|  |  |
| --- | --- |
| **O:\IMG_6334 (1).PNG** | Автономное учреждение профессионального образования  Ханты-Мансийского автономного округа – Югры  «Нефтеюганский политехнический колледж» |

**Курсовая работа (проект)**

Создание приложения «Мониторинг погоды»

09.02.07 Информационные системы и программирование

Выполнил(а) работу

обучающийся 3 курса \_\_\_\_\_\_\_\_\_\_\_\_ Кузьмин А.С.

(подпись)

Руководитель \_\_\_\_\_\_\_\_\_\_\_\_ Козырева В. В.

(подпись)

Нефтеюганск 2023 г

# Содержание

[Содержание 1](#_Toc1283520077)

[Введение 2](#_Toc1862238875)

[1. ТЕОРЕТИЧЕСКАЯ ЧАСТЬ 4](#_Toc815146162)

[1.1 Описание предметной области 4](#_Toc502780623)

[1.2 Обзор аналогов 5](#_Toc504375290)

[1.3. Обоснование необходимости разработки 6](#_Toc1918470394)

[1.4. Постановка задачи 7](#_Toc395001398)

[1.5. Используемое программное обеспечение 8](#_Toc2054980730)

[2. ПРАКТИЧЕСКАЯ ЧАСТЬ 9](#_Toc921369416)

[2.1. Выбор средства автоматизации 10](#_Toc1230008895)

[2.2. Требования к программе 12](#_Toc1277335022)

[2.3. Проектирование программного продукта 12](#_Toc1699714625)

[2.4. Текст программы с описанием 13](#_Toc1758504014)

[2.5. Тестирование и отладка 23](#_Toc1989144676)

# Введение

С давних времен люди стремились предугадывать погоду, пытались установить связь между ее изменениями и движением небесных тел. Но все эти попытки не имели научной основы и поэтому были безрезультатны.

В XX и XXI веке, ситуация кардинально изменилась. Для климатического мониторинга сегодня привлекаются огромные вычислительные мощности, сложнейшие приборы и оборудование. Сейчас считываются самые мелкие изменения в параметрах, которые невозможно заметить невооруженным глазом.

Климатический мониторинг погоды - это также международное занятие. Данные, собранные искусственными спутниками Земли, а также тысячами станций метеонаблюдения, поступают в международные дата-центры, где обрабатываются и анализируются.

В этой сфере есть много приложений на разных языках, которые позволяют чувствовать себя уверенно в любых погодных условиях. Независимо от того, находитесь ли вы на отдыхе, планируете ли выезд на природу или просто хотите быть в курсе местной погоды, приложения для мониторинга погоды сделают вашу жизнь гораздо проще и удобнее.

**Актуальность** данного проекта подтверждается тем, погодные приложения помогают нам принимать более обоснованные решения на работе, дома или в путешествиях, и снижать возможные риски для здоровья и безопасности. Например, приложения для мониторинга погоды могут предупреждать о неблагоприятных погодных условиях, таких как сильные дожди, снегопады или грозы, предоставлять информацию об уровне загрязнения воздуха, показывать температуру и влажность воздуха, а также обеспечивать прогнозы на долгосрочный период.

**Объект исследования:** Приложение для мониторинга прогноза погоды.

**Предмет исследования:** Разработка приложения с помощью HTML, CSS, JS.

**Цель:** Разработать приложение для мониторинга погоды, используя текстовый редактор Visual Studio Code с использование языка программирования JavaScript.

**Задачи:**

1. изучить соответствующие аналоги;
2. создать код приложения и интерфейсных компонентов на основе выбранных алгоритмов;
3. разработать интерфейс приложения;
4. протестировать программу с целью выявления ошибок.

**Методы:** поиск информации, анализ информации, синтез.

**Практическая значимость работы:** состоит в том, чтобы люди использовали программу в повседневной жизни, записывая те дела, которые они хотят сделать, мотивируя себя и не забывая, что им нужно сделать.

**Структура курсовой работы:** курсовая состоит из введения, теоретической части, практической части, заключения и списка использованной литературы.

# 1. ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

# 1.1 Описание предметной области

Приложение для мониторинга погоды – это программное обеспечение, которые позволяет пользователю получать информацию о текущей и прогнозируемой погоде в определенном регионе или на определенном месте. Основной целью этих приложений является предоставление точной и своевременной информации о погоде, которая может быть использована для принятия решений, таких как выбор одежды, планирование поездки и т.д.

В приложении для мониторинга погоды обычно доступна информация о температуре воздуха, давлении, влажности, скорости ветра, направлении ветра, осадках и других факторах погоды. Эти данные могут быть представлены в виде диаграмм, графиков или карт.

Приложения для мониторинга погоды основываются на использовании данных, собранных с локальных метеостанций, спутников и других источников. Некоторые приложения могут использовать искусственный интеллект для анализа этих данных и предоставления более точных прогнозов.

Пользователи могут настроить свои приложения для получения оповещений о критических изменениях погоды, таких как штормы или ураганы. Они также могут выбрать из различных единиц измерения, таких как Фаренгейт или Цельсий для температуры, или мили или километры для скорости ветра.

Эти приложения могут быть полезны как для повседневного использования, так и для профессиональной деятельности, связанной с погодой, таких как фермерство, авиационная и морская навигация.

Предметная область мониторинга прогноза погоды включает в себя сбор, обработку и анализ метеорологических данных с целью предоставления точной и надежной информации о погодных условиях в различных регионах мира.

# 1.2 Обзор аналогов

В современном мире существует множество приложений для мониторинга прогноза погоды которые предназначены для широкой аудитории пользователей. Среди наиболее популярных приложений для мониторинга прогноза погоды можно выделить следующие:

1. Яндекс.Погода (приложение 1) — это бесплатное приложение для пользователей в России, которые предоставляет прогноз погоды на ближайшие 10 дней, а также информацию о температуре, атмосферном давлении, скорости ветра и вероятности выпадения осадков. В основе сервиса — искусственный интеллект Meteum, который обрабатывает показания тысяч приборов на земле и в космосе.

Яндекс.Погода появилась в далёком 2000 году, это был один из первых сервисов компании. В начале пути он представлял собой погодный виджет от внешнего сервиса «Метео-ТВ». Позже в команде появились выделенные менеджер и разработчик, которые наладили показ на сайте «Яндекса» метеопрогноза финской компании Foreca для ограниченного списка городов.

1. AccuWeather (приложение 1) — это одно из наиболее популярных приложений для отслеживания погоды, которое может отображать еженедельные, ежедневные и ежечасные метеосводки. Каждый день более полутора миллиардов человек по всему миру полагаются на ресурс AccuWeather, который помогает им планировать свою жизнь, защищать бизнес и эффективно использовать время. AccuWeather ежеминутно с помощью приложения Superior Accuracy™ предоставляет точные и персонализированные прогнозы погоды, в том числе и видео прогнозы, доступные на ТВ, смартфонах, планшетах и других устройствах, имеющих доступ к интернету. Президент, основавший компанию в 1962 году, д-р Джоэл Майерс (Dr. Joel N. Myers) является членом американского метеорологического общества и признан одним из лучших предпринимателей в американской истории. AccuWeather предоставляет точные прогнозы погоды средствам массовой информации, корпоративным заказчикам, государственным и образовательным учреждениям, а также новости и видео более чем для 180 000 сторонних сайтов.
2. Gismeteo (приложение 1) — это метеорологический ресурс, которым ежедневно пользуются миллионы людей по всему миру. При помощи сайта, мобильных приложений и виджетов можно за секунду узнать текущую погоду, а также получить прогноз температуры, осадков и других необходимых параметров в различных пунктах по всей Земле. Также, используя профессиональные программные комплексы Гисметео, включающие в себя компоненты для авиации, морских служб, энергетики, сельского хозяйства, экологии и других отраслей, зависящих от погоды, можно провести исчерпывающий метеорологический анализ.

# 1.3. Обоснование необходимости разработки

Существует множество причин, почему приложения для мониторинга погоды являются необходимыми. Ниже перечислены некоторые из них:

1. Безопасность: Предупреждения о неблагоприятных погодных условиях, таких как ураганы, торнадо и сильные дожди, могут быть решающими при принятии решения о том, когда и как безопасно покинуть дом или принять меры предосторожности.
2. Планирование: Приложения для мониторинга погоды помогут вам планировать свои дни и недели в соответствии с прогнозами погоды. Например, вы можете избежать прогулок на открытом воздухе во время сильного ветра или дождя, или же планировать пикники или спортивные мероприятия в дни с хорошей погодой.
3. Экономия времени и денег: Поездки на работу или на дальние расстояния могут занимать дополнительное время и стать ненужными при плохих погодных условиях. С помощью приложения для мониторинга погоды можно избежать лишних поездок и траты денег на транспорт.
4. Помощь в уходе за здоровьем: Для людей, страдающих от заболеваний, связанных с дыхательной системой, приложение для мониторинга погоды может помочь предотвратить возникновение проблем и подготовиться к неблагоприятной погоде.
5. Удобство: Быстрый и удобный доступ к информации о погоде, в любое время и в любом месте, делает приложения для мониторинга погоды необходимыми для многих пользователей.

В целом, приложения для мониторинга погоды являются важным инструментом, который помогает людям принимать умные и информированные решения о своей жизни в зависимости от погодных условий.

Разработка своего приложения для мониторинга погоды может быть связана с желанием создать что-то уникальное, удобное и полезное для пользователей.

# 1.4. Постановка задачи

Цель данной курсовой работы состоит в создании программы для мониторинга погоды, которая будет предоставлять актуальную информацию о погоде в реальном времени, а также прогнозы на ближайшие дни. Программа будет разработана на языке программирования JavaScript с использованием открытых API погодных сервисов.

Задачи, которые необходимо решить для достижения поставленной цели:

1. Разработать архитектуру программы, определить ее функциональные возможности и интерфейс.
2. Написать код приложения на языке программирования JavaScript, включающую основные функции работы с API погодного сервиса, получение и анализ данных о погоде.
3. Протестировать программу и исправить обнаруженные ошибки.
4. Убедиться в соответствии программы требованиям задачи и ее полной функциональности.

# 1.5. Используемое программное обеспечение

Все элементы, необходимые для создания приложения, объединены в специальной программе IDE, а именно:

1. Кодовый редактор

2. Транслятор (компилятор или интерпретатор)

3. Средства автоматизированной сборки

4. Отладчик

Как правило, каждая IDE предназначена для нескольких языков программирования. Для разработки приложения на языке JavaScript можно выделить следующие среды разработки:

1. Sublime Text – легкий текстовый редактор с широким набором функциональных возможностей и поддержкой плагинов, который позволяет ускорить процесс разработки.
2. Visual Studio Code – бесплатный многофункциональный кодовый редактор с открытым исходным кодом, который поддерживает большое количество языков программирования, интегрированный отладчик и возможность запуска кода из самого редактора.
3. Atom – бесплатный редактор кода от Github, который поддерживает множество плагинов и тем, а также является настраиваемым и расширяемым.
4. Brackets – бесплатный редактор кода от Adobe, который дает возможность быстро и комфортно работать со своими проектами, благодаря функциональным возможностям и плагинам.
5. SciTe – простой, лёгкий и очень быстрый текстовый редактор, поддерживающий большую часть языков программирования и основные функциональные возможности.
6. Notepad++ – бесплатный редактор кода, который позволяет работать с большой частью языков программирования и обладает множеством функциональных возможностей.
7. Vim – текстовый редактор, который позволяет быстро и эффективно редактировать текст, благодаря своей уникальной системе управления.
8. Emacs – текстовый редактор, который поддерживает многоплатформенность и множество пакетов, а также специальных функций для упрощения работы.

# 2. ПРАКТИЧЕСКАЯ ЧАСТЬ

# 2.1. Выбор средства автоматизации

Данный проект необходимо реализовать в среде программирования Visual Studio Code с помощью языка JavaScript.

Visual Studio Code (VS Code) — текстовый редактор, разработанный Microsoft для Windows, Linux и macOS. Позиционируется как «лёгкий» редактор кода для кроссплатформенной разработки веб- и облачных приложений.

Преимущества Visual Studio Code:

1. Открытый и бесплатный исходный код: VS Code разработан и поддерживается Microsoft как бесплатный и открытый исходный код с поддержкой GitHub. Это означает, что он доступен для всех, а его исходный код может быть изменен и улучшен со всеми по мере необходимости.
2. Настраиваемый интерфейс: VS Code имеет настраиваемый интерфейс, который позволяет пользователям настраивать его в соответствии с его индивидуальными потребностями. С помощью него можно сделать очень удобный и интуитивно понятный интерфейс программирования.
3. Поддержка многих языков программирования: VS Code поддерживает множество языков программирования, что делает его очень полезным для различных программистов, включая PHP, C++, Java, Python, Ruby, SQL и многие другие.
4. Отличные инструменты для отладки: VS Code предоставляет широкий выбор отладочных инструментов, которые делают процесс отладки кода проще и быстрее.
5. Терминал внутри программы: Программа имеет командную строку, что позволяет вам работать с файловой системой и выполнить любые команды прямо внутри программы.
6. Расширяемость: VS Code расширяем и может быть дополнен различными плагинами, которые добавляют больше функциональности в процесс программирования.
7. Хорошее сообщество поддержки: Существует большое сообщество пользователей, которые активно развивают и поддерживают VS Code. Это означает, что вы можете получить помощь, если у вас возникнут проблемы, или задать вопрос, если есть необходимость.

JavaScript – это интерпретируемый язык программирования, который обычно используется для создания интерактивных веб-страниц. Он поддерживает многие функции, которые обычно присутствуют в объектно-ориентированных языках программирования, таких как наследование, инкапсуляция и полиморфизм.

JavaScript можно использовать как на стороне клиента в браузере, так и на стороне сервера благодаря Node.js. Синтаксис языка основан на C, у него есть своя стандартная библиотека, которая содержит множество полезных функций и методов для работы с различными типами данных и объектами. В JavaScript можно использовать различных фреймворки и библиотеки, такие как jQuery, React, Vue.js и другие, которые значительно упрощают работу с языком.

В качестве вспомогательных языков программирования были выбраны:

HTML (Hypertext Markup Language) является языком разметки, который используется для создания веб-страниц и придания им структурированности. Он определяет элементы веб-страницы и их расположение на странице, такие как заголовки, параграфы, ссылки, изображения и т.д. Кроме того, HTML позволяет веб-разработчику включать другие типы мультимедиа, такие как видео и аудио, в свои веб-страницы.

CSS (Cascading Style Sheets) – это язык стилей, который используется для изменения внешнего вида веб-страниц, созданных на HTML. CSS позволяет веб-разработчикам задавать цвета, шрифты, размеры и многое другое для элементов HTML. Он позволяет создавать красивые макеты и управлять внешним видом и расположением элементов веб-страницы. Одно из главных преимуществ в использовании CSS заключается в том, что он позволяет снизить объем HTML-кода, что улучшает производительность веб-страниц.

# 2.2. Требования к программе

Моя программа должна выполнять такие функции как:

* Иметь простой и интуитивно понятный интерфейс, позволяющий быстро и легко получать информацию о погоде.
* Принимать значения, введенные пользователем и после обрабатывать их.
* Принимать данные из API сайтов.
* Поддерживать несколько языков, что позволит использовать его пользователям в разных регионах мира.
* Предоставлять актуальные и точные данные о погоде на текущий момент времени, а также прогнозы на ближайшие дни.

USE-CASE схема программы (приложение 2)

# 2.3. Проектирование программного продукта

При создании приложения «Мониторинг погоды» был использован следующий прием – модульное программирование.

Модульное программирование — это организация программы как совокупности небольших независимых блоков, называемых модулями, структура и поведение которых подчиняются определённым правилам. Использование модульного программирования позволяет упростить тестирование программы и обнаружение ошибок. Аппаратно-зависимые подзадачи могут быть строго отделены от других подзадач, что улучшает мобильность создаваемых программ.

Модуль — функционально законченный фрагмент программы. Во многих языках (но далеко не обязательно) оформляется в виде отдельного файла с исходным кодом или поименованной непрерывной её части. Некоторые языки предусматривают объединение модулей в пакеты.

# 2.4. Текст программы с описанием

Программный продукт «Мониторинг погоды» выводит данные о погоде с сайта OpenWeather в соответствии с местоположением пользователя.

В своей программе я использовал:

* Язык программирования JavaScript
* Язык разметки HTML
* Язык описания внешнего вида CSS
* Текстовый редактор – Visual Studio Code

Плагины:

Live Server – позволяет вам видеть изменения кода, отраженные в браузере. Он запускает локальный сервер разработки с функцией перезагрузки в реальном времени как для статических, так и для динамических страниц. Каждый раз, когда вы сохраняете свой код, вы мгновенно видите изменения, отраженные в браузере.

[Prettier](https://prettier.io/) – Code Formatter – самоуверенный форматировщик кода. Он обеспечивает согласованный стиль, анализируя ваш код и перепечатывая его с помощью собственных правил, учитывающих максимальную длину строки, при необходимости перенося код.

(Приложение 2)

Размер - 18,7 МБ (19 650 091 байт)

На диске - 18,7 МБ (19 685 376 байт)

Код index.html:

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Прогноз погоды</title>

<link rel="stylesheet" href="./css/main.css">

</head>

<body>

<header class="header">

<h1 class="title">Прогноз погоды</h1>

<form class="form" id="form">

<input id="inputCity" class="input" type="text" placeholder="Введите название города">

<button class="btn">Показать</button>

</form>

</header>

<script src="./conditions.js" type="module"></script>

<script src="./main.js" type="module"></script>

</body>

</html>

Код main.css:

@import './reset.css';

@import url(https://fonts.googleapis.com/css?family=Rubik:300,regular,500,700);

html,

body {

min-height: 100vh;

}

body {

display: flex;

flex-direction: column;

align-items: center;

row-gap: 50px;

font-family: 'Rubik', sans-serif;

}

.header {

width: 100%;

display: flex;

flex-direction: column;

align-items: center;

padding: 40px 15px 50px;

background: linear-gradient(92.7deg, #4cb8c4 0%, #3cd3ad 100%), #232323;

}

.title {

margin-bottom: 20px;

font-family: 'Rubik';

font-weight: 700;

font-size: 44px;

line-height: 1.18;

text-align: center;

color: #ffffff;

}

.form {

display: flex;

justify-content: space-between;

width: 460px;

height: 54px;

background: #ffffff;

border-radius: 6px;

}

.input {

flex-grow: 1;

padding: 15px;

font-weight: 300;

font-size: 20px;

line-height: 1;

color: #000000;

}

.btn {

width: 145px;

height: 54px;

background: #e5464f;

border-radius: 6px;

font-weight: 500;

font-size: 20px;

line-height: 1;

color: #fff;

transition: background-color 0.2s ease-in;

}

.btn:hover {

background: #f5313b;

}

.card {

position: relative;

display: flex;

flex-direction: column;

row-gap: 20px;

padding: 30px;

min-width: 360px;

background-color: #fff;

border-radius: 6px;

box-shadow: 0px 5px 20px rgba(27, 108, 114, 0.25);

}

.card::before {

content: '';

position: absolute;

left: 10px;

bottom: -10px;

z-index: -1;

width: calc(100% - 20px);

height: 100%;

background: #ffffff;

box-shadow: 0px 5px 20px rgba(27, 108, 114, 0.25);

border-radius: 6px;

}

.card-city {

font-weight: 500;

font-size: 28px;

line-height: 33px;

color: #000000;

}

.card-city span {

display: inline-block;

vertical-align: text-top;

padding: 3px 5px;

background: #e5464f;

border-radius: 6px;

color: #fff;

font-weight: 500;

font-size: 12px;

line-height: 1;

}

.card-weather {

display: flex;

justify-content: space-between;

align-items: center;

}

.card-value {

font-size: 92px;

line-height: 1;

color: #000000;

}

.card-value sup {

font-size: 50%;

}

.card-img {

width: 120px;

height: 120px;

object-fit: cover;

}

.card-description {

font-size: 20px;

line-height: 1.2;

}

Код main.js:

import conditions from './conditions.js';

console.log(conditions);

const apiKey = 'e9a5d3b74bf84418b11193028231901';

// Элементы на странице

const header = document.querySelector('.header');

const form = document.querySelector('#form');

const input = document.querySelector('#inputCity');

function removeCard() {

const prevCard = document.querySelector('.card');

if (prevCard) prevCard.remove();

}

function showError(errorMessage) {

// Отобразить карточку с ошибкой

const html = `<div class="card">${errorMessage}</div>`;

// Отображаем карточку на странице

header.insertAdjacentHTML('afterend', html);

}

function showCard({ name, country, temp, condition, imgPath }) {

// Разметка для карточки

const html = `<div class="card">

<h2 class="card-city">${name} <span>${country}</span></h2>

<div class="card-weather">

<div class="card-value">${temp}<sup>°c</sup></div>

<img class="card-img" src="${imgPath}" alt="Weather">

</div>

<div class="card-description">${condition}</div>

</div>`;

// Отображаем карточку на странице

header.insertAdjacentHTML('afterend', html);

}

async function getWeather(city) {

const url = `http://api.weatherapi.com/v1/current.json?key=${apiKey}&q=${city}`;

const response = await fetch(url);

const data = await response.json();

console.log(data);

return data;

}

// Слушаем отправку формы

form.onsubmit = async function (e) {

// Отменяем отправку формы

e.preventDefault();

// Берем значение из инпута, обрезаем пробелы

let city = input.value.trim();

// Получаем данные с сервера

const data = await getWeather(city);

if (data.error) {

removeCard();

showError(data.error.message);

} else {

removeCard();

console.log(data.current.condition.code);

const info = conditions.find(

(obj) => obj.code === data.current.condition.code

);

console.log(info);

console.log(info.languages[23]['day\_text']);

const filePath = './img/' + (data.current.is\_day ? 'day' : 'night') + '/';

const fileName = (data.current.is\_day ? info.day : info.night) + '.png';

const imgPath = filePath + fileName;

console.log('filePath', filePath + fileName);

const weatherData = {

name: data.location.name,

country: data.location.country,

temp: data.current.temp\_c,

condition: data.current.is\_day

? info.languages[23]['day\_text']

: info.languages[23]['night\_text'],

imgPath,

};

showCard(weatherData);

}

};

Интерфейс программы (приложение 3)

2.5. Тестирование и отладка

Отладка — этап разработки компьютерной программы, на котором обнаруживают, локализуют и устраняют ошибки.

Для тестирования и отладки приложения для мониторинга погоды можно использовать различные подходы и техники.

1. Функциональное тестирование. Проверка корректности работы каждой функции приложения и достижения ожидаемого результата. Например, тестирование определения текущей погоды.
2. Тестирование производительности. Проверка скорости работы приложения и его отклика на запросы пользователя. Например, определение времени ответа приложения при загрузке данных о погоде.
3. Тестирование при различных условиях. Перед выпуском приложения важно также проверить его работы, что оно будет работать корректно на всех устройствах.
4. Тестирование стрессоустойчивости. Проверка работоспособности приложения в условиях повышенной нагрузки. Например, проверка работы приложения в период пикового спроса погодных данных.

В заключении данной курсовой работы можно подвести итоги, которые отражают актуальность данной темы. Приложение для мониторинга погоды востребовано настоящее время из-за необходимости получения точной и своевременной информации о погодных условиях. Это особенно важно для людей, чья жизнь и работа зависят от погоды. Кроме того, многие люди используют приложения, чтобы планировать свои отпуска, путешествия или просто выбирать подходящую одежду на каждый день. В целом, приложение для мониторинга погоды является важным инструментом для всех, кто хочет быть в курсе погодных условий в любое время и в любом месте.

Выводы проделанной мной работы:

1. Был проведен анализ приложений, имеющихся в общем доступе и соответствующих тематике погодных приложений.
2. Я сделал приложение, которое реализует следующие функции:

* Иметь простой и интуитивно понятный интерфейс, позволяющий быстро и легко получать информацию о погоде.
* Принимать значения, введенные пользователем и после обрабатывать их.
* Принимать данные из API сайтов.
* Поддерживать несколько языков, что позволит использовать его пользователям в разных регионах мира.
* Предоставлять актуальные и точные данные о погоде на текущий момент времени, а также прогнозы на ближайшие дни.

**Список литературы**

1. Википедия-свободная энциклопедия [Электронный ресурс] - https://ru.wikipedia.org/wiki/

## 2. Фримен, Робсон – Изучаем программирование на JavaScript, 2014 – 640 с.

3. Дронов Владимир JavaScript. Народные советы; БХВ-Петербург -

М., 2016 – 458с.

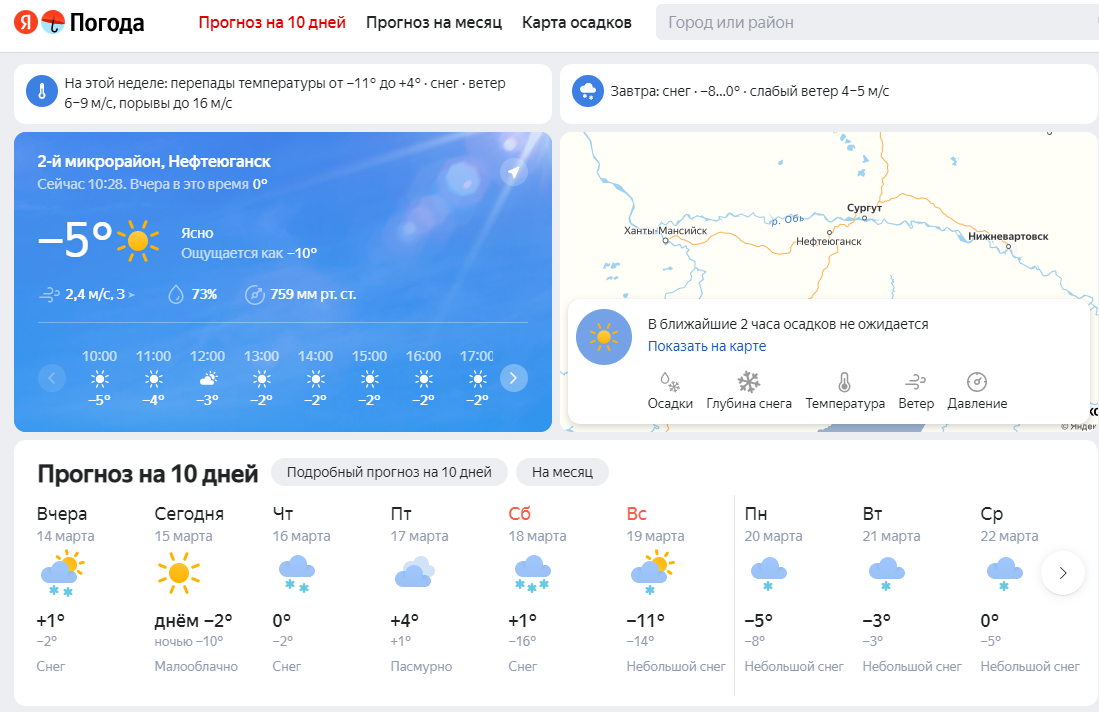
4. Metanit.com – сайт о программировании [Электронный ресурс] - <https://metanit.com/sharp/>

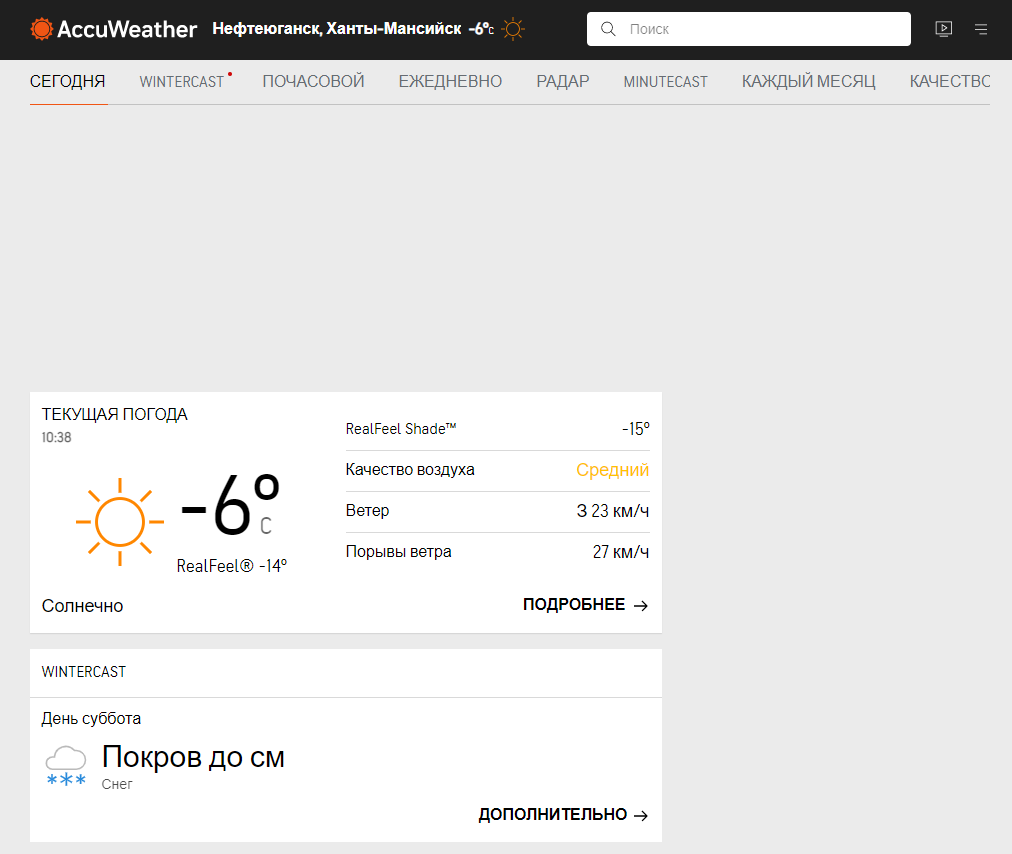
5. Робсон Э. Изучаем HTML, XHTML и CSS Питер 1 том 2-е издание 2020 - 720 с.

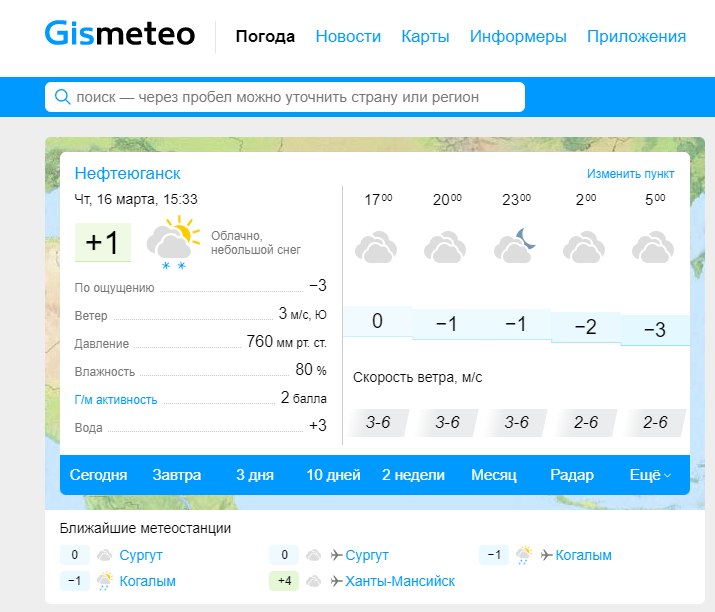
6. Дакетт Джон HTML и CSS. Разработка и дизайн вебсайтов. Эксмо 2020 - 480 с.

**Приложения**

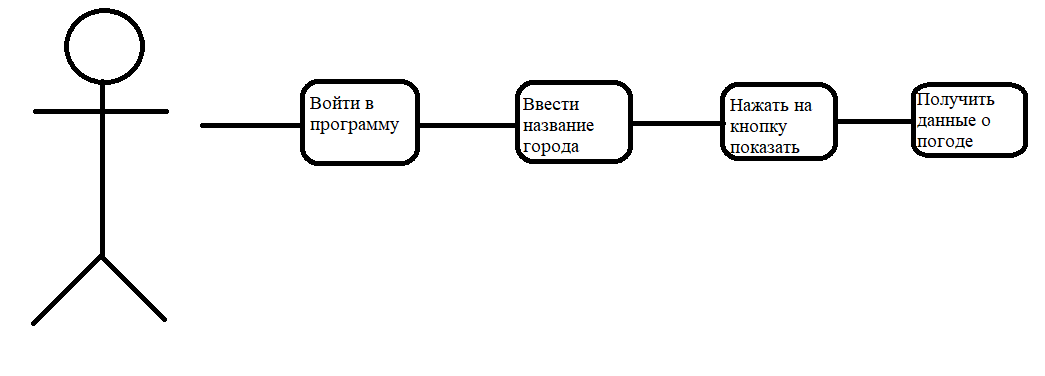
**Приложение 1**

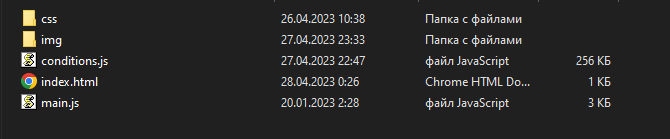






**Приложение 2**





Приложение 3

