бюджетное профессиональное образовательное учреждение Вологодской области

«Череповецкий лесомеханический техникум им. В.П. Чкалова»

Специальность **09.02.07** «Информационные системы и программирование»

**ОТЧЕТ ПО ПРОИЗВОДСТВЕННОЙ ПРАКТИКЕ**

**ПП по ПМ.02 Осуществление интеграции программных модулей**

Выполнил студент 2 курса группы ИС-\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

подпись \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

место практики \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

наименование юридического лица, ФИО ИП

Период прохождения:

с «\_\_\_» \_\_\_\_\_\_\_ 2024 г.

по «\_\_\_» \_\_\_\_\_\_\_ 2024 г.

Руководитель практики от

предприятия

должность\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

подпись\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

МП

Руководитель практики от

техникума: Материкова А.А.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Оценка:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

«\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2024 года

г. Череповец

2024

Введение:

Место прохождения производственной практики – ООО «Малленом Системс»

Сроки прохождения производственного практики – с 24.11.24 по 7.12.24

Задачами данной практики являются подготовка обучающихся осознанному и углубленному изучению дисциплин, привитие им практических умений и получение первичных профессиональных навыков по выбранной специальности.

Целями производственной практики (по профилю специальности) являются:

закрепление И совершенствование общих И профессиональных

компетенций:

ОК 01. Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам;

ОК 02. Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности;

ОК 03. Планировать и реализовывать собственное профессиональное и личностное развитие, предпринимательскую деятельность B профессиональной сфере, использовать знания по финансовой грамотности в различных жизненных ситуациях;

ОК 04. Эффективно взаимодействовать и работать в коллективе и команде; ОК 05. Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации с учетом особенностей социального и культурного контекста;

ОК 06. Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение на основе традиционных общечеловеческих ценностей, в том числе с учетом гармонизации межнациональных и межрелигиозных отношений, применять -стандарты антикоррупционного поведения;

ОК 07. Содействовать сохранению окружающей среды, ресурсосбережению, применять знания об изменении климата, принципы бережливого производства, эффективно действовать в чрезвычайных ситуациях;

ОК 08. Использовать средства физической культуры для сохранения и укрепления здоровья в процессе профессиональной деятельности и поддержания необходимого уровня физической подготовленности;

ОК 09. Пользоваться профессиональной документацией на государственном и иностранном языках.

ПК 2.1. Разрабатывать требования к программным модулям на основе анализа проектной и технической документации на предмет взаимодействия компонент

ПК 2.2. Выполнять интеграцию модулей в программное обеспечение

ПК 2.3. Выполнять отладку программного модуля с использованием специализированных программных средств использованием

ПК 2.4. Осуществлять разработку тестовых наборов и тестовых сценариев для программного обеспечения

ПК 2.5. Производить инспектирование компонент программного обеспечения на предмет соответствия стандартам кодирования

* Общая характеристика предприятия (организации)

«Малленом Системс» — ведущий российский разработчик и интегратор систем компьютерного зрения и промышленной видеоаналитики, основанный в 2011 году при поддержке инвестиционной компании «Малленом». Компания, состоящая из специалистов Санкт-Петербургского политехнического университета Петра Великого, использует передовые технологии машинного обучения и глубокого обучения для создания решений в различных отраслях, включая транспорт, машиностроение, нефтегазовый сектор, металлургию, пищевую промышленность, фармацевтику, алмазодобычу и атомную энергетику. Продукция компании представлена в России, странах СНГ и ЕС. Более 10 лет компания «Малленом Системс» является официальным партнером Cognex, мирового лидера в области машинного зрения, а с 2022 года — представителем Hikrobot (Hikvision) в РФ и ЕАЭС, предлагая оборудование для машинного зрения и робототехники. Компания входит в национальный рейтинг быстрорастущих технологических компаний «ТехУспех».

* Организационная структура предприятия

1.Высшее руководство

Генеральный директор (CEO): Ответственен за стратегическое управление и общее руководство компанией.

Директор по развитию: занимается стратегическим планированием и реализацией инициатив, направленных на рост бизнеса

Главный технический директор (CTO): отвечает за технологическое развитие и инновации.

2.Отдел разработки программного обеспечения:

Руководитель отдела разработки

разработка алгоритмов компьютерного зрения, программного обеспечения для анализа видео, интеграция с оборудованием. Может быть разделён на более мелкие команды, специализирующиеся на отдельных аспектах (например, обработка изображений, машинное обучение, разработка интерфейсов).

3.Отдел тестирования и контроля качества:

проверка качества программного и/или аппаратного обеспечения.

* Инженерные службы:

Руководитель направления системной интеграции: отвечает за управление проектами, связанными с интеграцией различных IT-систем и технологий в единую инфраструктуру.

• Отдел системной интеграции: Интегрирует решения в инфраструктуру клиентов.

• Отдел технической поддержки: Обеспечивает поддержку клиентов, решает технические проблемы.

* Маркетинг и продажи

• Отдел маркетинга: Занимается анализом рынка, PR и рекламой.

• Отдел продаж: Работает с клиентами, формирует коммерческие предложения и заключает контракты.

* Финансовый и административный отдел

• Бухгалтерия: Ведет финансовую отчетность и управление бюджетом.

• HR-отдел: Занимается подбором, обучением и развитием персонала.

* Юридический отдел

• Юридический консультант: Обеспечивает правовую поддержку компании, занимается контрактами и соблюдением законодательства.

1.2 Внутренний распорядок работы предприятия, охрана труда на предприятии (организации)

В компании «Малленом Системс» действует стандартный пятидневный рабочий график (5/2) с 9:00 до 18:00. Для обеспечения безопасности и соблюдения норм охраны труда в штате имеется специалист по охране труда, который проводит вводные инструктажи при приёме на работу и стажировке, а также выдаёт пропуска для инженеров на пусконаладочные работы. Специальная оценка условий труда от 20.09.2018 подтвердила отсутствие вредных производственных факторов на рабочих местах компании.

1.3 Должностные инструкции ИТ-специалистов предприятия

Техник выполняет следующие должностные обязанности:

Выполняет работу по проведению необходимых технических расчетов;

Осуществляет наладку, настройку, регулировку и опытную проверку оборудования и систем, следит за его исправным состоянием;

Принимает участие в проведение экспериментов и испытаний;

Принимает участие в разработке программ, инструкций и другой технической документации, в изготовлении макетов, а также в испытаниях и экспериментальных работах;

Выполняет работу по сбору, обработке и накоплению исходных материалов, данных статистической отчетности, научно-технической информации;

Составляет описания проводимых работ, необходимые спецификации, диаграммы, таблицы, графики и другую техническую документацию;

Выполняет работу по оформлению плановой и отчетной документации, вносит необходимые изменения и исправления в техническую документацию в соответствии с решениями, принятыми при рассмотрении и обсуждении выполняемой работы;

Систематизирует, обрабатывает и подготавливает данные для составления отчетов о работе;

Принимает необходимые меры по использованию в работе современных технических средств.

2. Осуществление интеграции программных модулей

2.1 Разработка требования к программным модулям на основе анализа проектной и технической документации на предмет взаимодействия компонент

1. Анализ проектной документации

• Изучение архитектуры системы: Определите основные компоненты системы, их функции и взаимосвязи.

• Определение бизнес-требований: Понять, какие задачи должен решать проект и какие требования предъявляются к функциональности.

2. Анализ технической документации

• Технические спецификации: Ознакомьтесь с техническими требованиями, такими как производительность, безопасность и совместимость.

• Стандарты и протоколы: Убедитесь, что все модули будут соответствовать установленным стандартам и протоколам взаимодействия.

3. Идентификация взаимодействий между компонентами

• Определение интерфейсов: Опишите, как различные модули будут взаимодействовать друг с другом. Это может включать API, сообщения или другие методы передачи данных.

• Протоколы взаимодействия: Установите, какие протоколы будут использоваться для обмена данными между компонентами.

4. Формирование требований к модулям

• Функциональные требования: Опишите, что каждый модуль должен делать. Например, обработка данных, взаимодействие с пользователем и т.д.

• Нефункциональные требования: Укажите требования к производительности, безопасности, надежности и удобству использования.

5. Документирование требований

• Создание спецификаций: Все требования должны быть четко задокументированы в формате, удобном для команды разработки.

• Использование моделей: Рассмотрите возможность использования UML-диаграмм или других визуальных средств для представления взаимодействий между компонентами.

6. Верификация и валидация требований

• Проверка на полноту и непротиворечивость: Убедитесь, что все требования полны и не противоречат друг другу.

• Обсуждение с заинтересованными сторонами: Привлеките всех заинтересованных лиц для обсуждения и подтверждения требований.

▎7. Управление изменениями

• Процесс управления изменениями: Установите процедуры для обработки изменений в требованиях на протяжении всего жизненного цикла проекта.

# 2.2. Выполнение интеграции модулей в программное обеспечение

- Интеграция на уровне исходного кода: Сборка и компиляция кода модулей, настройка необходимых зависимостей и библиотек.

- Настройка окружения: Обеспечение необходимой инфраструктуры для работы модулей (установка серверов, баз данных и т.д.).

- Финальная сборка программы: Компиляция и связывание всех модулей в единое приложение.

- Первичная проверка работоспособности: Проведение тестов на совместимость интегрированных модулей, выявление и устранение ошибок на начальных этапах.

# 

# 2.3. Выполнение отладки программного модуля с использованием специализированных программных средств

- Использование отладчиков: Применение инструментов для анализа и диагностики работы модулей (например, GDB, Visual Studio Debugger).

- Логирование и профилирование: Настройка логирования для получения данных о работе модуля и его производительности.

- Поиск и исправление ошибок: Идентификация и устранение проблем, возникающих при выполнении модулей, включая неочевидные ошибки и утечки памяти.

- Тестирование на крайних значениях: Проверка работы модулей с нештатными входными данными и сценариями.

# 2.4. Осуществление разработки тестовых наборов и тестовых сценариев для программного обеспечения

1. Определение целей тестирования

• Выявление требований: Ознакомьтесь с функциональными и нефункциональными требованиями к программному обеспечению. Это поможет определить, что именно нужно тестировать.

• Определение типа тестирования: Решите, какие виды тестирования будут проводиться (функциональное, регрессионное, нагрузочное, пользовательское и т.д.).

2. Проектирование тестовых наборов

• Создание тестового набора: Тестовый набор — это коллекция тестов, которые проверяют определенные аспекты системы. Разделите тесты на категории, такие как:

• Функциональные тесты

• Тесты производительности

• Тесты безопасности

• Тесты совместимости

• Определение критериев успешности: Установите четкие критерии, по которым будет оцениваться успешность прохождения тестов.

3. Разработка тестовых сценариев

• Создание сценариев: Каждый тестовый сценарий должен описывать конкретный случай использования или функциональность, которую нужно проверить. Сценарий должен включать:

• Идентификатор сценария: Уникальный номер или название для удобства отслеживания.

• Описание: Краткое описание того, что будет тестироваться.

• Предусловия: Условия, которые должны быть выполнены перед запуском теста.

• Шаги выполнения: Пошаговое руководство по выполнению теста.

• Ожидаемый результат: Четкое описание того, что должно произойти, если система работает корректно.

4. Подготовка данных для тестирования

• Создание тестовых данных: Подготовьте данные, необходимые для выполнения тестов. Это могут быть как валидные, так и невалидные данные для проверки различных сценариев.

• Обеспечение конфиденциальности: Если используются реальные данные, убедитесь в соблюдении норм конфиденциальности и защиты данных.

5. Автоматизация тестирования (при необходимости)

• Выбор инструментов автоматизации: Если проект позволяет, рассмотрите возможность автоматизации повторяющихся тестов с помощью специализированных инструментов (например, Selenium, JUnit, TestNG).

• Создание автоматизированных сценариев: Напишите скрипты для автоматизации выполнения тестов, если это целесообразно.

6. Выполнение тестирования

• Запуск тестов: Выполните тесты согласно разработанным сценариям.

• Запись результатов: Зафиксируйте результаты выполнения каждого теста, включая успешные и неуспешные случаи.

7. Анализ результатов и отчетность

• Анализ результатов: Проанализируйте результаты тестирования для выявления ошибок и проблем.

• Создание отчетов: Подготовьте отчеты о проведенных тестах, включающие статистику успешности, найденные дефекты и рекомендации по их исправлению.

8. Обновление тестовой документации

• Корректировка сценариев: На основе полученных результатов обновите тестовые сценарии и наборы для учета новых требований или изменений в системе.

• Документирование процесса: Зафиксируйте весь процесс разработки и выполнения тестов для будущих ссылок.

2.5. Инспектирование компонент программного обеспечения на предмет соответствия стандартам кодирования

- Проверка стандартов кодирования: Сравнение написанного кода с установленными стандартами кодирования (PEP8 для Python).

- Проведение код-ревью: Оценка качества кода другими разработчиками для выявления потенциальных проблем и улучшения читаемости.

- Использование статического анализа: Применение инструментов для статического анализа кода, которые автоматически находят нарушения стандартов.

- Подготовка отчета о соответствии: Составление отчета о проведенных инспекциях с выводами и рекомендациями по улучшению качества кода.

# 3. Выполняемые задания

В начале производственной практики был создан GIT-репозиторий с соответствующей структурой.

Структура репозитория:

- Отчет

- Задания

- Документы

Ссылка на репозиторий:

Смирнов Руслан Викторович [https://github.com/Tokvik/-](https://github.com/Tokvik/-" \t "_blank)

Добавление отчета и документов на практику:

- Отчет- Отчет по практике

- Задания- Программный код

- Документы- Данные практики

В ходе производственной практики, выполненной по заказу ООО «Малленом Системс», были разработаны два модуля — обработки изображений и взаимодействия с пользователем, — а также техническое задание. Настоящий отчёт описывает процесс разработки, включая проектирование, реализацию и тестирование этих модулей.

Техническое задание на разработку модулей.

Работу выполнил, практикант группы ИС-23

Смирнов Руслан

Череповец 2024

Оглавление

1. Введение 3
2. Основания для разработки 4
3. Назначение разработки 5
4. Требования к программе или программному изделию 6
5. Требования к программной документации 7
6. Технико-экономические показатели 8
7. Стадии и этапы разработки 9
8. Порядок контроля и приемки 10
9. Заключение 11

Введение

Данное техническое задание (ТЗ) определяет требования к разработке двух модулей: модуля обработки и работы с изображениями и модуля взаимодействия с пользователем и формирования данных. Модули будут использоваться для обработки изображений, получения информации о них и взаимодействия с пользователем.

Основания для разработки

Разработка данных модулей обусловлена необходимостью автоматизации процесса обработки изображений, улучшения взаимодействия пользователей с системой и упрощения получения информации о файлах изображений.

Назначение разработки

Цель разработки состоит в создании программного решения, которое позволит пользователям:

• Получать информацию о изображениях (размер, разрешение, дата создания).

• Переименовывать изображения.

• Удобно взаимодействовать с системой для передачи данных о изображениях.

Требования к программе или программному изделию

А)Модуль обработки и работы с изображениями

• Выдача информации о изображении:

• Размер в байтах.  
  
• Разрешение (ширина и высота в пикселях).  
  
• Дата создания файла.

• Переименование изображения по заданному пользователем шаблону.

• Обработка изображений должна выполняться в течение не более чем 2 секунд для файлов размером до 10 МБ.

Б) Модуль взаимодействия с пользователем

• Функциональные требования:

• Пользователь должен иметь возможность задать путь к изображению через графический интерфейс или командную строку.

• Модуль должен передавать путь к изображению в модуль обработки и получать результаты.

Требования к программной документации

• Документация должна включать:

• Описание архитектуры системы.

• Руководство пользователя с примерами использования.

• Техническую документацию для разработчиков, включая описание API модулей.

Технико-экономические показатели

• Ожидаемая стоимость разработки: [указать сумму].

• Сроки разработки: [указать сроки].

• Ожидаемая экономия времени на обработку изображений: [указать процент].

Стадии и этапы разработки

1. Анализ требований.

2. Проектирование модулей.

3. Разработка модуля обработки изображений.

4. Разработка модуля взаимодействия с пользователем.

5. Тестирование и отладка модулей.

6. Документация и подготовка к внедрению.

Порядок контроля и приемки

• Контроль за выполнением этапов разработки будет осуществляться на каждом этапе через промежуточные отчеты.

• Приемка будет проводиться по завершении тестирования, при условии выполнения всех функциональных требований.

Заключение

Разработка модулей обработки изображений и взаимодействия с пользователем является важным шагом к автоматизации и оптимизации работы с графическими данными. В соответствии с представленным техническим заданием, предполагается создание двух взаимосвязанных модулей, которые обеспечат пользователям удобный и эффективный интерфейс для работы с изображениями.

1 Модуль обработки и работы с изображениями.

Пояснения к коду:

1. Импорт необходимых библиотек:

• PIL (Pillow) для работы с изображениями.

• os для работы с файловой системой.

• datetime для получения даты создания файла.

2. Класс ImageProcessor:

• Метод \_\_init\_\_: Инициализирует объект с путем к изображению.

• Метод load\_image: Загружает изображение и обрабатывает возможные ошибки.

• Метод get\_image\_info: Получает информацию об изображении (размер, режим, дата создания и разрешение).

• Метод rename\_image: Переименовывает файл изображения и обновляет путь.

3. Пример использования:

• Запрашивает у пользователя путь к изображению.

• Загружает изображение и выводит его информацию.

• Запрашивает новое имя для изображения и переименовывает его.

2 Модуль взаимодействия с пользователем и формирование и хранений данных.

Как использовать:

Установите Pillow: pip install Pillow

Запустите скрипт: Он запросит путь к изображению.

Введите путь: Введите полный путь к вашему изображению.

Результат: Будет выведен словарь с информацией об изображении (путь, формат, ширина, высота, цветовой режим).

Улучшения:

Обработка ошибок: Код включает обработку FileNotFoundError, предотвращая аварийную остановку при ошибках.

Более информативный вывод: Вывод теперь включает цветовой режим (RGB, L и т.д.), делая информацию более полезной.

Закрытие файла: Код закрывает объект Image с помощью img.close(). Это важно для освобождения ресурсов и предотвращения проблем с файлом.

Возврат информации: Функция get\_image\_info возвращает словарь с данными. Это гораздо более удобный способ передавать данные о изображении в программу.

Встроенная проверка: Функция get\_image\_path\_from\_user() теперь проверяет, существует ли файл перед тем, как продолжить.

PEP8:

PEP 8 — это руководство по стилю кода для языка программирования Python, которое описывает рекомендации по написанию кода, чтобы он был читаемым и поддерживаемым.

1. Общие рекомендации

• Читаемость: Код должен быть легким для чтения и понимания.

• Именование: Используйте понятные и описательные имена для переменных, функций и классов.

2. Отступы

• Используйте 4 пробела для каждого уровня отступа.

• Не используйте табуляцию; используйте только пробелы.

3. Максимальная длина строки

• Ограничьте длину строки 79 символами.

• Для комментариев и документации используйте максимальную длину 72 символа.

4. Пустые строки

• Используйте пустые строки для разделения функций и классов.

• Внутри функций можно использовать пустые строки для разделения логических блоков кода.

5. Импорт модулей

• Импортируйте модули в следующем порядке: стандартные библиотеки, сторонние библиотеки, локальные модули.

• Каждый импорт должен быть на отдельной строке.

6. Именование

• Переменные и функции: используйте стиль lowercase\_with\_underscores.

• Классы: используйте стиль CapitalizedWords.

• Константы: используйте стиль ALL\_CAPS\_WITH\_UNDERSCORES.

7. Пробелы в выражениях и инструкциях

• Используйте пробелы вокруг операторов и после запятых.

• Не добавляйте пробелы перед запятой или скобками.

Провести инспектирование разрабатываемого программного кода:

1. Отступы: Обратите внимание на правильное использование отступов (4 пробела).

2. Комментарии: Для более ясного выделения комментариев используйте пробелы.

3. Импорт: Импортируйте модули в первой части файла.

4. Проверка имени: Убедитесь, что используете двойные подчеркивания в

if \_\_name\_\_ == "\_\_main\_\_":.

Интеграция модулей:

При интеграции двух модулей был добавлен главный модуль def main(рис.3)

Отладка:

При отладке проблем не было обнаружено.

тестирование программных модулей:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Наименование  теста | Тестовые  данные | Ожидаемый  результат | Фактический  результат | Результат  тестирования | Комментарий |
| Test1 | Ширина: 961  Длина: 686 | Информация об изображении:  Размер: (961, 686)  Режим: RGB  Дата создания: 2024-11-20 22:52:18  Разрешение: (94, 94)  меняет название изображения | Информация об изображении:  Размер: (961, 686)  Режим: RGB  Дата создания: 2024-11-20 22:52:18  Разрешение: (94, 94)  меняет название изображения | Код работает отлично, без ошибок, но принимает только .Jpg | - |
| Test2 | Ширина: 961  Длина:  686 | Получает и отображает информацию об изображении.  Запрашивает путь к изображению у пользователяю  Выводим словарь с информацией | Получает и отображает информацию об изображении.  Запрашивает путь к изображению у пользователяю  Выводим словарь с информацией | Код работает отлично. | - |

Заключение:

Производственная практика позволила мне расширить свои знания Python, результатом чего стало создание двух взаимодействующих модулей: обработки изображений и взаимодействия с пользователем

# 

# Список использованных источников

1. работа с Pillow - <https://pythonru.com/biblioteki/osnovnye-vozmozhnosti-biblioteki-python-imaging-library-pillow-pil>

2. Os - <https://docs.python.org/3/library/os.html>

3. Работа с модулями - <https://metanit.com/python/tutorial/2.10.php>

4. ГОСТ 7.80-2000 «Библиографическая запись. Заголовок. Общие

требования и правила составления» [Электронный ресурс]/ Электронный

правовой и нормативно-технической документации- режим доступа: <http://docs.cntd.ru/document/gost-7-80-2000>

5. ГОСТ 7.32-2001 «Отчет о научно-исследовательской работе. Структура и правила оформления» [Электронный ресурс]/ Электронный правовой и нормативно-технической документации- режим доступа:

<http://docs.cntd.ru/document/gost-7-32-2001-sibid>

6. ГОСТ 7.1-2003 № 332-ст «Система стандартов по информации, библиотечному и издательскому делу. Библиографическая запись. Библиографическое описание. Общие требования и правила составления»

[Электронный ресурс]/ Электронный правовой и нормативно-технической

документации- режим доступа: <http://docs.cntd.ru/document/gost-7-1-2003-sibid>

7. ГОСТ Р 7.0.5-2008 «Система стандартов по информации, библиотечному и издательскому делу. Библиографическая ссылка. Общие требования и правила составления» [Электронный ресурс]/ Библиотека ГОСТов стандартов и нормативов- режим доступа: <http://www.standartov.ru/norma_doc/53/53649/index.htm>

8. ГОСТ Р 7.0.11-2011 «Система стандартов по информации, библиотечному и издательскому делу. Диссертация и автореферат диссертации. Структура и правила оформления» [Электронный ресурс]/ Электронный правовой и нормативно-технической документации- режим доступа: <http://docs.cntd.ru/document/gost-r-7-0-11-2011>

9. Официальный сайт Microsoft Office [Электронный ресурс]- режим

доступа: <https://products.office.com/ru-RU/>

10. Сайт компании нанимателя- <https://www.mallenom.ru/>

11. PEP8 - <https://peps.python.org/pep-0008/>

12. Отладка - <https://habr.com/ru/companies/vk/articles/205426/>

Приложения:





Рис3

