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**

Место прохождения производственной практики – ООО “Малленом Системс”

Сроки прохождения производственного практики – с 10.11.24 по 23.11.24

Цель производственной практики — улучшить подготовку студента в выбранной области, познакомить его с тонкостями профессии на реальном производстве, дать возможность приобрести и закрепить практические навыки, а также развить профессиональные качества и лучше понять свою специальность.

Задачи производственной практики:

* Ознакомится с деятельностью предприятия
* Укрепить теоретические знания, полученные во время учёбы
* Получить практический опыт по специальности
* Выполнить задание предоставленное руководителем практики от предприятия
* Подготовить отчет по прохождению производственной практики

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

Генеральный директор – Живиця Анна Эдуардовна

Учредители – 10 физ. лиц

Сотрудники – более 100 физ. лиц

* 1. Внутренний распорядок работы предприятия, охрана труда на предприятии

**Режим работы**

С понедельника по пятницу 9:00-18:00

**Охрана труда**

На предприятии созданы все условия для безопасного труда сотрудников. Проводится регулярное обучение по охране труда, используются современные средства индивидуальной защиты, соблюдаются требования пожарной безопасности.

* 1. Должностные инструкции ИТ-специалистов предприятия

Должностные инструкции ИТ-специалистов «Маленное Системс» определяют их обязанности, права и ответственность. В зависимости от должности, специалисты могут заниматься разработкой программного обеспечения, тестированием, технической поддержкой, администрированием серверов и баз данных, а также другими задачами.

1. **Осуществление интеграции программных модулей**

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

* Изучить документацию, чтобы понять требования к программному обеспечению и взаимодействие его компонентов.
* Определить, какие функции должны быть реализованы в программе: изменение расширения файла и перемещение изображения в другие папки.
* Сформулировать требования к каждому модулю на основе анализа документации.
* Учесть взаимодействие компонентов и обеспечить совместимость модулей.
* Согласовать требования с заинтересованными сторонами.

2.2 Выполнение интеграции модулей в программное обеспечение

* Создать проект для интеграции модулей.
* Реализовать интерфейсы взаимодействия между модулями.
* Протестировать интеграцию модулей на соответствие требованиям.

2.3 Выполнение отладки программного модуля с использованием специализированных программных средств

* Использовать специализированные инструменты отладки для поиска и исправления ошибок в модулях.
* Провести тестирование модулей на различных сценариях использования.
* Отладить взаимодействие между модулями для обеспечения корректной работы программного обеспечения.

2.4 Осуществление разработки тестовых наборов и тестовых сценариев для программного обеспечения

* Разработать тестовые наборы для проверки функциональности и надёжности программного обеспечения.
* Сформулировать тестовые сценарии для каждого модуля и интеграции модулей.

2.5 Инспектирование компонентов программного обеспечения на предмет соответствия стандартам кодирования

* Проверить код модулей на соответствие стандартам кодирования.
* Убедиться в соблюдении стиля кодирования и читаемости кода.
* Исправить обнаруженные нарушения стандартов кодирования.

**3. Выполнение задания**

Для выполнения задания анализируем требования к разрабатываемым модулям и составляем техническое задание (см. приложение 1). Продумываем архитектуру модулей принцип их взаимодействия. Прежде чем приступить к написанию модулей, стоит вспомнить про стандарт кодирования PEP8.

PEP 8 — это руководство по стилю написания кода на Python, разработанное Гвидо ван Россумом и его командой. Оно содержит рекомендации по форматированию кода, чтобы сделать его более читаемым и понятным для других разработчиков.

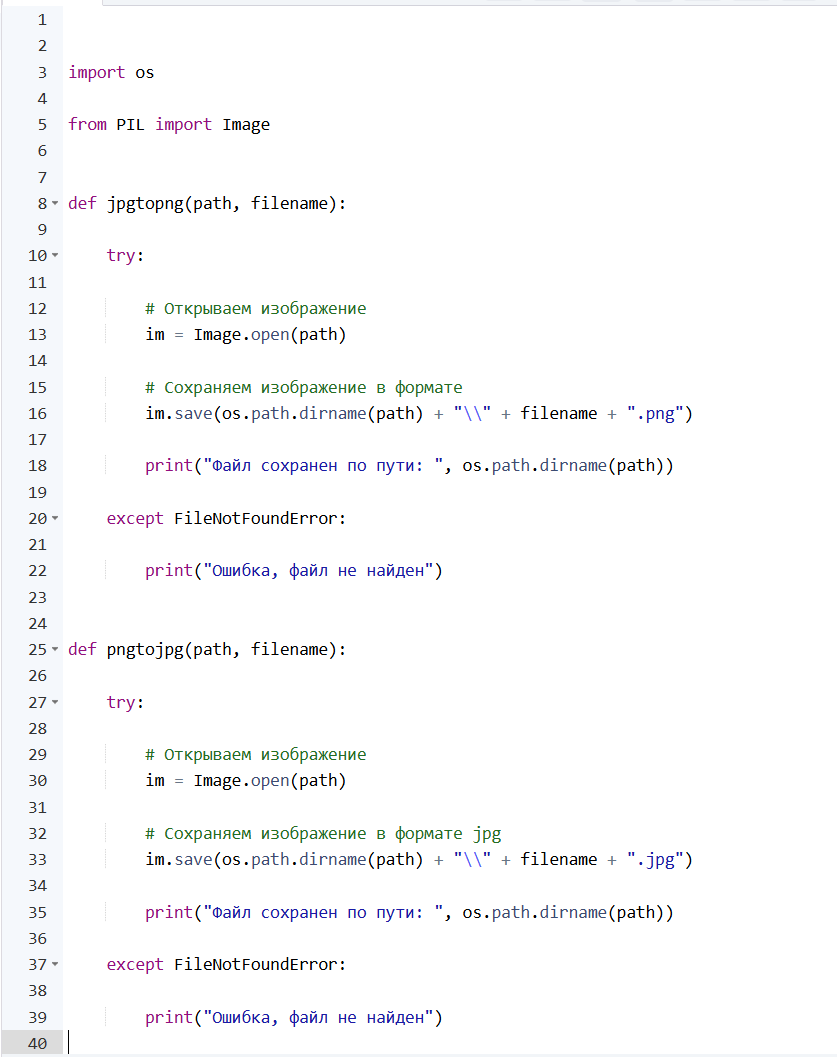
Основные принципы PEP 8:

1. Отступы. Используйте отступы в 4 пробела для блоков кода. Это помогает визуально выделить блоки кода и облегчает чтение кода.
2. Длина строк. Ограничьте длину строк до 79 символов. Это позволяет разместить код на одном экране и упрощает навигацию по коду.
3. Имена переменных. Используйте понятные имена переменных, которые отражают их назначение. Избегайте использования слишком длинных или сложных имён.
4. Комментарии. Пишите комментарии, чтобы объяснить сложные участки кода или предоставить дополнительную информацию. Используйте комментарии только там, где это необходимо.
5. Функции и методы. Определяйте функции и методы с короткими именами, которые описывают их действие. Избегайте длинных и сложных имен.
6. Импорт. Импортируйте модули и библиотеки с использованием синтаксиса import или from ... import. Избегайте импорта большого количества модулей в начале файла.
7. Документация. Предоставляйте документацию для функций, методов и классов с помощью комментариев и докстрингов.
8. Тестирование. Разрабатывайте тесты для проверки функциональности кода. Используйте фреймворки для тестирования, такие как unittest или pytest.
9. Согласованность. Соблюдайте единый стиль форматирования кода во всём проекте. Это облегчает понимание кода другими разработчиками.
10. Читаемость. Код должен быть легко читаемым и понимаемым. Используйте простые конструкции и избегайте сложных и запутанных выражений.

Для удобной работы с файлам, контроля версий модулей используем репозиторий Github.

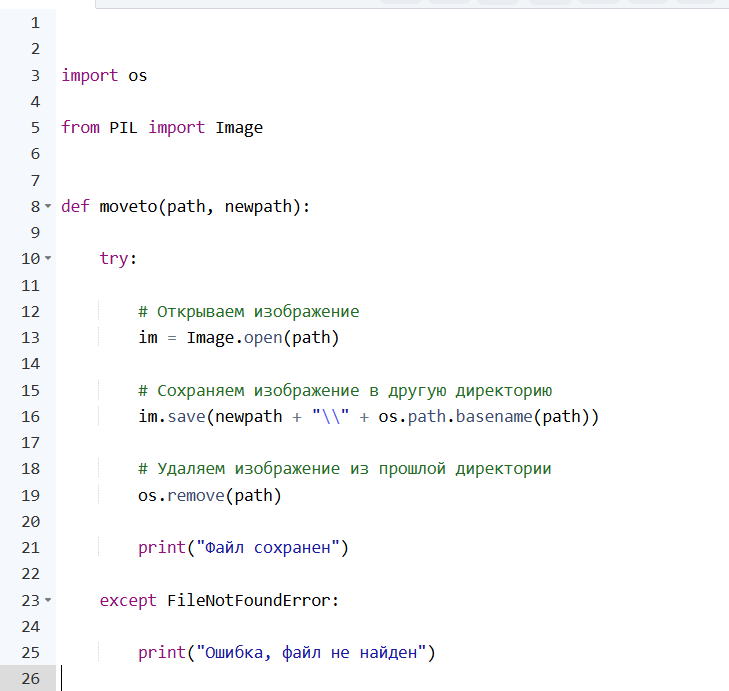
Приступаем к написанию модулей. Первый модуль - это модуль конвертации изображения. Он представляет из себя две функции: конвертация изображения из jpg в png и из png в jpg. Функции принимают путь до изображения и новое название конвертированного изображения. Выбранное изображение конвертируется и сохраняется в ту же папку с новым именем.

Программный код модуля конвертации (Рисунок 1)



Следующий модуль - это модуль перемещения изображения в другую папку. Модуль состоит из функции, которая берет выбранное пользователем изображение, сохраняет его в новую директорию и удаляет из предыдущей.

Программный код модуля перемещения изображений в другие папки (Рисунок 2)



Для объединения модулей нам потребуется еще один модуль. Этот же модуль будет отвечать за взаимодействие с пользователем. Данный модуль спрашивает у пользователя что нужно сделать: конвертировать изображение или переместить изображение. После выбора действия модуль запрашивает у пользователя нужные данные и передает их модулям для работы с изображениями. Также модуль определяет тип конвертации и

Программный код модуля для работы с пользователем и интеграции модулей для работы с изображениями (Рисунок 3)



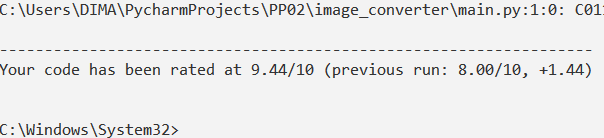
Во время написания программного кода производим отладку с помощью Python Debugger в командной строке, для устранения ошибок.

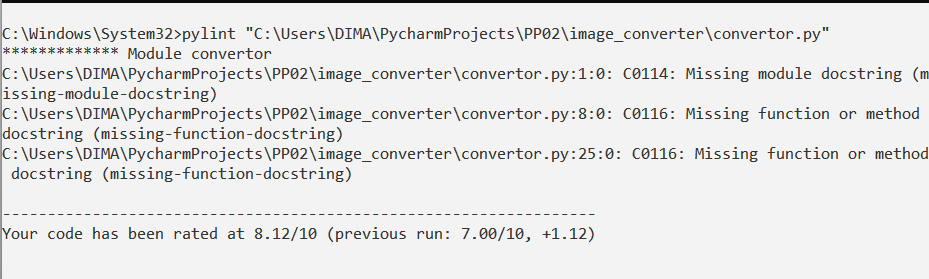
(Рисунок 4)

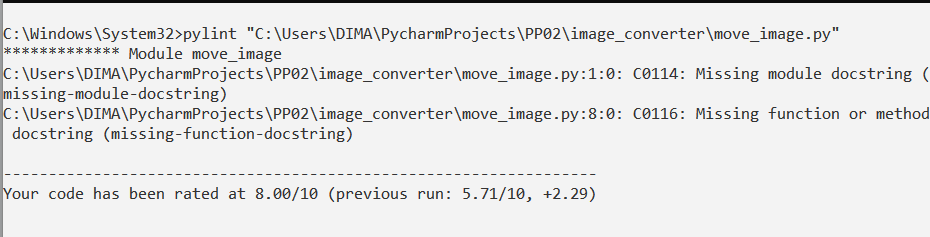


После отладки кода проводим инспектирование каждого модуля на соответствие стандартам кодирования PEP8 с помощью pylint

(Рисунок 5), (Рисунок 6), (Рисунок 7)







После инспектирования разрабатываемого программного кода нужно провести тестирование нужно провести тестирование программных модулей, чтобы убедиться в корректности работы программы. Для тестирования используем тестовые сценарии

| Наименование теста | Тестовые данные | Ожидаемый результат | Фактический результат | Результат тестирования |
| --- | --- | --- | --- | --- |
| Тестирование модуля конвертации изображения | Изображение Лес.png, новое имя файла Лес\_конв | Конвертированное изображение Лес\_конв.jpg | Конвертированное изображение Лес\_конв.jpg | Изображение конвертировано и сохранено в ту же директорию с новым именем |
| Тестирование модуля перемещения изображения | Изображение  Лес.png в директории C:\Photo, директория для перемещения C:\Image | Изображение Лес.png из директории C:\Photo переместится в директорию C:\Image | Изображение Лес.png из директории C:\Photo перемещено в директорию C:\Image | Изображение из выбранной директории перемещено в другую директорию |
| Тестирование модуля для работы с пользователем №1 | Выберем, путем ввода в консоль “1” конвертирование изображения | Модуль запросить у пользователя ввести в консоль путь до изображения и новое имя файла. После того как пользователь введет нужную информацию, модуль передаст ее в модуль для конвертации изображения | Модуль запрашивает у пользователя через консоль путь до изображения и новое имя файла. После того как пользователь введет нужную информацию, модуль передает ее в модуль для конвертации изображения | Информация полученная от пользователя передана модулю для конвертации |
| Тестирование модуля для работы с пользователем №2 | Выберем, путем ввода в консоль “2” перемещение изображения в другую папку | Модуль запросить у пользователя ввести в консоль путь до изображения и путь, по которому изображение должно будет находится после перемещения. После того как пользователь введет нужную информацию, модуль передаст ее в модуль для перемещения изображения | Модуль запрашивает у пользователя ввести в консоль путь до изображения и путь, по которому изображение должно будет находится после перемещения. После того как пользователь введет нужную информацию, модуль передает ее в модуль для перемещения изображения | Информация полученная от пользователя передана модулю для перемещения изображений |

После проверки всех модулей скомпилируем модули в .EXE файл, чтобы программу могли пользоваться пользователи без установленных библиотек и самого пайтон на компьютере. Для этого используем pyinstaller. После компиляции получаем готовый .EXE файл, который можем передать пользователям.

**Список литературы**

Стандарт PEP 8 - https://peps.python.org/pep-0008/

Библиотека pillow - <https://python-scripts.com/pillow>

Библиотека os - <https://pythonworld.ru/moduli/modul-os.html>

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

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

Компиляция - <https://pythonru.com/biblioteki/pyinstaller>

**Приложение**

Приложение 1

Техническое задание на разработку модуля обработки и работы с изображениями, а также модуля взаимодействия с пользователем и формирования и хранения данных.

**Цель разработки:**

Программу, которая позволит пользователям легко конвертировать изображения из одного формата в другой (например, PNG в JPG или наоборот), а также перемещать изображения между папками.

**Модуль обработки и работы с изображениями:**

Функционал: Модуль должен предоставлять возможность конвертации изображений из одного формата (PNG, JPG) в другой. Также должна быть предусмотрена функция перемещения изображений между различными папками.

Требования к модулю:

Поддержка различных форматов изображений.

Автоматический выбор типа конвертации (из PNG в JPG, из JPG в PNG).

Передача данных модулю работы с изображением от модуля взаимодействия с пользователем.

Генерация результата конвертации или перемещения.

**Модуль взаимодействия с пользователем:**

Функциональность: Модуль обеспечивает взаимодействие пользователя с системой. Пользователь может задать путь к изображению, выбрать тип конвертации и указать папку для перемещения файла.

**Требования:**

Удобный интерфейс для ввода пути к файлу изображения.

Автоматический выбор типа конвертации.

Указание папки для перемещения изображения.

**Системные требования:**

Операционная система Windows

**Технические требования:**

Использование языка программирования python, библиотеки pillow.

Соблюдение стандартов кодирования и проектирования.

**Этапы разработки:**

Проектирование архитектуры системы.

Разработка модулей обработки и взаимодействия.

Тестирование системы на соответствие требованиям.

Доработка системы по результатам тестирования.

Сроки выполнения:

Разработка программы должна быть завершена в срок производственной практики.