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

Период прохождения:

с «\_\_\_» \_\_\_\_\_\_\_ 2024 г.

по «\_\_\_» \_\_\_\_\_\_\_ 2024 г.

Руководитель практики от

предприятия

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

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

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

Руководитель практики от

техникума: Материкова А.А.

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

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

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

г. Череповец

2024

**Содержание**

Введение...................................................................................................................3

Общая характеристика предприятия.....................................................................4

Осуществление интеграции программных модулей.........................................5-6

Выполняемые задания........................................................................................7-17

Заключение.............................................................................................................18

Используемые источники......................................................................................19

**Введение**

Место прохождения производственной практики – ООО “Малленом Системс”

Сроки прохождения производственного практики – с 10.11.24 по 23.11.24

Основная цель производственной практики заключается в повышении уровня подготовки студента в выбранной сфере, знакомстве с особенностями профессии в реальных условиях, а также в приобретении и закреплении практических навыков и развитии профессиональных качеств.

Задачи производственной практики:

1. Углубление теоретических знаний, полученных в процессе обучения

2. Приобретение практического опыта в рамках выбранной специальности

3. Ознакомление с особенностями работы и внутренними процессами профильных предприятий

4. Освоение должностных обязанностей, соответствующих специальности

5. Получение знаний о нормативных требованиях в основных направлениях производственной деятельности

1. **Общая характеристика предприятия**
   1. Организационная структура предприятия

Генеральный директор – Живиця Анна Эдуардовна

Учредители – 10 физ. лиц

Сотрудники – более 100 физ. Лиц

"Маленном Системс" представляет собой современную IT-компанию, специализирующуюся на разработке программного обеспечения и предоставлении услуг в области информационных технологий.

1.2 Внутренний распорядок работы предприятия, охрана труда на предприятии

Внутренний распорядок работы в "Маленном Системс" основан на принципах гибкого графика, что позволяет сотрудникам эффективно организовывать свое время. Рабочий день начинается в 9:00 и заканчивается в 18:00.

Охрана труда в компании является приоритетом. "Маленном Системс" обеспечивает соблюдение всех необходимых норм и стандартов безопасности. Компания предоставляет необходимые средства индивидуальной защиты и организует комфортные условия труда.

1.3 Должностные инструкции ИТ-специалистов предприятия

Должностные инструкции ИТ-специалистов в "Маленном Системс" четко регламентируют их обязанности и ответственность. Основные должности включают:

• Разработчик: отвечает за создание и тестирование программного обеспечения, участие в проектировании архитектуры систем.

• Системный администратор: обеспечивает бесперебойную работу серверов и сетевой инфраструктуры, управляет доступом пользователей к ресурсам.

• Технический специалист: занимается поддержкой пользователей, решением технических проблем и обучением сотрудников работе с ПО.

• Аналитик: анализирует бизнес-процессы, разрабатывает технические задания и участвует в проектировании новых решений.

1. **Осуществление интеграции программных модулей**
   1. Разработка требования к программным модулям на основе анализа проектной и технической документации на предмет взаимодействия компонентов

На этом этапе необходимо:

• Изучить проектную и техническую документацию: Внимательно ознакомиться с описанием функциональности каждого модуля, их назначением и ожидаемым поведением.

• Определить требования к интерфейсам: Установить, какие данные будут передаваться между модулями, в каком формате и с какой частотой.

• Выявить зависимости: Определить, какие модули зависят друг от друга и как изменения в одном модуле могут повлиять на другие.

• Сформулировать требования к производительности: Установить допустимые задержки и объемы передаваемых данных для обеспечения эффективного взаимодействия.

* 1. Выполнение интеграции модулей в программное обеспечение

Процесс интеграции включает:

• Объединение модулей: Настройка окружения для тестирования и интеграции различных модулей

• Конфигурация интерфейсов: Настройка всех необходимых параметров для обеспечения корректной работы интерфейсов между модулями.

• Тестирование интеграции: Проведение первичных тестов для проверки, что модули работают совместно без ошибок.

* 1. Выполнение отладки программного модуля с использованием специализированных программных средств

На этом этапе необходимо:

• Использовать отладчики: Применять инструменты отладки для пошагового анализа выполнения кода и выявления ошибок.

• Анализ логов: Изучать логи выполнения программы для поиска аномалий и проблем в работе модулей.

• Тестирование отдельных функций: Проверять каждую функцию или метод на наличие ошибок и соответствие требованиям.

* 1. Осуществление разработки тестовых наборов и тестовых сценариев для программного обеспечения

Разработка тестов включает:

• Определение целей тестирования: Установить, что именно будет проверяться (функциональность, производительность).

• Создание тестовых наборов: Формировать наборы тестов, которые охватывают все основные функции и сценарии использования.

• Автоматизация тестирования: Если возможно, реализовать автоматизированные тесты для повышения эффективности процесса.

* 1. Инспектирование компонент программного обеспечения на предмет соответствия стандартам кодирования

Этот процесс включает:

• Проверка на соответствие стандартам: Использование инструментов статического анализа для проверки кода на соответствие установленным стандартам кодирования.

• Документирование результатов инспекции: Фиксация выявленных проблем и рекомендаций по их устранению для дальнейшего улучшения качества кода.

1. **Выполняемые задания**

Задания выполняются на платформе GitHub, которая представляет собой многофункциональный инструмент для совместной работы над проектами. GitHub включает в себя:

1. Облачное хранилище: Хранение всех файлов проекта в облаке.

2. Социальная сеть для разработчиков: Возможности для взаимодействия и обмена идеями.

3. Система контроля версий: Отслеживание всех изменений в коде.

Техническое задание для разрабатываемых модулей можно найти в приложении 1.

Модули:



Модуль 1 - Изменение цветовой гаммы изображения (в черно-белое)

Этот модуль меняет цветовую гамму на черно-белую, сохраняя в ту же директорию.



Модуль 2 - Перемещение изображения в другие папки

Этот модуль перемещает файл из одной директории в другую.



Модуль 3 – Работа с пользователем

Этот модуль спрашивает, что нужно сделать, затем от выбора зависят дальнейшие действия:

1. Модуль запрашивает путь до файла, затем сохраняет измененный файл в ту же директорию
2. Модуль запрашивает путь до файла, после спрашивает путь куда его сохранить и сохраняет его в выбранную директорию

Описать стандарт кодирования на языке Python, используя стандарт PEP 8.

PEP 8 — это стиль кодирования для языка Python, который предлагает рекомендации по написанию чистого и читаемого кода. Вот основные принципы и рекомендации, изложенные в PEP 8:

1. Отступы

• Используйте 4 пробела на уровень отступа.

• Никогда не используйте табуляцию (tab) для отступов.

2. Максимальная длина строки

• Ограничьте длину строки 79 символами для кода и 72 символами для комментариев и строк документации (docstrings).

3. Пустые строки

• Используйте пустые строки для разделения функций и классов.

• Внутри функций используйте одну пустую строку для разделения логических блоков кода.

4. Импорт

• Импортируйте стандартные библиотеки, сторонние библиотеки и локальные модули в следующем порядке, разделяя их пустыми строками.

• Используйте абсолютные импорты, а не относительные.

5. Именование

• Используйте snake\_case для именования функций и переменных.

• Используйте CamelCase для именования классов.

• Константы должны быть написаны в верхнем регистре с подчеркиваниями (UPPER\_CASE).

6. Пробелы

• Не ставьте пробелы перед запятыми, точками с запятой и двоеточиями.

• Ставьте пробелы после запятых, точек с запятой и двоеточий.

• Не ставьте пробелы внутри скобок.

7. Комментарии

• Комментарии должны быть ясными и краткими. Используйте полные предложения.

• Однострочные комментарии начинаются с заглавной буквы и заканчиваются точкой.

• Многострочные комментарии могут быть написаны с использованием # или строк документации.

8. Строки документации (docstrings)

• Используйте строки документации для описания модулей, классов и функций.

• Первое слово должно быть в заглавной букве, а строка должна заканчиваться точкой.

9. Выражения и операторы

• Используйте явные скобки для повышения читаемости сложных выражений.

• Старайтесь избегать использования сложных выражений в одной строке.

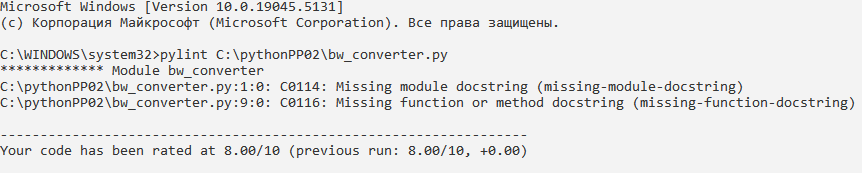
10. Прочие рекомендации

• Избегайте использования import \*, чтобы не загрязнять пространство имен.

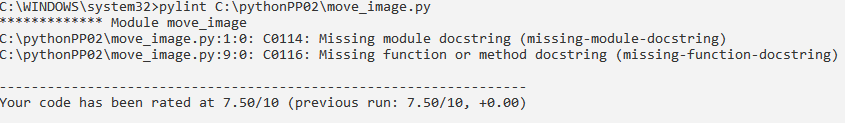
• Используйте is для сравнения с None, а не ==.

Инспектирование разрабатываемого кода

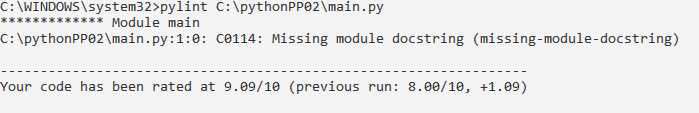
Для инспектирования разрабатываемого кода использовался инструмент pylint.



Результат тестирования модуля: изменение цветовой гаммы изображения (в черно-белое)



Результат тестирования модуля: перемещение изображения в другие папки



Результат тестирования модуля: Работа с пользователем

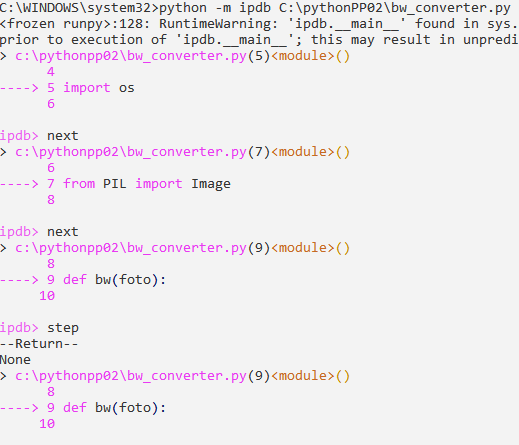
Интеграция модулей

Чтобы объединить модули, я сделал 3 модуль «Модуль работы с пользователем», он включает в себя функции 2 модулей: «изменение цветовой гаммы изображения (в черно-белое) и перемещение изображения в другие папки», этот модуль спрашивает у пользователя, что нужно сделать: «изменить цветовую гамму на черно-белую или переместить файл в другую директорию», затем после выбора, модуль запрашивает нужные данные и передает их модулям для работы с изображениями

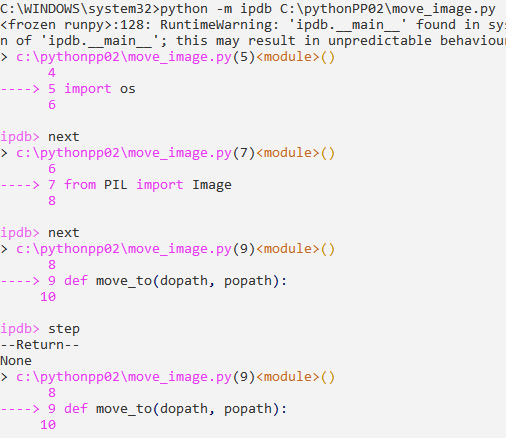
Отладка модулей

Отладка модулей проходила в командной строке с помощью модуля ipdb.

Отладка модуля «Изменение цветовой гаммы изображения (в черно-белое)»



Отладка модуля «перемещение изображения в другие папки»



Отладка модуля «Работа с пользователем»



Тестирование программных модулей

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Наименование теста | Тестовые данные | Ожидаемый результат | Фактический результат | Результат тестирования |
| 1. Тест модуля  «Работа с пользователем» | Подготовить цветное изображение для тестирования.  Выбрать опцию 1 | Изображение должно быть конвертировано в черно-белый формат и сохранено в указанном месте | Изображение конвертировалось в черно-белый формат | Изображение конвертировалось  В черно-белый формат и сохранилось в ту же директорию. |
| 2. Тест модуля  «Работа с пользователем» | Подготовить изображение для тестирования. Выбрать опцию 2 (переместить изображение) | Изображение должно быть перемещено в указанную папку (проверить наличие файла в новой папке и отсутствие в старой). | Изображение переместилось в указанную папку и пропало из старой | Изображение переместилось в указанную папку и пропало из старой |

**Заключение**

Во время производственной практики была реализована задача по разработке и интеграции модулей для обработки изображений и взаимодействия с пользователем. В ходе работы были использованы стандарты кодирования PEP 8, что способствовало созданию понятного и поддерживаемого кода.

В результате проведенной работы были созданы 3 модуля:

1. Модуль изменение цветовой гаммы изображения (в черно-белое): меняет цветовую гамму на черно-белую, сохраняя в ту же директорию.

2. Модуль перемещение изображения в другие папки: этот модуль перемещает файл из одной директории в другую.

3. Модуль Работа с пользователем: этот модуль запрашивает нужные данные у пользователя и модуль передает их модулям для работы с изображениями.

Практика продемонстрировала значимость соблюдения стандартов кодирования, а также применения инструментов для отладки и тестирования. Приобретенные навыки и опыт окажутся полезными в будущем и помогут в создании более сложных и надежных программных решений.

**Литература**

[Компания «Малленом Системс»: Системы технического зрения | Системы распознавания номеров | Системы моделирования](https://www.mallenom.ru/?ysclid=m3qmgtgaf9200667568)

[Добро пожаловать в Python.org](https://www.python.org/)

[Работаем с Git: первые шаги в GitHub / Хабр](https://habr.com/ru/companies/yandex_praktikum/articles/700708/)

[Инспектирование кода: лучшая практика](https://tproger.ru/translations/code-review-best-practices)

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

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

<https://metanit.com/python/tutorial/2.10.php>