|  |  |
| --- | --- |
| Описание: Рисунок1 | 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 \_\_\_курса \_\_\_\_\_\_\_\_\_\_\_\_ Зарифуллин В.С

(подпись)

Руководитель \_\_\_\_\_\_\_\_\_\_\_\_ Кравченко В. В.

(подпись)

Нефтеюганск 2021 г

Содержание

[Введение 3](#_Toc72143157)

[1. Теоретическая часть 5](#_Toc72143158)

[1.1. Описание предметной области 5](#_Toc72143159)

[1.2. Обоснование необходимости разработки 9](#_Toc72143160)

[1.3. Постановка задачи 10](#_Toc72143161)

[1.4. Состав ПЭВМ 11](#_Toc72143162)

[1.5. Используемое программное обеспечение 14](#_Toc72143163)

[2. Практическая часть 17](#_Toc72143164)

[2.1. Выбор средства автоматизации 17](#_Toc72143166)

[2.2. Дополнительные программные средства 21](#_Toc72143167)

[2.3. Требования к программе 26](#_Toc72143168)

[2.4. Проектирование программного продукта 27](#_Toc72143169)

[2.5. Текст программы с описанием 29](#_Toc72143170)

[2.6. Тестирование и отладка 36](#_Toc72143171)

[Заключение 38](#_Toc72143172)

[Список литературы 40](#_Toc72143173)

# Введение

Все чаще в жизни людей возникает потребность в использовании компьютерных технологий. Стремительное развитие информационных технологий, повышение скорости передачи данных, увеличение ресурсов глобальной сети, динамическое развитие технологий по сбору и анализу погодных данных ведет к появлению альтернативных способов предоставления информации о погодных условиях. Одним из таких приложений является приложение для мониторинга погоды относительно местности, в которой расположен субъект.

Еще несколько лет назад для того чтобы узнать погодные данные человеку было необходимо читать газету, смотреть телевизор в ожидании данных о погоде и погодных условиях которые могли быть не актуальны для зрителя именно в этот момент времени. Сегодня у любого интернет-пользователя есть возможность бесплатно отслеживать данные о погоде, а также прогноз погоды на ближайшие дни в режиме реального времени на специальных сайтах посвященных данной тематике.

Сайты, предоставляющие информацию о прогнозе погоды, являются очень распространенным явлением на сегодняшний день, и многие пользователи никак не могут определиться на каком из сайтов информация будет достовернее других. Поэтому им приходится ходить по множеству интернет ресурсов и сравнивать данные для того чтобы узнать актуальную погоду в данный момент времени.

Актуальность данной работы подтверждается тем фактом, что с каждым днем появляется все больше источников предоставляющих погодные данные, тем самым возникает необходимость в приложениях, которые могли бы в удобной структурированной форме их показывать пользователю в режиме реального времени.

Объект исследования: Разработка погодных приложений с использованием данных о погоде.

Предмет исследования: вывод данных о погоде с нескольких источников в одном приложении с использованием средств автоматизации разработки и дополнительных программных средств.

Целью данного курсового задания является создание приложения, позволяющего облегчить пользователю доступ к множеству информации о погодных данных выбранного региона прямо с сайтов, с которых обычно собиралась информация.

Задачами курсовой работы в связи с поставленной целью являются:

* на основе выбранных алгоритмов создать код приложения включающий также коды интерфейсных компонентов (пользовательских форм);
* разработать интерфейс взаимодействия с пользователем;
* протестировать программный продукт с целью выявления ошибок;
* опубликовать полученный программный продукт в облачном сервисе для предоставления общего доступа к нему;

Из всего вышеперечисленного, достижение поставленной задачи курсового задания осуществляется путем разработки программного средства.

# Теоретическая часть

## Описание предметной области

Целью курсовой работы является разработка компьютерного приложения для мониторинга данных о погоде по населенному пункту. В рамках текущей работы предлагается следующее решение поставленной задачи: приложение, в функционал которого входит возможность визуализации полученных от погодных сайтов, данных в простом и удобочитаемом виде.

Приложение ориентированно на обычных людей, которым необходимо узнать достоверную информацию о погоде в населенном пункте, где они проживают в реальном времени.

Используемыми платформами для персонального компьютера являются Windows, Mac OS и Linux. В свою очередь для мобильных устройств: iOS, Android и Windows Phone.

На данный момент существует несчетное множество способов визуализации данных о погоде где-либо. Существуют отдельные телепередачи, пункты в газетах, погоду рассказывают по радио, показывается на сайтах. Все способы для получения информации о погоде имеют разные возможности. Из множества способов для визуализации данных о погоде, самыми распространенными являются сайты или приложения для различных платформ. Есть приложения, которым необходимо подключение к интернету, есть приложения на которых, кроме того, чтобы узнать погоду можно узнать прогнозы на следующий день/месяц/неделю. Например, популярный Российский сайт, предоставляющий данные о погоде жителям России – Gismeteo.ru.

Гисметео (Рисунок 1.1) — один из самых популярных российских сайтов о погоде. Сайт был создан [12 декабря 2000 года](https://ru.wikinews.org/w/index.php?title=12_%D0%B4%D0%B5%D0%BA%D0%B0%D0%B1%D1%80%D1%8F_2000_%D0%B3%D0%BE%D0%B4%D0%B0&action=edit&redlink=1), однако прогнозы погоды [публиковались](https://www.gismeteo.ru/about.html) с середины 1990-х на BBS и в сети [FIDONet](https://ru.wikinews.org/w/index.php?title=FIDONet&action=edit&redlink=1). На сайте представлена информация о погоде в различных городах России и других стран мира. Кроме прогноза температуры, имеется информация об осадках, направлении ветра, давлении, влажности, видимости, продолжительности дня и магнитных бурях, также имеется сервис радара осадков и гроз. Сайт предоставляет API для профессиональных метеорологов. Владелец ресурса — компания «Мэп Мейкер».



Рисунок 1.1 - Интерфейс сайта Gismeteo

Есть также другой сайт – OpenWeatherMap (Рисунок 1.2)  - онлайн сервис, который предоставляет платный (есть функционально ограниченная бесплатная версия) [API](https://ru.wikipedia.org/wiki/%D0%98%D0%BD%D1%82%D0%B5%D1%80%D1%84%D0%B5%D0%B9%D1%81_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F_%D0%BF%D1%80%D0%B8%D0%BB%D0%BE%D0%B6%D0%B5%D0%BD%D0%B8%D0%B9) 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.

Информация обрабатываются OpenWeatherMap, после чего, на основе данных строится прогноз погоды и [погодные карты](https://ru.wikipedia.org/wiki/%D0%A1%D0%B8%D0%BD%D0%BE%D0%BF%D1%82%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B0%D1%8F_%D0%BA%D0%B0%D1%80%D1%82%D0%B0), например, карты облачности и осадков. Основной идеей сервиса OWM является использование частных погодных станций, которые помогают повысить точность исходной погодной информации и, как следствие, точность прогнозов погоды.

Вдохновителями OpenWeatherMap были картографический сервис [OpenStreetMap](https://ru.wikipedia.org/wiki/OpenStreetMap) и [Википедия](https://ru.wikipedia.org/wiki/%D0%92%D0%B8%D0%BA%D0%B8%D0%BF%D0%B5%D0%B4%D0%B8%D1%8F), создающие свободную информацию и предоставляющие её бесплатно для всех. В настоящее время OWM является коммерческой организацией.

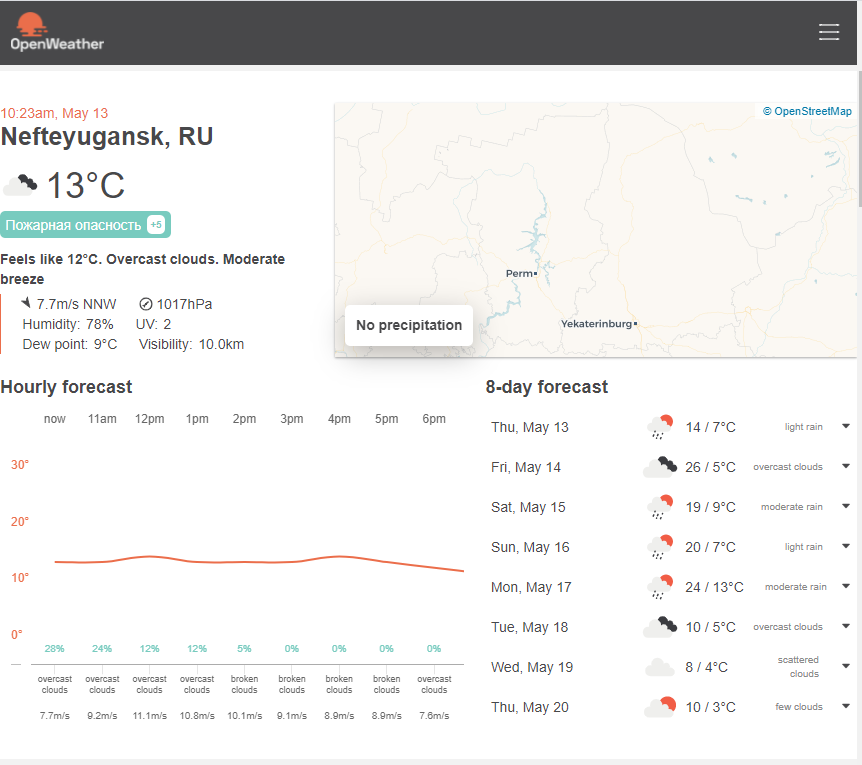


Рисунок 1.2 – Интерфейс сайта OpenWeatherMap

При исследовании приложений было найдено популярное приложение для платформы Android:

Яндекс Погода (Рисунок 1.3) – С помощью технологии Meteum Яндекс может рассчитать локальный прогноз погоды - с точностью до дома. Meteum знает про температуру и влажность воздуха, про атмосферное давление, про ветер, про осадки - дождь и снег.

Приложение Яндекс Погода показывает прогноз на сегодня, на завтра или на целую неделю для всего города, отдельного района, станции метро или конкретного адреса.

Можно задать список мест, погода в которых вас интересует, и быстро переключаться между ними.

Если включить уведомления о резких изменениях погоды, Яндекс Погода предупредит о похолодании, оттепели или начале череды дождей.

В приложении есть красивая карта, на которой можно в реальном времени следить за осадками. Сейчас она охватывает территории в радиусе двухсот километров.

Предусмотрены виджеты для главного экрана и панели уведомлений - с ними вы всегда можно быстро узнать, сколько градусов на улице, и найти что-нибудь в Яндексе. В настройках можно изменить облик и содержание виджетов.

Если сдвинуть экран вправо, покажутся дополнительные сведения о погоде. Там и информация о ветре, и температура по ощущениям, и давление и влажность, и время восхода и заката солнца.

В приложении есть специальный экран, на котором можно сообщить Яндексу о погоде. Набранная статистика используется для уточнения прогноза. Яндекс Погода приспособлена и для смартфонов, и для планшетов.

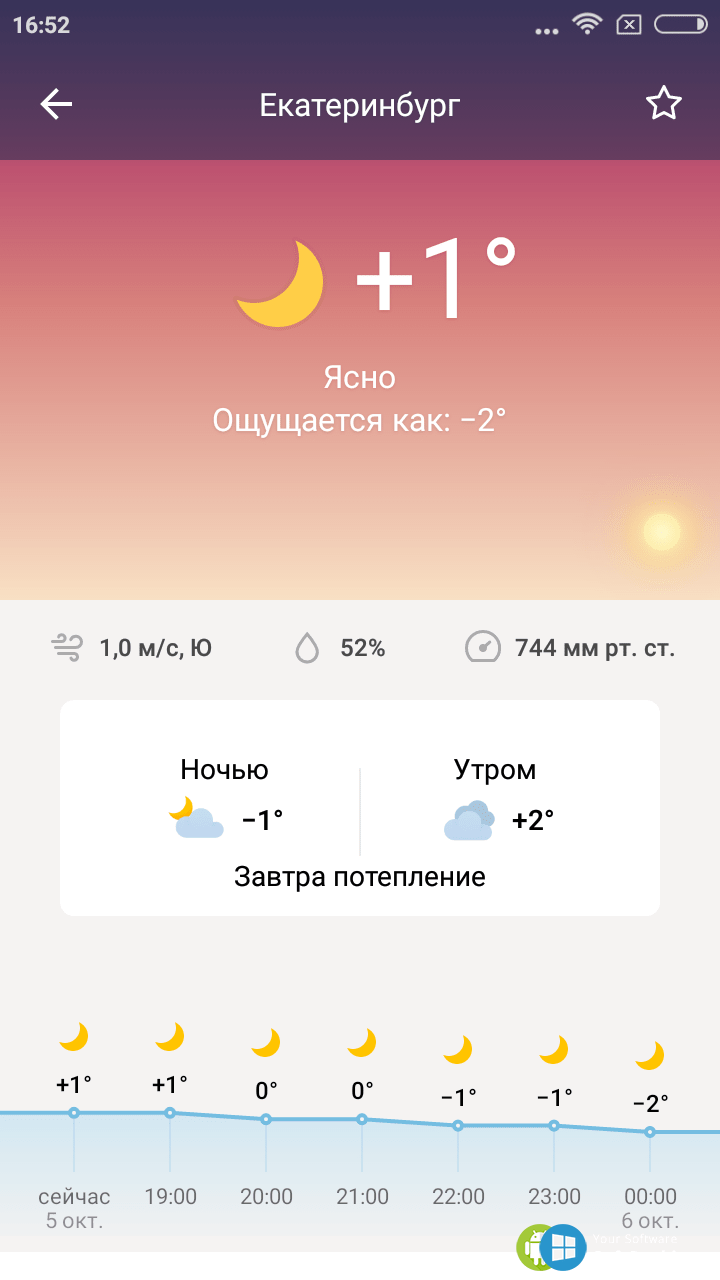


Рисунок 1.3 – интерфейс Яндекс Погоды

## Обоснование необходимости разработки

Технический процесс не стоит на месте и с каждым годом появляются все больше источников информации об окружающем мире, создаются все более новые приборы для исследования окружающий среды. Со временем люди научились даже предсказывать природные погодные явления. Собираемые данные позволяют узнавать: температуру воздуха, скорость ветра, количество выпадающих осадков, индекс ультрафиолетового облучения и еще много других данных. Но все эти данные собирают метеорологические станции и специально обученные люди, но возникает вопрос как же обычным людям узнавать погоду. Для этого были созданы специальные организации, занимающиеся сбором, и обработкой всех данных для того чтобы в максимально простой форме изложить их обычным людям. Позже технологический прогресс позволил людям прямо по телевизору, радио или другими способами не выходя из дома заранее узнавать погодные условия. Смотря на то сколько различных источников для того чтобы узнать погоду существует, люди задумываются в достоверности данных и поэтому стало обыденным смотреть погоду сразу с нескольких источников. Разрабатываемое приложение будет призвано ускорить и автоматизировать процесс изучения погодных данных с множества источников всего лишь одном приложении.

## Постановка задачи

Целью работы является разработка приложения на операционной системе Windows. Для достижения этой цели будет использована среда программирования Python 3.

Основными задачами, которые должны быть решены во время выполнения работы, являются:

* Построение алгоритма - наиболее эффективно математическую модель можно реализовать на компьютере в виде алгоритмической модели. Для этого может быть использован язык блок-схем;
* Программирование - программирование на определенном языке в определенной системе программирования. Программирование включает в себя следующие виды работ: выбор языка программирования; уточнение способов организации данных; запись алгоритма на выбранном языке программирования;
* Разработка удобной пользовательского интерфейса;
* Отладка и тестирование программы - под отладкой программы понимается процесс испытания работы программы и исправления обнаруженных при этом ошибок. Обнаружить ошибки, связанные с нарушением правил записи программы на языке программирования, помогает используемая система программирования. Пользователь получает сообщение об ошибке, исправляет ее и снова повторяет попытку исполнить программу;

«Мониторинг погоды» – компьютерное приложение, с помощью которого можно будет получать данные о погоде (Температура, скорость ветра, UV индекс и пр.) прямо с сайтов на которых предоставляются эти данные, без необходимости на них заходить. Приложение будет, устанавливается на компьютер пользователя, и запускаться одним “.exe” файлом. Пользователь, вводя лишь название города, в котором он хочет узнать погоду, сможет получить все необходимые данные в удобном интерфейсе, без необходимости регистрироваться или выполнять какие-либо еще действия, как если бы это происходило на сайте полностью на русском языке. При необходимости пользователь сможет перейти по ссылке на официальный сайт и там увидеть интересующую его информацию в полном виде.

## Состав ПЭВМ

Оперативная память (RAM — Random Access Memory) — это массив кристаллических ячеек, способных хранить данные. Существует много различных типов оперативной памяти, но с точки зрения физического принципа действия различают: - динамическую память (DRAM) - статическую память (SRAM). Ячейки динамической памяти (DRAM) можно представить в виде микро конденсаторов, способных накапливать заряд на своих обкладках. Это наиболее распространенный и экономически доступный тип памяти. Недостатки этого типа связаны, во-первых, с тем, что как при заряде, так и при разряде конденсаторов неизбежны переходные процессы, то есть запись данных происходит сравнительно медленно. Второй важный недостаток связан с тем, что заряды ячеек имеют свойство рассеиваться в пространстве, причем весьма быстро. Если оперативную память постоянно не «подзаряжать», утрата данных происходит через несколько сотых долей секунды. Для борьбы с этим явлением в компьютере происходит постоянная регенерация (освежение, подзарядка) ячеек оперативной памяти. Регенерация осуществляется несколько десятков раз в секунду и вызывает непроизводительный расход ресурсов вычислительной системы.

Процессор — основная микросхема компьютера, в которой и производятся все вычисления. Конструктивно процессор состоит из ячеек, похожих на ячейки оперативной памяти, но в этих ячейках данные могут не только храниться, но и изменяться. Внутренние ячейки процессора называют регистрами. Важно также отметить, что данные, попавшие в некоторые регистры, рассматриваются не как данные, а как команды, управляющие обработкой данных в других регистрах. Таким образом, управляя засылкой данных в разные регистры процессора, можно управлять обработкой данных. На этом и основано исполнение программ. С остальными устройствами компьютера, и в первую очередь с оперативной памятью, процессор связан несколькими группами проводников, называемых шинами. Основных шин три: шина данных, адресная шина и командная шина.

Тактовая частота Исполнение каждой команды занимает определенное количество тактов. Чем выше частота тактов, поступающих на процессор, тем больше команд он может исполнить в единицу времени, тем выше его производительность. Первые процессоры х86 могли работать с частотой не выше 4,77 МГц, а сегодня рабочие частоты некоторых процессоров уже превосходят 3 миллиарда тактов в секунду (3 ГГц). Кэш-память Обмен данными внутри процессора происходит в несколько раз быстрее, чем обмен с другими устройствами, например, с оперативной памятью. Для того чтобы уменьшить количество обращений к оперативной памяти, внутри процессора создают буферную область — так называемую кэш-память. Это как бы «сверхоперативная память». Когда процессору нужны данные, он сначала обращается в кэш-память, и' только если там нужных данных нет, происходит его обращение в оперативную память.

Видеокарта (видеоадаптер) Совместно с монитором видеокарта образует видеоподсистему персонального компьютера. Физически видеоадаптер выполнен в виде отдельной дочерней платы, которая вставляется в один из слотов материнской платы и называется видеокартой. Видеоадаптер взял на себя функции видеоконтроллера, видеопроцессора и видеопамяти. Параметры видеоподсистемы.

Разрешение экрана (является одним из важнейших). Для каждого размера монитора существует свое оптимальное разрешение экрана, которое должен обеспечивать видеоадаптер

Цветовое разрешение (глубина цвета) определяет количество различных оттенков, которые может принимать отдельная точка экрана.

Видео ускорение — одно из свойств видеоадаптера, которое заключается в том, что часть операций по построению изображений может происходить без выполнения математических вычислений в основном процессоре компьютера, а чисто аппаратным путем — преобразованием данных в микросхемах видео ускорителя.

Для нормальной работы программы необходим IBM совместимый компьютер с тактовой частотой процессора не ниже 144МГц и выше, оперативной памятью 2Gb, жесткий диск объемом не менее 500Мб.

Минимальные системные требования для работы программы «Мониторинг погоды» разрабатываемой на языке Python 3:

* Процессор: Intel Core i3 или аналогичный
* Оперативная память: 2гб или выше
* Видеокарта: любая видеокарта с объемом видеопамяти не менее 500 мб. Например: GeForce 7300 GT, или Intel HD Graphics 620.

Рекомендуемые системные требования:

* Процессор: двухъядерный Intel Core i5 или аналогичный
* Оперативная память: 4гб
* Видеокарта: любая видеокарта с объемом видеопамяти не менее 1гб. Подойдут: Intel HD Graphics 610, NVIDIA GeForce 930MX

Полученный в результате программный продукт должен работать на любом компьютере под управлением операционной системы от Windows 7 и выше.

## Используемое программное обеспечение

Данное задание необходимо реализовать в среде программирования Python 3.8. Данная среда программирования представляет хорошие возможности для разработки такого приложения и является наиболее изученной. Она позволяет достаточно быстро создать приложение для Windows за счет упрощения разработки кодовой части проекта.

Она эффективна, надежна и при эксплуатации обладает предсказуемым поведением. Все вышесказанное и стало причиной выбора данной среды программирования для создания приложения.

Язык программирования, который используется в одноимённой среде разработки:

Язык Python был задуман как потомок [языка программирования ABC](https://ru.wikipedia.org/wiki/ABC_(%D1%8F%D0%B7%D1%8B%D0%BA_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F)), способный к обработке исключений и взаимодействию с операционной системой [Амёба](https://ru.wikipedia.org/wiki/Amoeba_(%D0%BE%D0%BF%D0%B5%D1%80%D0%B0%D1%86%D0%B8%D0%BE%D0%BD%D0%BD%D0%B0%D1%8F_%D1%81%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0)). Ван Россум является основным автором Python и продолжал выполнять центральную роль в принятии решений относительно развития языка вплоть до 12 июля 2018 года.

Python  [высокоуровневый язык программирования](https://ru.wikipedia.org/wiki/%D0%92%D1%8B%D1%81%D0%BE%D0%BA%D0%BE%D1%83%D1%80%D0%BE%D0%B2%D0%BD%D0%B5%D0%B2%D1%8B%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F) общего назначения с [динамической](https://ru.wikipedia.org/wiki/%D0%94%D0%B8%D0%BD%D0%B0%D0%BC%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B0%D1%8F_%D1%82%D0%B8%D0%BF%D0%B8%D0%B7%D0%B0%D1%86%D0%B8%D1%8F) [строгой](https://ru.wikipedia.org/wiki/%D0%A1%D1%82%D1%80%D0%BE%D0%B3%D0%B0%D1%8F_%D1%82%D0%B8%D0%BF%D0%B8%D0%B7%D0%B0%D1%86%D0%B8%D1%8F) типизацией и автоматическим управлением памятью, ориентированный на повышение производительности разработчика, читаемости кода и его качества, а также на обеспечение переносимости написанных на нём программ. Язык является полностью объектно-ориентированным  — всё является объектами. Необычной особенностью языка является выделение блоков кода пробельными отступами. [Синтаксис](https://ru.wikipedia.org/wiki/%D0%A1%D0%B8%D0%BD%D1%82%D0%B0%D0%BA%D1%81%D0%B8%D1%81_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)) ядра языка минималистичен, за счёт чего на практике редко возникает необходимость обращаться к документации. Сам же язык известен как интерпретируемый и используется, в том числе для написания скриптов.

В качестве вспомогательных языков программирования были выбраны:

HTML (Hyper Text Markup Language) означает язык разметки гипертекста. Этот язык был разработан Тимом Бернерсом-Ли в рамках создания проекта распределенной гипертекстовой системы, которую он назвал World Wide Web (WWW) или Всемирная паутина. HTML предназначен для написания гипертекстовых документов, публикуемых в World Wide Web. Документ на языке HTML может включать следующие компоненты:

* стилизованный и форматированный текст;
* команды включения графических и звуковых файлов;
* гиперсвязи с различными ресурсами Internet;
* скрипты на языке JavaScript и VBScript;
* различные объекты, например, Flash-анимацию.

CSS – это формальный язык, служащий для описания оформления внешнего вида документа, созданного с использованием языка разметки (HTML, XHTML, XML). Название происходит от английского Cascading Style Sheets, что означает «каскадные таблицы стилей»

JavaScript – это мультипарадигменный язык программирования, который обычно применяется в качестве встраиваемого инструмента для программного доступа к различным объектам приложений. Язык JS – это то, что «оживляет» разметку страниц (HTML) и пользовательский функционал (CMS) сайтов. Он полностью совместим с операционными системами Windows, Linux, Mac OS, а также всеми популярными мобильными платформами.

# Практическая часть



## Выбор средства автоматизации

Программы необходимые для работы со средой разработки Python:

- IDLE Python - это **интегрированная среда разработки для изучения Python**, которая является альтернативой командной строке. Некоторые операционные системы поставляются с интерпретатором Python, но в других системах может потребоваться его установка. В Gnu / Linux, **IDLE распространяется как отдельное приложение** которые можно установить из репозиториев каждого дистрибутива.

Эта IDE**бесплатно и с открытым исходным кодом**. Хотя его интерфейс (Рисунок 2.1) гораздо менее богато украшен, чем у других IDE, в IDLE нет недостатка в функциях.

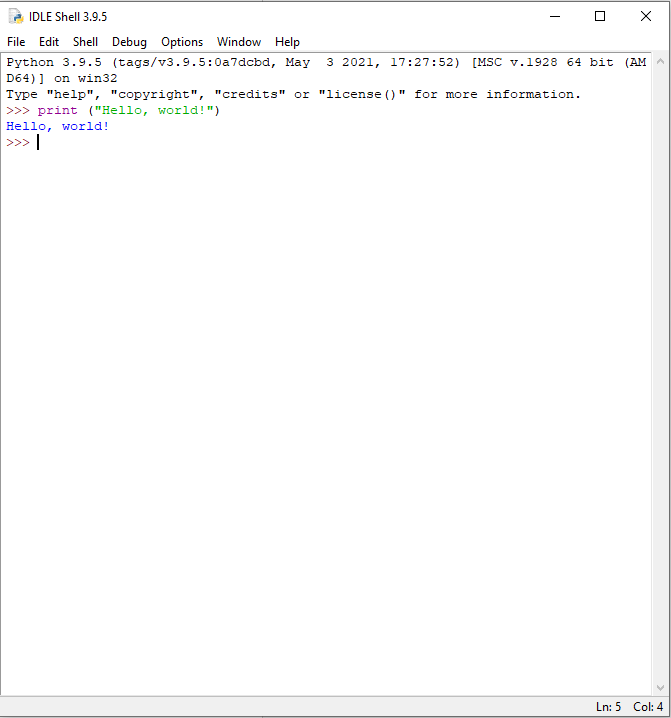


Рисунок 2.1 – Интерфейс IDLE Python 3.9.5

- Sublime Text - [проприетарный](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D0%BF%D1%80%D0%B8%D0%B5%D1%82%D0%B0%D1%80%D0%BD%D0%BE%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%BD%D0%BE%D0%B5_%D0%BE%D0%B1%D0%B5%D1%81%D0%BF%D0%B5%D1%87%D0%B5%D0%BD%D0%B8%D0%B5) [текстовый редактор](https://ru.wikipedia.org/wiki/%D0%A2%D0%B5%D0%BA%D1%81%D1%82%D0%BE%D0%B2%D1%8B%D0%B9_%D1%80%D0%B5%D0%B4%D0%B0%D0%BA%D1%82%D0%BE%D1%80) (Рисунок 2.2). Поддерживает [плагины](https://ru.wikipedia.org/wiki/%D0%9F%D0%BB%D0%B0%D0%B3%D0%B8%D0%BD) на языке программирования [Python](https://ru.wikipedia.org/wiki/Python).

Некоторые возможности[[4]](https://ru.wikipedia.org/wiki/Sublime_Text#cite_note-4):

Быстрая навигация (Goto Anything)

Командная палитра (Command Palette)

[API](https://ru.wikipedia.org/wiki/API) [плагинов](https://ru.wikipedia.org/wiki/%D0%9F%D0%BB%D0%B0%D0%B3%D0%B8%D0%BD) на [Python](https://ru.wikipedia.org/wiki/Python)

Одновременное редактирование (Split Editing)

Высокая степень настраиваемости (Customize Anything)

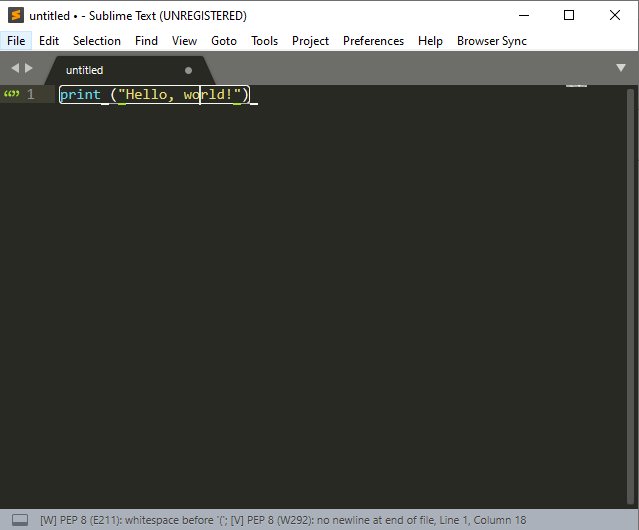


Рисунок 2.2 – Интерфейс Sublime Text 3

- Visual Studio Code — это [редактор исходного кода](https://ru.wikipedia.org/wiki/%D0%A0%D0%B5%D0%B4%D0%B0%D0%BA%D1%82%D0%BE%D1%80_%D0%B8%D1%81%D1%85%D0%BE%D0%B4%D0%BD%D0%BE%D0%B3%D0%BE_%D0%BA%D0%BE%D0%B4%D0%B0). Он поддерживает ряд [языков программирования](https://ru.wikipedia.org/wiki/%D0%AF%D0%B7%D1%8B%D0%BA_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F), [подсветку синтаксиса](https://ru.wikipedia.org/wiki/%D0%9F%D0%BE%D0%B4%D1%81%D0%B2%D0%B5%D1%82%D0%BA%D0%B0_%D1%81%D0%B8%D0%BD%D1%82%D0%B0%D0%BA%D1%81%D0%B8%D1%81%D0%B0), [IntelliSense](https://ru.wikipedia.org/wiki/IntelliSense), [рефакторинг](https://ru.wikipedia.org/wiki/%D0%A0%D0%B5%D1%84%D0%B0%D0%BA%D1%82%D0%BE%D1%80%D0%B8%D0%BD%D0%B3), [отладку](https://ru.wikipedia.org/wiki/%D0%9E%D1%82%D0%BB%D0%B0%D0%B4%D0%BA%D0%B0_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D1%8B), навигацию по коду[[18]](https://ru.wikipedia.org/wiki/Visual_Studio_Code#cite_note-18), поддержку [Git](https://ru.wikipedia.org/wiki/Git) и другие возможности. Многие возможности Visual Studio Code недоступны через [графический интерфейс](https://ru.wikipedia.org/wiki/%D0%93%D1%80%D0%B0%D1%84%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B8%D0%B9_%D0%B8%D0%BD%D1%82%D0%B5%D1%80%D1%84%D0%B5%D0%B9%D1%81_%D0%BF%D0%BE%D0%BB%D1%8C%D0%B7%D0%BE%D0%B2%D0%B0%D1%82%D0%B5%D0%BB%D1%8F), зачастую они используются через палитру команд или [JSON](https://ru.wikipedia.org/wiki/JSON)-файлы (например, пользовательские настройки). Палитра команд представляет собой подобие [командной строки](https://ru.wikipedia.org/wiki/%D0%98%D0%BD%D1%82%D0%B5%D1%80%D1%84%D0%B5%D0%B9%D1%81_%D0%BA%D0%BE%D0%BC%D0%B0%D0%BD%D0%B4%D0%BD%D0%BE%D0%B9_%D1%81%D1%82%D1%80%D0%BE%D0%BA%D0%B8), которая вызывается [сочетанием клавиш](https://ru.wikipedia.org/wiki/%D0%A1%D0%BE%D1%87%D0%B5%D1%82%D0%B0%D0%BD%D0%B8%D0%B5_%D0%BA%D0%BB%D0%B0%D0%B2%D0%B8%D1%88).

С 2018 года появилось расширение [Python](https://ru.wikipedia.org/wiki/Python) для Visual Studio Code с открытым исходным кодом. Оно предоставляет разработчикам широкие возможности для редактирования, отладки и тестирования кода. Интерфейс программы изображен на Рисунке 2.3.

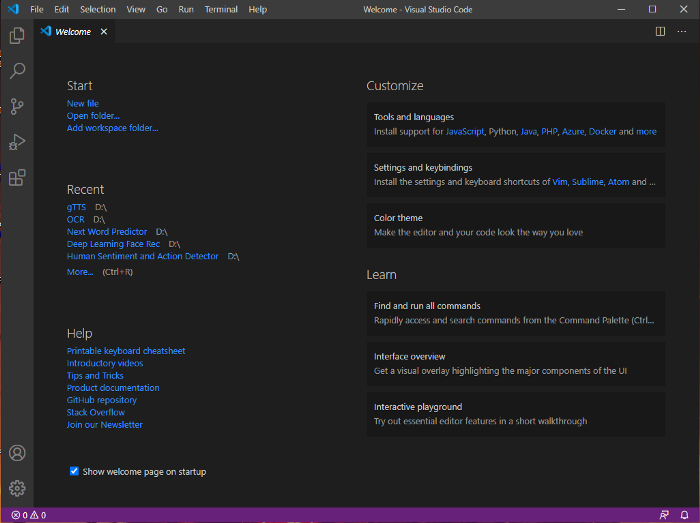


Рисунок 2.3 – Интерфейс Visual Studio Code

- PyCharm — [интегрированная среда разработки](https://ru.wikipedia.org/wiki/%D0%98%D0%BD%D1%82%D0%B5%D0%B3%D1%80%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%BD%D0%B0%D1%8F_%D1%81%D1%80%D0%B5%D0%B4%D0%B0_%D1%80%D0%B0%D0%B7%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%BA%D0%B8) для языка программирования [Python](https://ru.wikipedia.org/wiki/Python). Предоставляет средства для анализа кода, графический отладчик, инструмент для запуска юнит-тестов и поддерживает веб-разработку на [Django](https://ru.wikipedia.org/wiki/Django). PyCharm разработана компанией [JetBrains](https://ru.wikipedia.org/wiki/JetBrains)[[5]](https://ru.wikipedia.org/wiki/PyCharm#cite_note-5) на основе [IntelliJ IDEA](https://ru.wikipedia.org/wiki/IntelliJ_IDEA).

PyCharm — это кросс-платформенная среда разработки, которая совместима с [Windows](https://ru.wikipedia.org/wiki/Windows), [macOS](https://ru.wikipedia.org/wiki/MacOS), [Linux](https://ru.wikipedia.org/wiki/Linux). PyCharm Community Edition (бесплатная версия) находится под лицензией [Apache License](https://ru.wikipedia.org/wiki/%D0%9B%D0%B8%D1%86%D0%B5%D0%BD%D0%B7%D0%B8%D1%8F_Apache), а PyCharm Professional Edition (платная версия) является [проприетарным ПО](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D0%BF%D1%80%D0%B8%D0%B5%D1%82%D0%B0%D1%80%D0%BD%D0%BE%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%BD%D0%BE%D0%B5_%D0%BE%D0%B1%D0%B5%D1%81%D0%BF%D0%B5%D1%87%D0%B5%D0%BD%D0%B8%D0%B5). Интерфейс программы показан на Рисунке 2.4.

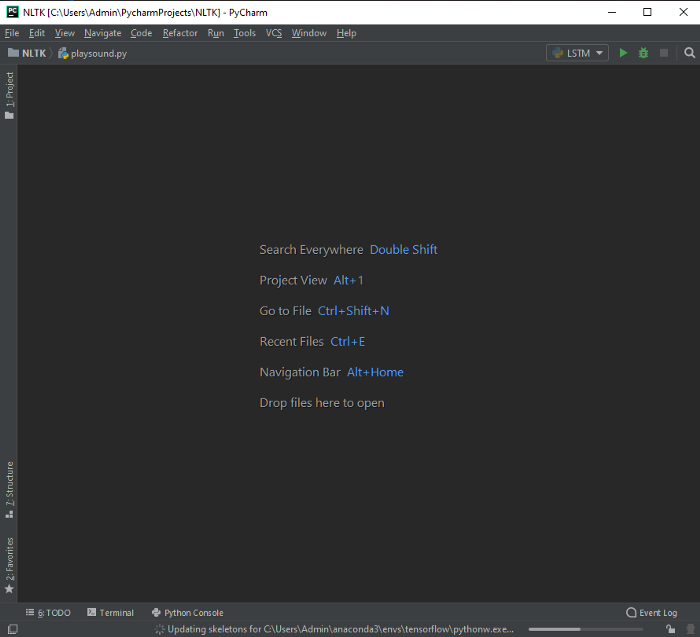


Рисунок 2.4 – Интерфейс PyCharm

После исследования доступных сред для разработки на языке программирования Python была выбрана среда разработки Sublime Text 3. Причиной выбора данной среды разработки является ее простота и возможность установки любым необходимых плагинов для гибкой настройки.

## Дополнительные программные средства

В программном продукте «Мониторинг погоды» при разработке появилась необходимость в использовании API для получения метеорологических данных о погоде.

API — это спецификация возможных взаимодействий с компонентом ПО. Например, если бы машина была компонентом ПО, в ее API содержалась бы информация о том, что машина может ускоряться, тормозить и включать радио. Ее API также бы включало информацию о том, как ускоряться.

API работает на 3 уровнях:

Приложение: Это приложения, которые вы используете на смартфоне, или ПО.

Программирование: Программисты используют API, чтобы писать код.

Интерфейс: Как вы взаимодействуете с приложением.

В приложении были использованы API трех сайтов с метеорологическими данными : OpenWeatherMap, WeatherStack, Yahoo Weather.

OpenWeatherMap использует платный API (есть функционально ограниченный бесплатный доступ), чтобы предоставить данные текущей погоды, прогноза и карт с погодными явлениями, такими как облака, ветер, давление и осадки. Все погодные данные могут быть получены в форматах JSON, XML или HTML Пример ответа OpenWeatherMap Api изображён на Рисунке 3.2.

Текущие погодные данные могут быть найдены по городу (более 200.000 доступных городов) или географическими координатам. Данные обновляются каждые 10 минут. Пример запроса с OpenWeatherMap Api изображён на Рисунке 3.1.



Рисунок 3.1 – Запрос к OpenWeatherMap Api на Python



Рисунок 3.2 – Ответ от OpenWeatherApi в формате JSON

Weatherstack API содержит набор инструментов для различных языков программирования, подкрепленных данными о погоде на основе местоположения. Доступны шесть конечных точек:

**Текущая погода: предоставляет** текущие данные о погоде.

**Историческая погода:** просмотр исторических данных о погоде.

**Исторические временные ряды:** Возвращает исторические временные ряды погодных данных.

**Прогноз погоды:** Получить прогноз погоды на срок до 14 дней.

**Поиск местоположения:** Дает возможность проверить погоду в одном или нескольких местах.

Между тем API Weatherstack также предоставляет почасовые данные о погоде для миллионов мест по всему миру. Данные доставляются в миллисекундах с сервера с почти 100-процентным временем безотказной работы. Примеры использования WeatherStack Api на Рисунке 3.3 и Рисунке 3.4



Рисунок 3.3 – Запрос к WeatherStack Api на Python



Рисунок 3.4 – Ответ от WeatherStack Api в формате JSON

Yahoo Weather API позволяет разработчикам интегрировать информацию о погоде для любого места в сторонние приложения. Доступная информация включает 5-дневный прогноз, ветер, атмосферные условия, осадки, время заката, астрономические условия и многое другое. Примеры использования Yahoo Weather Api изображены на Рисунке 3.5 и Риснке 3.6.



Рисунок 3.5 - Запрос к Yahoo Weather Api на Python

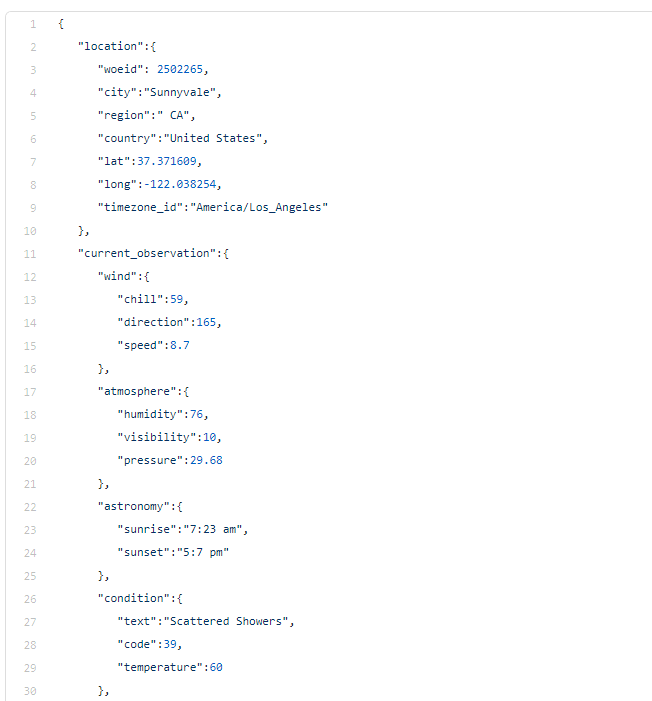


Рисунок 3.6 - Ответ от Yahoo Weather Api в формате JSON

## Требования к программе

Требования к функциональным характеристикам. Программа должна обеспечивать возможность выполнения перечисленных ниже функций:

* Отправлять запросы на сервер с метеорологическими данными.
* Принимать данные json,xml формата и работать с ними.
* Принимать значения, введенные пользователем и после обрабатывать их.
* Принятие данных из API сайтов.
* Переводить данные с английского на русский и с русского на английский
* Выводить данные о погоде в текстовом формате: Температура, ощущаемая температура, влажность, скорость ветра, направление ветра, индекс ультрафиолетового облучения, давление воздуха и пр.

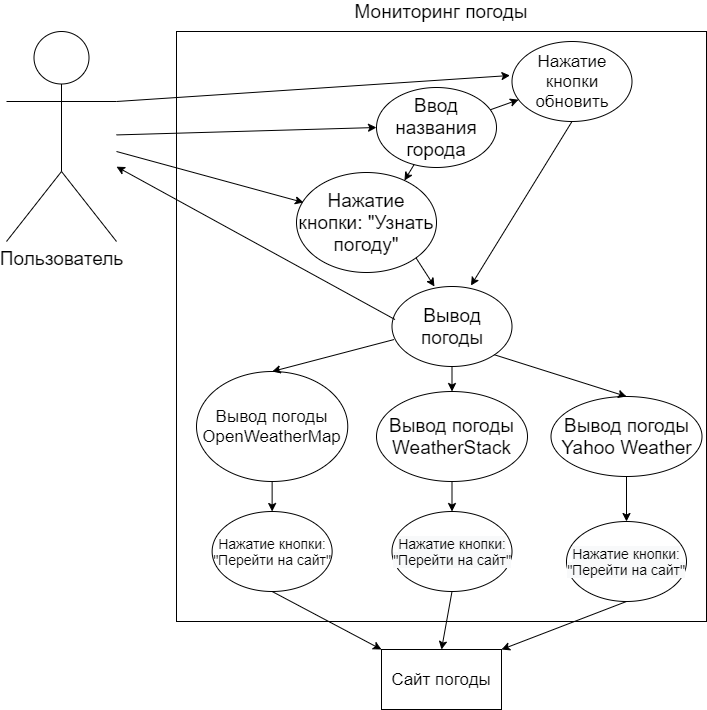


Рисунок 4 – USE-CASE схема программы

## Проектирование программного продукта

При создании программного продукта «Мониторинг погоды» был использован прием проектирования – 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](https://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BC%D0%BF%D1%8C%D1%8E%D1%82%D0%B5%D1%80%D0%BD%D0%B0%D1%8F_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B0). Во многих языках (но далеко не обязательно) оформляется в виде отдельного [файла](https://ru.wikipedia.org/wiki/%D0%A4%D0%B0%D0%B9%D0%BB) с [исходным кодом](https://ru.wikipedia.org/wiki/%D0%98%D1%81%D1%85%D0%BE%D0%B4%D0%BD%D1%8B%D0%B9_%D0%BA%D0%BE%D0%B4) или поименованной непрерывной её части. Некоторые языки предусматривают объединение модулей в [пакеты](https://ru.wikipedia.org/wiki/%D0%9F%D0%B0%D0%BA%D0%B5%D1%82_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)). UML Диаграмма проекта «Мониторинг погоды» изображена на Рисунке 5

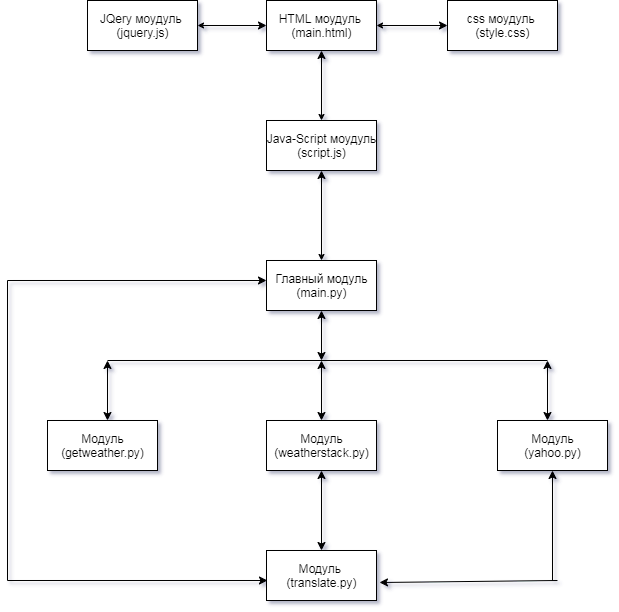


Рисунок 5 – UML Диаграмма модулей проекта «Мониторинг погоды»

## Текст программы с описанием

Программа «Мониторинг погоды» выводит данные о погоде с разных сайтов в соответствии с указанным городом.

Программный продукт создан с использованием следующих программных средств:

* Язык программирования Python 3
* Язык гипертекстовой разметки HTML
* Каскадная таблица стилей (CSS)
* Язык программирования скриптов Java-Script
  + jQuery — это популярный сегодня JavaScript-фреймворк, основной функцией которого является обеспечение взаимодействия между HTML и JavaScript на странице
* Текстовый редактор Sublime Text 3
* Для упрощения работы с часть кода на языке Python были использованы следующие библиотеки (модули):
* Eel - это небольшая библиотека Python для создания простых, подобных Electron, автономных приложений с графическим интерфейсом HTML / JS с полным доступом к возможностям и библиотекам Python. **Eel размещает локальный веб-сервер, а затем позволяет аннотировать функции в Python, чтобы их можно было вызывать из Javascript, и наоборот.** Eel разработан, чтобы упростить написание коротких и простых приложений с графическим интерфейсом.
* PyOWM - это клиентская библиотека-оболочка Python для веб-API OpenWeatherMap (OWM).
* Math – [Python math модуль](https://docs.python.org/3/library/math.html) является важным элементом предназначен для вычисления математических операций.
* Googletrans - это **бесплатная**и**неограниченная** библиотека Python, в которой реализован API Google Translate.
* Requests - библиотека позволяет нам легко и с минимальным количеством кода взаимодействовать с веб-приложениями. Это необходимо нам для решения любых задач, связанных с передачей информации от пользователя к серверу и обратно
* YahooWeather – библиотека для Python созданная с целью упростить взаимодействий программиста с api Yahoo Weather

Структура проекта представлена следующим образом. В папке с проектом на первом уровне находятся папка Web, Главный модуль программы, а также вторичные модули. В папке Web расположены файлы, связанные с формированием интерфейса html файл, стили, а также папка с скриптами, написанными на Java-Script. В другой папке расположены все имеющиеся в программе рисунки. Вся структура показана на Рисунке 6.1, Рисунке 6.2.

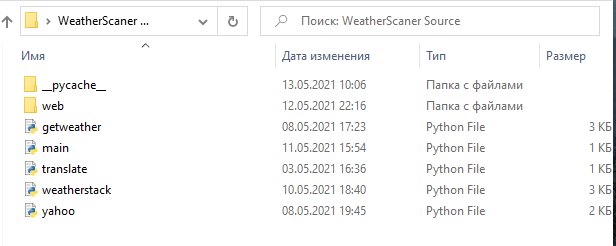


Рисунок 6.1 – Первый уровень проекта

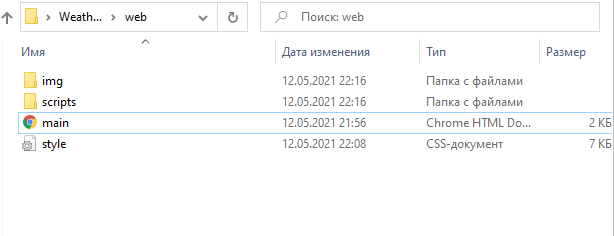


Рисунок 6.2 – Второй уровень – папка Web

Весь проект занимает:

* Размер: 25 МБ (26214400 байт)
* На диске: 25 МБ (26214400 байт)

Описание модулей и взаимосвязи между модулями программы:

Главный модуль программы называется main.py модуль служит для объединения всех остальным модулей и генерации интерфейса посредством использования библиотеки eel. В начале импортируется библиотека для создания интерфейсов с использованием html файлов eel. Также импортируются функции из модулей, отвечающих за вывод информации по соответствующему сайту. Далее идет объявление 3 функций, которые получают аргументом переменную place(город). Это значение переводится при помощи использования импортированной функции библиотеки googletrans на английский язык, поскольку большинство погодных api принимают лишь английские string значения. Каждая из функций исполняется в своем модуле и потом возвращает назад в java-script результат выполнения - погодные данные. Для того чтобы была возможность передавать значения между JS и Python используется функция “@eel.expose”. Функция, обозначенная этой значением может передавать результат в скрипты и обратно. Для создания интерфейса используются функции eel.init и eel.start. Первая позволяет выбрать директорию, в которой расположен исполняемый html файл, вторая позволяет выбрать сам файл и запустить его в отдельном от браузера окошке с указанными в параметрах размерами, как будто бы это было независимым веб приложением.

Текст модуля main.py:

import eel #подключение библиотеки веб интерфесов

from getweather import get\_weather

from weatherstack import weatherstack

from yahoo import yahoo\_weather

from translate import translate

@eel.expose #говорим, что эту функцию можно использовать с js

def main(place):#Объявляем функция показа погоды

translate(place) #отправляем значение place в переводчик

overlay = get\_weather(place)# запускаем функцию для вывода данных из OpenWeatherMap

return overlay #возвращаем результат выполнения функции

@eel.expose #

def two(place):

weat = weatherstack(place)

return weat

@eel.expose

def yahoo(place):

weat = yahoo\_weather(place)

return weat

eel.init("web") #Создание окна веб приложения

eel.start("main.html", size=(700, 700))#исполняемый html файл для окна веб приложения

Модуль, отвечающий за перевод текста с английского языка и обратно translate.py. Текст в этом модуле начинается с импорта библиотеки googletrans и ее главной функции Translator. Сам модуль состоит из одной функции, которая принимает на вход значение переменной place перед тек как исполнить ее в модуле с получением погоды и переводит ее на английский язык. Затем возвращает обратно в главный модуль.

Текст модуля translate.py:

import googletrans

from googletrans import Translator

def translate (place): # где place это город введенный пользователем

trans = Translator()

result = trans.translate(place)

return place # возвращает обратно на английском языке (по умолчанию)

Модуль отвечающий за отправку запроса на OpenWeatherMap Api строится следующим образом. Вначале импортируется библиотека pyowm, она служит для облегчения отправки запросов на языке Python, а также библиотека math которая нужна для использования сложных математических функций. Также импортируются настройки конфигурации модуля для того чтобы изменить язык текста на русский язык. Модуль представляет из себя функцию get\_weather(place) Данная функция использует полученное из главного модуля название города а также Api ключ и отправляет запрос на сервер, присваивая полученные данные переменным, которые позже выводятся как div или ul теги в html выводя список данных о погоде прямо с сайта OpenWeatherMap с иконками погоды, температурой, описанием, данными об осадках и прочим

Текст модуля getweather.py:

import pyowm #подключение библиотеки для взаимодействия с API OpenWeatherMap

import math

from pyowm.utils.config import get\_default\_config

config\_dict = get\_default\_config()

config\_dict['language'] = 'ru'

owm = pyowm.OWM("f17480b84563d8c6cde6484c071ca775")

def get\_weather(place):

city = place #присваиваем город к переменной city

mgr = owm.weather\_manager() #присваиваем модуль api к переменной mgr

observation = mgr.weather\_at\_place(city) # присваиваем mgr+погода для города

w = observation.weather

temp = math.ceil(w.temperature('celsius')['temp'])

temp\_feel = math.ceil(w.temperature('celsius').get('feels\_like', None))

details = w.detailed\_status

icon = w.weather\_icon\_url(size='2x')

result1 = "<li>" + "<span>" + details + "</span>" + "<div class='right'>" + "<img src=" + icon +" >" + "<div>" + "<p>" +str(temp)+ "℃" + "</p>" +"<p id='feels\_like'>" + "Ощущается:" + "</p>" + "<p>" +str(temp\_feel)+ "℃" + "</p>" + "</div>" + "</div>" + "</li>"

wind = w.wind()['speed']

winddeg = w.wind()['deg']

humidity = w.humidity

humidity\_full = "<div>" + "<img src='img/humidity.svg' class='icon-wind-direction'>" + "Влажность: " + str(humidity) + "%" + "</div>"

result2 = "<li>" + "<div>" + "<svg data-v-47880d39='' viewBox='0 0 1000 1000' enable-background='new 0 0 1000 1000' xml:space='preserve' class='icon-wind-direction' style='transform: rotate("+str(winddeg)+"deg);'>"+"<g data-v-47880d39='' fill='#48484a'>"+"<path data-v-47880d39='' d='M510.5,749.6c-14.9-9.9-38.1-9.9-53.1,1.7l-262,207.3c-14.9,11.6-21.6,6.6-14.9-11.6L474,48.1c5-16.6,14.9-18.2,21.6,0l325,898.7c6.6,16.6-1.7,23.2-14.9,11.6L510.5,749.6z'>"+"</path><path data-v-47880d39='' d='M817.2,990c-8.3,0-16.6-3.3-26.5-9.9L497.2,769.5c-5-3.3-18.2-3.3-23.2,0L210.3,976.7c-19.9,16.6-41.5,14.9-51.4,0c-6.6-9.9-8.3-21.6-3.3-38.1L449.1,39.8C459,13.3,477.3,10,483.9,10c6.6,0,24.9,3.3,34.8,29.8l325,898.7c5,14.9,5,28.2-1.7,38.1C837.1,985,827.2,990,817.2,990z M485.6,716.4c14.9,0,28.2,5,39.8,11.6l255.4,182.4L485.6,92.9l-267,814.2l223.9-177.4C454.1,721.4,469,716.4,485.6,716.4z'>"+"</path></g></svg>" + "Ветер: " + str(wind) + " м/с" + "</div>" + humidity\_full + "</li>"

visibilit = w.visibility() / 1000

visibilit\_full = "<div>" + "<img src='img/visibility.svg' class='icon- wind-direction'>" + "Видимость: " + str(visibilit) + "км" + "</div>"

pressur = w.pressure['press']

result3 = "<li>" + "<div>" + "<img src='img/gauge.svg' class='icon-wind-direction'>" + "Давление: " + str(pressur) + "hPa" + "</div>" + visibilit\_full + "</li>"

result = "<ul>"+result1+result2+result3+"</ul>"

return result

Два других модуля weatherstack.py и yahoo.py аналогично выполняют те же функции что и getweather.py с отличием в том, что они выполняет запросы для своих сайтов и формируют блоки для своих окошек в html. Для запуска программы нужно найти в папке dist .exe файл и запустить его.

## Тестирование и отладка

В процессе разработки приложения производилось поэтапное тестирование с целью выявления программных ошибок. Для этого был создан эмулятор ПЭВМ с разными диагоналями экрана для разных версий ОС. Тестируемый программный продукт запускался на этот эмулятор, его поведение анализировалось, и при необходимости по результатам анализа вносились изменения в код. Для тестирования отдельных модулей были внесены специальные функции, выводящие сообщение об ошибке в системный журнал. Они также известны как юнит-тесты.

Были проведены приведенные ниже тесты.

* Каждый блок был подвергнут юнит-тестированию с целью выявления ошибок, вызванных несоответствием ожидаемых и полученных параметров. Для этого для каждого блока был создан специальный юнит-класс, посылающий в блок различные верные и неверные параметры. При аномальном поведении блока или его сбое, мною анализировалось поведение, и ошибка исправлялась.
* Намеренно вносились недопустимые данные в соответствующие поля, которые могли быть неверно интерпретированы программой. Затем мною анализировалось поведение программы во время обработки недопустимых данных.
* Приложение было запущено на устройствах, работающих под управлением разных версий Windows с целью выявления особенностей работы приложения, запущенного в разных операционных системах.
* После завершения цикла разработки, программный продукт тестировался на реальных устройствах.

# Заключение

Поднимая вопрос о получении множества достоверных метеорологических данных из множества источников одновременно нельзя не сказать об актуальности разработки программного приложения, соответствующего тематике данной проблемы и реализованного в соответствие с новейшими информационными технологиями.

Целью моей работы было: создание приложения, позволяющего облегчить пользователю доступ к множеству информации о погодных данных выбранного региона прямо с сайтов, с которых обычно собиралась информация.

В рамках курсовой работы были достигнуты перечисленные ниже результаты.

1. Был проведен анализ приложений, имеющихся на в общем доступе и соответствующих тематике погодных приложений (OpenWeatherMap, Gismeteo и других).
2. Было создано приложение, реализующее следующие функции:

* Отправлять запросы на сервер с метеорологическими данными.
* Принимать данные json,xml формата и работать с ними.
* Принимать значения, введенные пользователем и после обрабатывать их.
* Принятие данных из API сайтов.
* Переводить данные с английского на русский и с русского на английский
* Выводить данные о погоде в текстовом формате: Температура, ощущаемая температура, влажность, скорость ветра, направление ветра, индекс ультрафиолетового облучения, давление воздуха и пр.

1. Приложение было протестировано и показало свою работоспособность как на стандартных эмуляторах вроде VirtualBox, так и на реальных устройствах на платформе Windows .

# Список литературы

1. [daniilgorbenko](https://habr.com/ru/users/daniilgorbenko/) [Python](https://habr.com/ru/hub/python/), [JavaScript](https://habr.com/ru/hub/javascript/), [Программирование](https://habr.com/ru/hub/programming/), [HTML](https://habr.com/ru/hub/html5/). Python & EEL. Делаем просто на Python’е и красиво на JS. [Электронный ресурс]. - <https://habr.com/ru/post/550426/>.
2. Yanovets [Разработка веб-сайтов](https://habr.com/ru/hub/webdev/), [Python](https://habr.com/ru/hub/python/), [Разработка мобильных приложений](https://habr.com/ru/hub/mobile_dev/), [API](https://habr.com/ru/hub/api/). Узнаем текущую погоду и прогноз [Электронный ресурс]. -<https://habr.com/ru/post/315264/>.
3. Лутц М. Изучаем Python, 4-е издание. – Пер. с англ. – СПб.: Символ-Плюс, 2011. – 1280с.
4. Златопольский Д.М. Основы программирования на языке Python. – М.: ДМК Пресс, 2017. – 284с.
5. Лутц М. Программирование на Python, том I, 4-е издание. – Пер. с англ. – СПб.: Символ-Плюс, 2011. – 992с.
6. Бибо Бер, Кац Иегуда jQuery. Подробное руководство по продвинутому JavaScript; Символ-плюс - М., **2017. –** 624с.
7. Дронов Владимир JavaScript и AJAX в Web-дизайне; БХВ-Петербург - М., **2015. –** 736с.
8. Дронов Владимир JavaScript. Народные советы; БХВ-Петербург - М., **2016. –** 458с.
9. Дунаев Вадим JavaScript. Самоучитель; Питер - М., **2015. –** 400с.
10. Дунаев Вадим HTML, скрипты и стили; БХВ-Петербург - М., 2015. – 816с.
11. Википедия – свободная энциклопедия [Электронный ресурс]. - <http://wikipedia.org>.