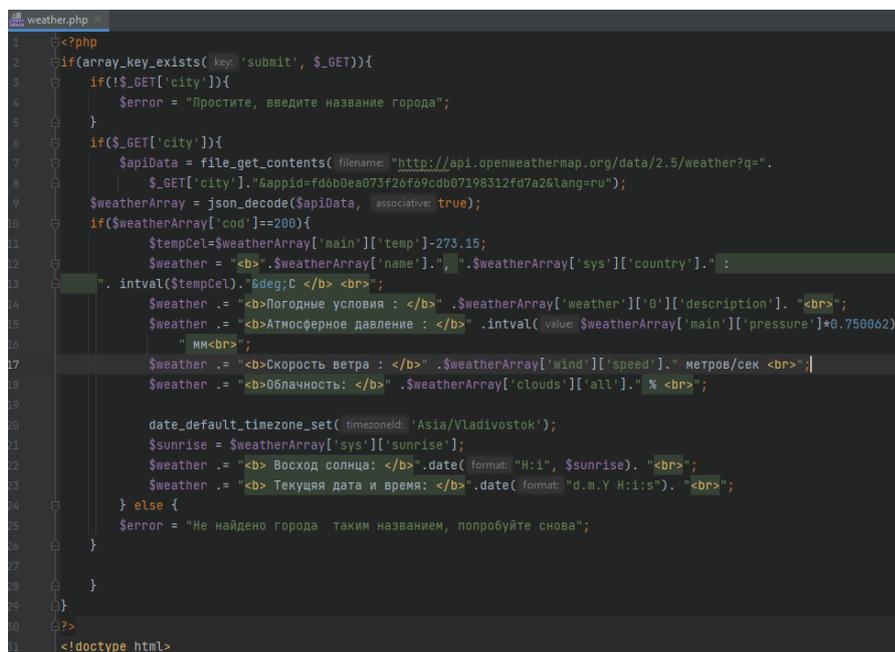
```
{ "coord": { "lon": 135.0928, "lat": 48.4808 }, "weather": [ { "id": 802, "main": "Clouds", "description": "переменная облачность", "icon": "03d" } ], "base": "stations", "main": { "temp": 248.04, "feels_like": 241.04, "temp_min": 248.04, "temp_max": 248.04, "pressure": 1017, "humidity": 77, "visibility": 3600, "wind": { "speed": 4, "deg": 10 }, "clouds": { "all": 40 }, "dt": 1641613939, "sys": { "type": 1, "id": 8867, "country": "RU", "sunrise": 1641595812, "sunset": 1641626496 }, "timezone": 36000, "id": 2022890, "name": "Хабаровск", "cod": 200 }
```

Рисунок 9 – Пример API запроса

Теперь, зная получаемую форму ответа от сервиса, можно написать код обработки запроса и вывода информации, а также код обработки ошибок (Рис. 10).

Если поле ввода пустое, то выводим ошибку, а иначе передаём запрос сервису OpenWeatherMap с введенным названием и API ключом. Если код ответа от сервиса равен 200 (успешный), то извлекаем необходимую информацию о погоде в данном городе: температура в градусах Цельсия,

погодные условия, атмосферное давление, скорость ветра, облачность, время восхода солнца, текущая дата и время и помещаем в переменную \$weather. В ином случае выводим ошибку о том, что город с таким названием не найден. Следует пояснить, что градусы сервисом передаются в Кельвинах, поэтому для удобства они были переведены в Цельсии. Точно также и с атмосферным давлением, которое было переведено из Паскалей в миллиметры ртутного столба.



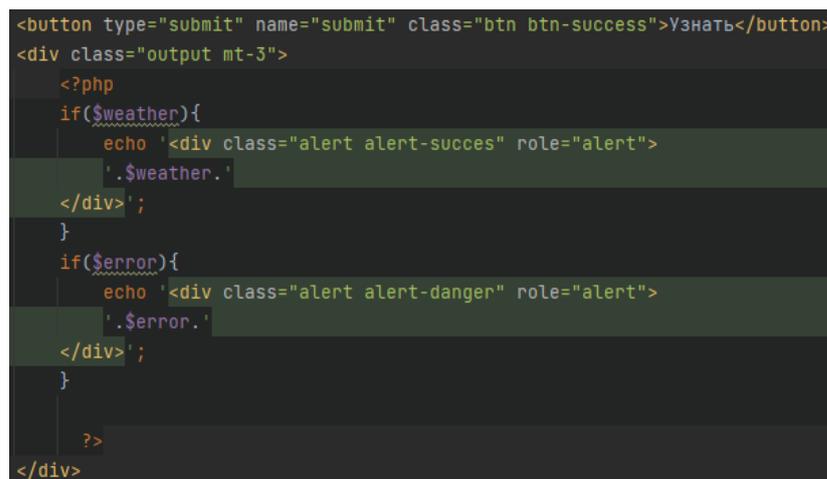
```

1 <?php
2 if(array_key_exists( key: 'submit', $_GET)){
3     if(!$_GET['city']){
4         $error = "Простите, введите название города";
5     }
6     if($_GET['city']){
7         $apiData = file_get_contents( filename: "http://api.openweathermap.org/data/2.5/weather?q=" .
8             $_GET['city'] . "&appid=f40b0ea073f26f09cd07198312fd7a2&lang=ru");
9         $weatherArray = json_decode($apiData, associative: true);
10        if($weatherArray['cod']==200){
11            $tempCel=$weatherArray['main']['temp']-273.15;
12            $weather = "<b>". $weatherArray['name']. ", ". $weatherArray['sys']['country']. " :
13            ". intval($tempCel). "°deg;C </b> <br>";
14            $weather .= "<b>Погодные условия : </b>" . $weatherArray['weather'][0]['description']. " <br>";
15            $weather .= "<b>Атмосферное давление : </b>" . intval( value: $weatherArray['main']['pressure']*0.750062).
16            " мм<br>";
17            $weather .= "<b>Скорость ветра : </b>" . $weatherArray['wind']['speed']. " м/сек <br>";
18            $weather .= "<b>Облачность : </b>" . $weatherArray['clouds']['all']. " % <br>";
19
20            date_default_timezone_set( timezoneId: 'Asia/Vladivostok');
21            $sunrise = $weatherArray['sys']['sunrise'];
22            $weather .= "<b> Восход солнца: </b>" . date( format: "H:i", $sunrise). " <br>";
23            $weather .= "<b> Текущая дата и время: </b>" . date( format: "d.m.Y H:i:s"). " <br>";
24        } else {
25            $error = "Не найдено города таким названием, попробуйте снова";
26        }
27    }
28 }
29
30 <?>
31 <!doctype html>

```

Рисунок 10 – Php код обработки ошибок и ответа от сервиса

Последним шагом будет вставка переменных \$weather и \$errors в html структуру (Рис. 11).



```

<button type="submit" name="submit" class="btn btn-success">Узнать</button>
<div class="output mt-3">
    <?php
    if($weather){
        echo '<div class="alert alert-success" role="alert">
        ' . $weather . '
        </div>';
    }
    if($error){
        echo '<div class="alert alert-danger" role="alert">
        ' . $error . '
        </div>';
    }
    ?>
</div>

```

Рисунок 11 – Вставка переменных \$weather и \$errors

Далее необходимо произвести проверку приложения на неверный ввод и правильность отображения сообщений об ошибках (Рис. 12-13).

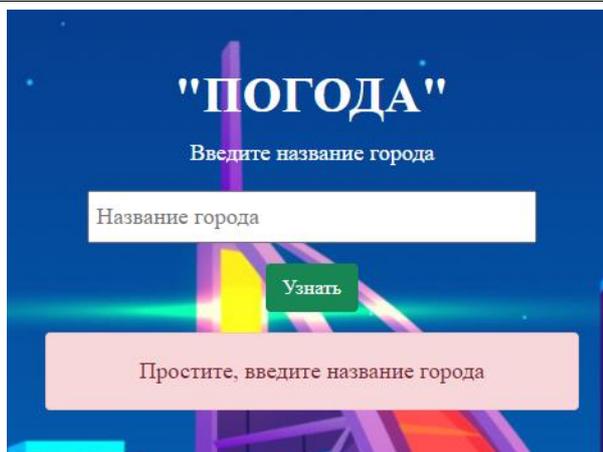


Рисунок 12 – Ошибка о пустом поле

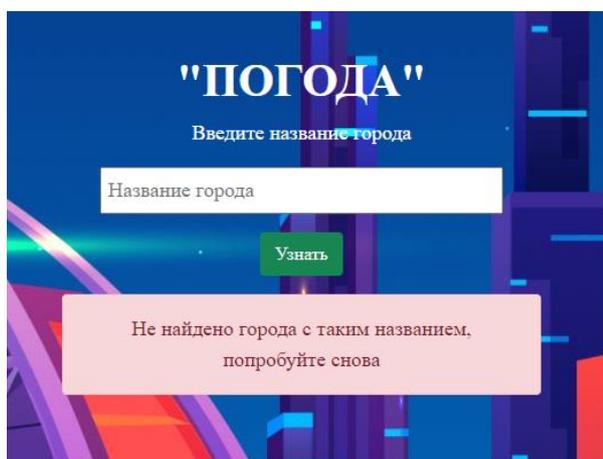


Рисунок 13 – Ошибка о неверном названии города

Теперь введем название города и посмотрим на результат работы приложения (Рис. 14-15).

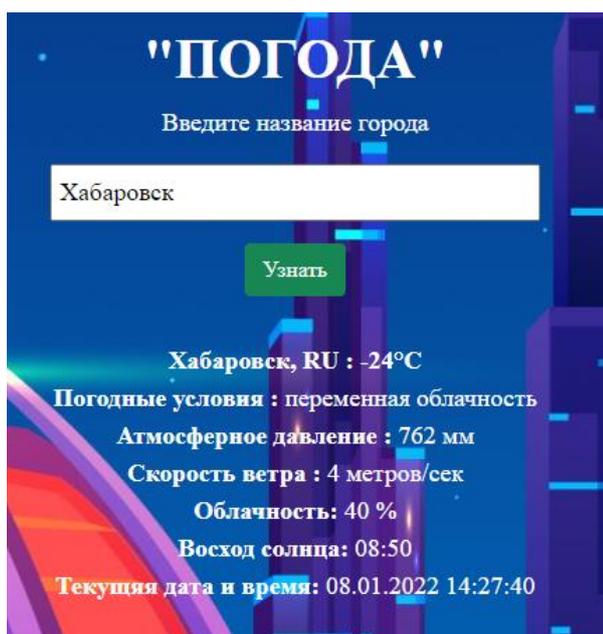


Рисунок 14 – Погода в Хабаровске

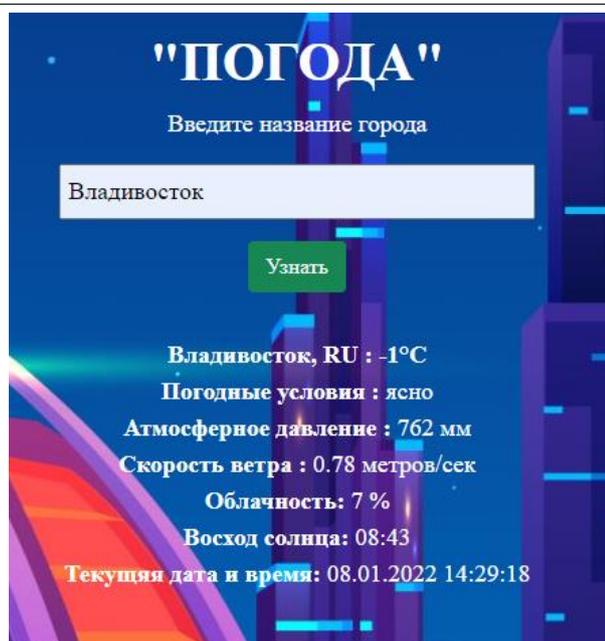


Рисунок 15 – Погода во Владивостоке

Выводы

Погодные условия всегда будут интересовать людей, так как от них напрямую зависит жизнедеятельность людей. Ведь даже простой поход на улицу планируется с учетом того какая погодная ситуация за окном. Именно поэтому всегда будут актуальными различные сервисы, сайты и приложения, помогающие быстро узнать текущую информацию о погоде в различных областях мира. В рамках данной статьи было разработано приложение «Погода» для получения информации о погодных условиях в конкретном городе на текущий момент. Также была рассмотрена технология API для взаимодействия созданного приложения с онлайн-сервисом OpenWeatherMap и получения актуальных данных о погоде.

Библиографический список

1. Премыслов И. Д. Особенности реализации web-приложения по сбору погодных данных городов мира // Конкурентоспособность территорий. 2019. С. 163-165.
2. Татаринцев Е.С., Кондауров К. С., Сукачев А. И., Сукачева Е. А. Разработка мобильной системы мониторинга погодных условий для фермерского хозяйства // Научная опора Воронежской области. 2020. С. 255-256.
3. Шевченко В. И., Бродовский Т. А. Мобильное приложение для обработки и представления метеорологических данных // Экологическая, промышленная и энергетическая безопасность-2019. 2019. С. 1784-1788.
4. Munandar A. et al. Design of real-time weather monitoring system based on mobile application using automatic weather station // 2017 2nd International Conference on Automation, Cognitive Science, Optics, Micro Electro-Mechanical System, and Information Technology (ICACOMIT). IEEE, 2017.

- С. 44-47.
5. Sun X. et al. Development of a Web-based visualization platform for climate research using Google Earth // Computers & Geosciences. 2012. Т. 47. С. 160-168.
 6. Purwandari K. et al. Multi-class Weather Forecasting from Twitter Using Machine Learning Approaches // Procedia Computer Science. 2021. Т. 179. С. 47-54.
 7. PHP URL: <https://www.php.net/> (дата обращения: 07.01.2022).
 8. CSS URL <https://ru.wikipedia.org/wiki/CSS> (дата обращения: 07.01.2022).
 9. HTML URL: <https://ru.wikipedia.org/wiki/HTML> (дата обращения: 07.01.2022).
 - 10.PHPStorm URL: <https://www.jetbrains.com/ru-ru/phpstorm/> (дата обращения: 07.01.2022).
 - 11.OpenWeatherMap URL: <https://openweathermap.org/> (дата обращения: 07.01.2022).
 - 12.Bootstrap URL: <https://getbootstrap.com/> (дата обращения: 07.01.2022).
 - 13.API URL: <https://ru.wikipedia.org/wiki/API> (дата обращения: 07.01.2022).
 - 14.OpenServer Panel URL: <https://ospanel.io/> (дата обращения: 07.01.2022).