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.03 РЕВЬЮИРОВАНИЕ ПРОГРАММНЫХ МОДУЛЕЙ**

Выполнил студент 3 курса группы ИС-31

Авдонина Елизавета Максимовна

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

место практики: ООО «Малленом Системс»

Период прохождения:

с «08» декабря 2024 года

по «21» декабря 2024 года

Руководитель практики от

предприятия, должность:

Южакова Н.В., специалист по кадрам

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

МПРуководитель практики от

техникума: Материкова А.А.

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

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

СОДЕРЖАНИЕ

[Введение 3](#_Toc185338289)

[1 Общая характеристика компании 4](#_Toc185338290)

[1.1 Организационная структура компании 4](#_Toc185338291)

[1.2 Внутренний распорядок работы компании и охрана труда 5](#_Toc185338292)

[1.3 Должностные инструкции ИТ-специалиста компании 5](#_Toc185338293)

[2 Ревьюирование программных продуктов 7](#_Toc185338294)

[2.1 Ревьюирование программного кода в соответствии с технической документацией 7](#_Toc185338295)

[2.2 Измерение характеристик компонентов программного продукта 7](#_Toc185338296)

[2.3 Исследование созданного программного кода с использованием специализированных программных средств 8](#_Toc185338297)

[2.4 Сравнительный анализ программных продуктов и средств разработки 8](#_Toc185338298)

[3 Выполняемые задания 10](#_Toc185338299)

[Заключение 19](#_Toc185338300)

[Список используемых источников 20](#_Toc185338301)

# Введение

Производственная практика является важным компонентом образовательного процесса, который способствует качественной подготовке специалистов и их успешной интеграции в профессиональное сообщество. Она помогает студентам познакомиться с культурой работы в компании, включая понимание внутренней структуры, процедур и этики. Практика проходила в ООО «Малленом Системс» с 08.12.2024 по 21.12.2024.

Целью практики было освоение основного вида деятельности по направлению «Ревьюирование программных модулей» и формирование соответствующих общих и профессиональных компетенций.

В рамках выполнения поставленной цели, были сформулированы следующие задачи:

1. Осуществить ревьюирование программного кода в соответствии с технической документацией;
2. Выполнить измерение характеристик компонентов программного продукта;
3. Произвести исследование созданного программного кода с использованием специализированных программных средств;
4. Провести сравнительный анализ программных продуктов и средств разработки.

В процессе практики был изучен ключевой цикл разработки программного обеспечения в компании, а также применены ранее приобретенные знания на практике в условиях реального производственного процесса.

# Общая характеристика компании

Малленом Системс – ведущая российская компания в области разработки и внедрения систем компьютерного зрения, промышленной видеоаналитики на основе технологий машинного зрения и искусственного интеллекта (машинное обучение, нейронные сети глубокого обучения) и интеллектуальной обработки данных.

Малленом Системс была создана в 2011 году на базе команды ученых и программистов Санкт-Петербургского политехнического университета Петра Великого. Сегодня в компании более 100 сотрудников. Глубокие компетенции в сфере машинного зрения и большой опыт успешной реализации проектов на промышленных предприятиях позволяет успешно решать большой спектр задач в различных отраслях. В Центре исследований и разработки интеллектуальных систем ведется работа по созданию новых решений и развитию продуктов компании. Более 5 лет компания готовит кадры по технологическому стеку компании в Акселераторе ML START.

## Организационная структура компании

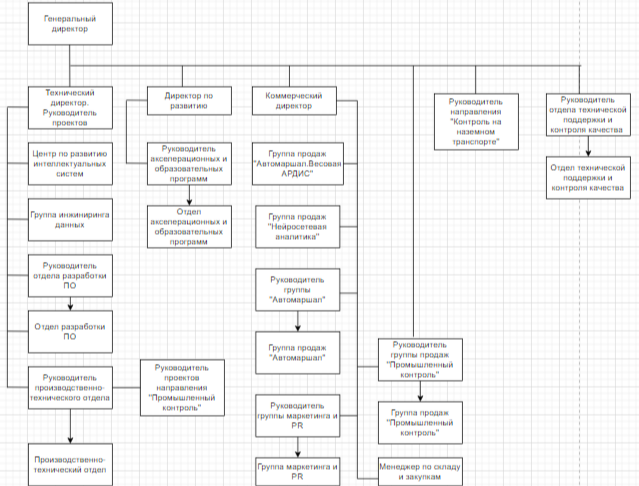


Рисунок 1 – организационная структура компании

## Внутренний распорядок работы компании и охрана труда

В компании "Малленом Системс" действует график работы 5/2, с 09:00 до 18:00. В штате компании есть отдельный специалист по охране труда, который проводит вводные инструктажи при приеме на работу и практике, а также занимается выдачей пропусков для пусконаладочных работ инженеров. В компании 20.09.2018 г. была проведена специальная оценка условий труда, согласно которой рабочие места, на территории которых установлены вредные производственные факторы, отсутствуют.

Так же производим обучение сотрудников по промышленной безопасности, по электробезопасности, охране труда и обучению в области применения средств защиты и оказания первой медицинской помощи в специализированном учебном центре (Негосударственное частное образовательное учреждение дополнительного профессионального образования «Учебный центр «Экоконсалт») с последующей аттестацией в Северо-Западном управлении Ростехнадзора в Вологодской области (только для промышленной безопасности и электробезопасности).

## Должностные инструкции ИТ-специалиста компании

* 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.

# Ревьюирование программных продуктов

Ревьюирование программных продуктов (или code review) — это важный этап разработки, который позволяет улучшить качество кода, обеспечить его соответствие стандартам и содействовать профессиональному росту разработчиков.

## Ревьюирование программного кода в соответствии с технической документацией

Ревьюирование программного кода в соответствии с технической документацией — важный этап разработки программного обеспечения, способствующий улучшению качества и надежности кода. Вот некоторые ключевые аспекты, на которые следует обратить внимание при осуществлении такого ревью:

1. Соответствие требованиям: Убедитесь, что код соответствует функциональным и нефункциональным требованиям, изложенным в технической документации.
2. Стиль кодирования: Проверьте, соблюдены ли стандарты кодирования, рекомендуемые для проекта или компании.
3. Читаемость: Код должен быть легко читаемым и понятным. Измена переменных и функций должны быть описательными и отражать их назначение.

Этап ревьюирования программного кода помогает выявить потенциальные проблемы на ранней стадии, что значительно снижает риски на следующих этапах разработки и последующей эксплуатации.

## Измерение характеристик компонентов программного продукта

**Измерение характеристик компонентов программного продукта** включает описание аспектов программной системы, отдельных её компонентов или процессов разработки с помощью математического аппарата.

Для измерения характеристик компонентов программного продукта могут использоваться **метрики —** характеристики программной системы, артефактов разработки (например, системной документации) или процессов разработки, которые можно объективно измерить. Примеры метрик: количество строк кода, количество дефектов на строку кода, время выполнения программы, покрытие кода тестами, цикломатическая сложность.

## Исследование созданного программного кода с использованием специализированных программных средств

Исследование программного кода с использованием специализированных программных средств является важным этапом в разработке, сопровождении и обеспечении безопасности программного обеспечения. Этот процесс называется анализом кода, он включает в себя динамический и статический анализ.

В статическом анализе специализированные программы автоматически анализируют исходный код без его выполнения. Они проверяют соответствие стандартам кодирования, выявляют потенциальные уязвимости, ошибки, утечки памяти и другие проблемы. Примеры инструментов: SonarQube, Pylint для Python, ESLint для JavaScript.

Динамический анализ кода проводится во время выполнения программы. Это позволяет выявить ошибки, которые могут не быть видны при статическом анализе. Например, проблемы с управлением памятью или неправильное поведение программы при различных входных данных. Примеры инструментов: Valgrind для C/C++, JUnit для Java.

## Сравнительный анализ программных продуктов и средств разработки

Сравнительный анализ программных продуктов и средств разработки выполняется для оценки различных решений, доступных на рынке, с целью выбора наиболее подходящего для конкретных нужд и задач. Он помогает выявить сильные и слабые стороны различных решений, что позволяет выбрать наиболее подходящий инструмент для конкретных задач. Такой анализ также способствует оптимизации затрат, упрощает процесс обучения пользователей и помогает в принятии обоснованных решений при внедрении новых технологий в организации.

Цель сравнительного анализа — помочь организациям и разработчикам сделать обоснованный выбор, уменьшить риски, связанные с внедрением нового программного обеспечения, и оптимизировать затраты.

# Выполняемые задания

* 1. Задания первой недели

Сначала я создала репозиторий GIT под названием «Practice\_PP03\_IS31» для своей производственной практики. В этом репозитории я разместила отчёты, выполненные задания и документы, которые требуются для данной практики.

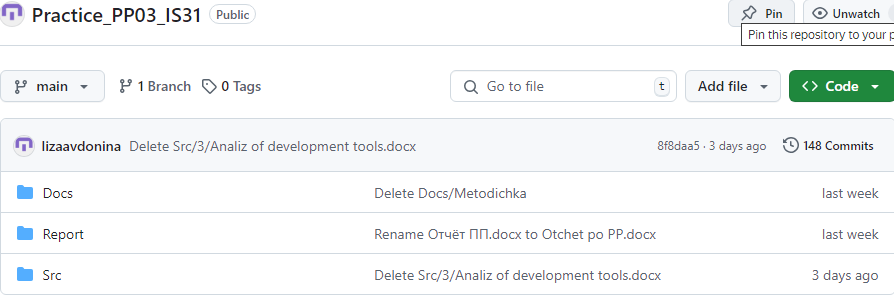


Рисунок 2 – репозиторий «Practice\_PP03\_IS31»

Далее мне дали задание разработать модуль обработки и работы с изображениями и модуля взаимодействия с пользователем, формирования и хранения данных: Поворот изображения.

Код программы представлен в приложении А.

Программный код нужно было разделить на следующие модули:

ImageModul.py – этот модуль упрощает некоторые основные операции обработки изображений, такие как загрузка, изменение размера и поворот. 

Рисунок 4 – модуль ImageModul.py программному коду «Поворот изображения»

MainModul.py - файл, который запускает приложение.

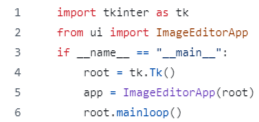


Рисунок 5 – модуль MainModul.py к программному коду «Поворот изображения»

UiModul.py – этот модуль используется для создания пользовательского интерфейса. Представлен в приложении Б.

3.2 Задания второй недели

Следующим заданием требовалось выполнить реверс-инжиниринг программного кода, который осуществляет "Поворот изображения", с использованием UML-диаграмм. Было необходимо создать и описать диаграммы Компонентов, Сценариев использования, Последовательности и Деятельности.

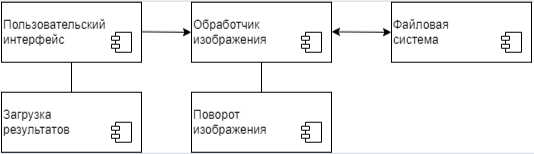


Рисунок 7 – диаграмма компонентов к программному коду «Поворот изображения»

• Пользовательский интерфейс предоставляет пользователю доступ к функционалу программы. Пользователь может загружать изображения и инициировать их поворот.

* Получает изображения от Пользовательского интерфейса. Выполняет операции по изменению изображений, такие как поворот.
* Файловая система устанавливает взаимодействие между приложением и системными файлами.

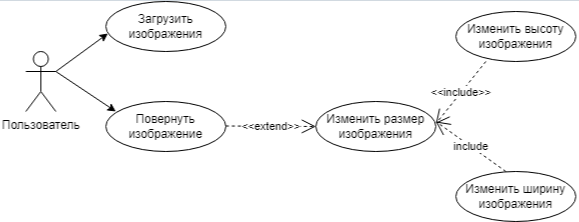


Рисунок 8 – диаграмма сценариев использования к программному коду «Поворот изображения»

• Пользователь загружает изображение в приложение для его последующей обработки.

• Основной сценарий использования, где пользователь может изменить ориентацию загруженного изображения.

• Изменить высоту и ширину изображения - это дополнительная функциональность, которая может быть использована вместе с поворотом изображения по мере необходимости.

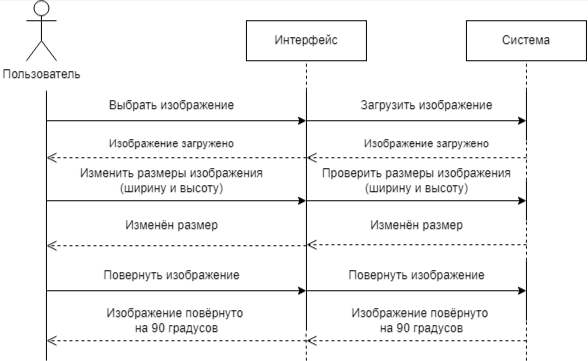


Рисунок 9 – диаграмма последовательностей к программному коду «Поворот изображения»

• Пользователь выбирает изображение для загрузки или обработки.

• Система загружает выбранное изображение.

• Пользователь вводит новые размеры (ширина и высота).

• Система проверяет допустимость новых размеров.

• Пользователь указывает на необходимость поворота изображения.

• Система выполняет операцию поворота и отправляет готовое изображение обратно на интерфейс.

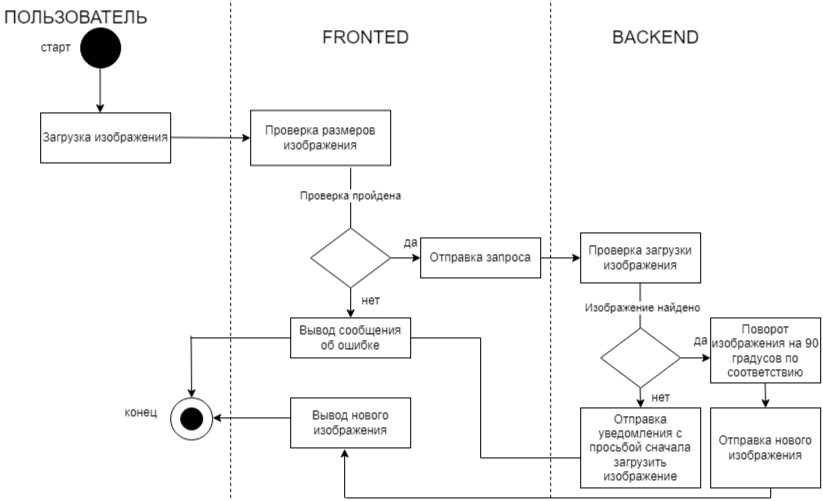


Рисунок 10 – диаграмма деятельности к программному коду «Поворот изображения»

• Пользователь вводит новые размеры для изображения (ширину и высоту).

• Fronted проверяет размеры изображения (ширину и высоту)

• Если проверка не удалась:

• Fronted выводит сообщение об ошибке.

• Если проверка удалась:

• Fronted отправляет запрос Backend.

• Backend проверяет размеры изображения на соответствие запросу.

• Если изображение не найдено:

• Backend отправляет уведомление с просьбой сначала загрузить изображение.

• Fronted выводит сообщение об ошибке.

• Если изображение найдено:

• Backend поворачивает изображение на 90 градусов по соответствию.

• Backend отправляет новое изображение.

• Fronted выводит новое изображение.

Следовало также протестировать скорость выполнения программного кода для «Поворота изображения» и его модулей.

Таблица 1 – Скоростные показатели программного кода и его модулей

|  |  |
| --- | --- |
| Программный код «Поворот изображения» | Время загрузки изображения: 0,033952 секунд. Изменение размера: 0,2548 секунд. |
| Модуль ImageModulTime.py | Время выполнения load\_image: 0,0056789 секунд. Время выполнения resizeimage: 0,2765948 секунд. Время выполнения rotate\_image: 0.0135498 секунд. |
| Модуль MainModulTime.py | Время запуска приложения: 1,2976548 секунд. Время отклика на команды пользователя: 0,023545666 секунд. |
| Модуль UiModulTime.py | Время выполнения load\_image: 0,016585 секунд. Время выполнения: display\_image: 0,05674 секунд. Время выполнения resize\_image: 0,01345683 секунд. |

Таблица 2 – Размеры программного кода и его модулей

|  |  |
| --- | --- |
| Программный код «Поворот изображения» | Размер: 296542. Размер модуля sys 75 байт. |
| Модуль ImageModulTime.py | Размер объекта 49 байт |
| Модуль MainModulTime.py | Размер объекта 778439 байт. |
| Модуль UiModulTime.py | Размер объекта 51 байт |

Я выполнила исследование средств разработки программ на языке программирования Python.

VisualStudioCode

Плюсы:

* Бесплатность и открытость: VS Code распространяется бесплатно и является продуктом с открытым исходным кодом, что делает его доступным для широкого круга пользователей.
* Кроссплатформенность: он доступен для Windows, macOS и Linux, что позволяет использовать его на большинстве популярных операционных систем.
* Интеграция с Git: Встроенная поддержка систем контроля версий, включая Git, позволяет удобно работать с репозиториями.

Минусы:

* Зависимость от расширений: Многие функции доступны только через сторонние расширения, что может приводить к проблемам с совместимостью или качеством некоторых из них.
* Отсутствие некоторых встроенных функций IDE: Такие функции, как встроенный редактор UML диаграмм или более сложные инструменты профилирования, могут отсутствовать.

Вид приложения представлен в приложении В.

PyCharm

Плюсы:

* Интеллектуальное автодополнение: PyCharm предоставляет контекстное автодополнение кода, которое ускоряет процесс написания кода и снижает количество ошибок.
* Панели терминала и базы данных: Встроенные средства позволяют работать с терминалом и базами данных непосредственно в IDE.
* Кроссплатформенность: PyCharm доступен для Windows, macOS и Linux.

Минусы PyCharm:

* Ресурсоемкость: PyCharm может быть довольно требовательной к ресурсам системы, что может привести к снижению производительности на менее мощных компьютерах.
* Платная версия: Полная функциональность доступна только в платной версии — PyCharm Professional.

Вид приложения представлен в приложении Г.

WING IDE

Плюсы:

* Рефакторинг кода: Инструменты для автоматического рефакторинга кода помогают улучшать качество и структуру программ без изменения их функциональности.
* Специализация на Python: Wing IDE специально разработана для Python, что делает её мощной и оптимизированной для работы с этим языком.
* Проверка кода в реальном времени: IDE предлагает встроенную проверку кода на лету, которая позволяет идентифицировать ошибки и предупреждения сразу же при написании кода.

Минусы:

* Стоимость: Полная версия Wing IDE является платной, и это может быть сдерживающим фактором для индивидуальных разработчиков или небольших команд с ограниченным бюджетом.
* Ресурсозатратность: Запуск полной версии Wing IDE требует значительных ресурсов системы, что может замедлить работу на слабых компьютерах.

Вид приложения представлен в приложении Д.

IDLE (Integrated Development and Learning Environment)

Плюсы:

* Простота использования: IDLE имеет интуитивно понятный и минималистичный интерфейс, что делает его подходящим вариантом для новичков, только начинающих изучать Python.
* Синтаксическая подсветка: Среда предлагает подсветку синтаксиса, что помогает легче читать и писать код.
* Интерактивный режим: IDLE предоставляет интерактивный режим (REPL), который позволяет быстро тестировать небольшие куски кода, что особенно полезно для экспериментов и обучения.

Минусы:

* Производительность: Может быть не оптимален при работе с большими проектами или файлами из-за ограниченных возможностей по управлению проектами.
* Пользовательский интерфейс: Имеет устаревший вид и ограниченные возможности кастомизации по сравнению с другими современными IDE.

Вид приложения представлен в приложении Е.

Из представленных выше средств разработки программ я выбрала IDLE, потому что:

* IDLE известен своей простотой и краткостью синтаксиса, что делает его легким для изучения и использования, особенно для начинающих программистов.
* IDLE поддерживает одно из крупнейших сообществ разработчиков. Это означает, что есть множество учебных материалов, документации и форумов, где можно получить помощь.
* IDLE работает на всех основных платформах, включая Windows, macOS и Linux, что делает его универсальным инструментом для разработки.
* IDLE поддерживает объектно-ориентированное, процедурное и функциональное программирование, что позволяет выбирать наиболее подходящий стиль для решения конкретных задач.

Поддержка языка и функции для разработчиков:

1. Отладка и Тестирование: Встроенные модули, такие как pdb для отладки и unittest, pytest для тестирования, облегчают разработку надежного кода.
2. Интерактивный интерпретатор: Python предоставляет интерактивный интерпретатор, который удобен для экспериментирования и быстрого прототипирования.
3. Автоматизация задач: Поддержка инструментов автоматизации, таких как fabric и invoke, позволяет легко автоматизировать развертывание и другие рабочие процессы.
4. Поддержка современных стандартов: Поддержка таких современных функций, как генераторы, функторы, аннотации типов и асинхронное программирование.

# Заключение

На практике я освоила следующее:

1. Осуществление ревьюирования программного кода в соответствии с технической документацией;
2. Выполнение изменений характеристик компонентов программного продукта;
3. Произведение исследования созданного программного кода с использованием специализированных программных средств;
4. Проведение сравнительного анализа программных продуктов и средств разработки.

Также приобрела ключевые навыки в области «Ревьюирование программных модулей» и развила соответствующие общие и профессиональные компетенции.

э

# Список используемых источников

1. Метанит. Python. Урок 2.10. Синтаксис языка Python. URL: <https://metanit.com/python/tutorial/2.10.php>.
2. Метанит. C#. Урок 3.46. Порядок вызова членов класса. URL: <https://metanit.com/sharp/tutorial/3.46.php>.
3. [Diagrams.net](https://Diagrams.net). Онлайн инструмент для создания диаграмм. URL: <https://app.diagrams.net/>.
4. Mallenom. UML-диаграммы. URL: <https://www.mallenom.ru/?ysclid=m4l2ditzeb835497887>.
5. Practicum by Yandex. Блог о UML-диаграммах. URL: <https://practicum.yandex.ru/blog/uml-diagrammy/> .
6. GeeksforGeeks. Как проверить время выполнения скрипта на Python. URL: https://www.geeksforgeeks.org/how-to-check-the-execution-time-of-python-script/Приложения

Приложение А







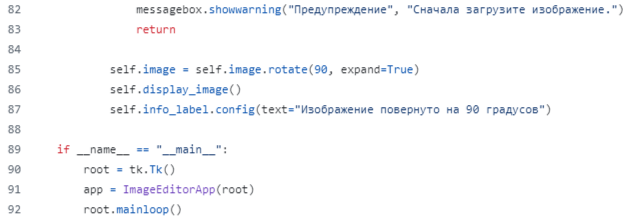
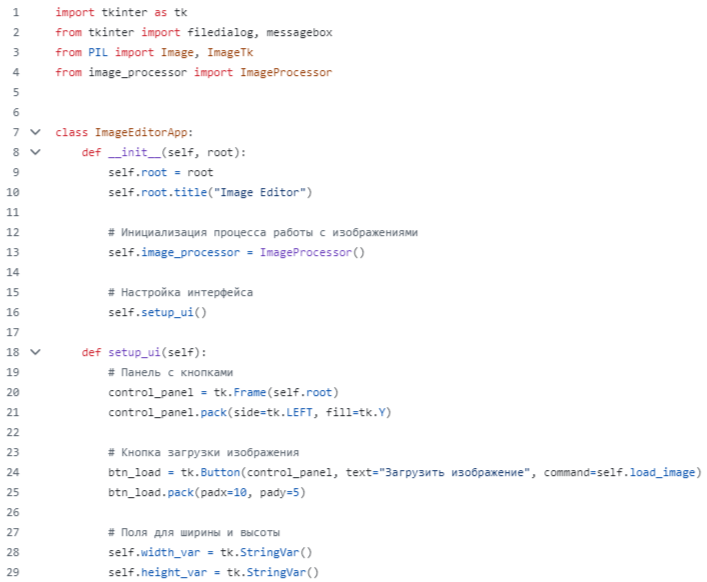


Рисунок 3 – программный код «Поворот изображения»

Приложение Б







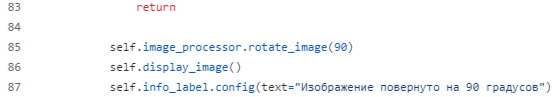


Рисунок 6 – модуль UiModul.py к программному коду «Поворот изображения»

л

Приложение В

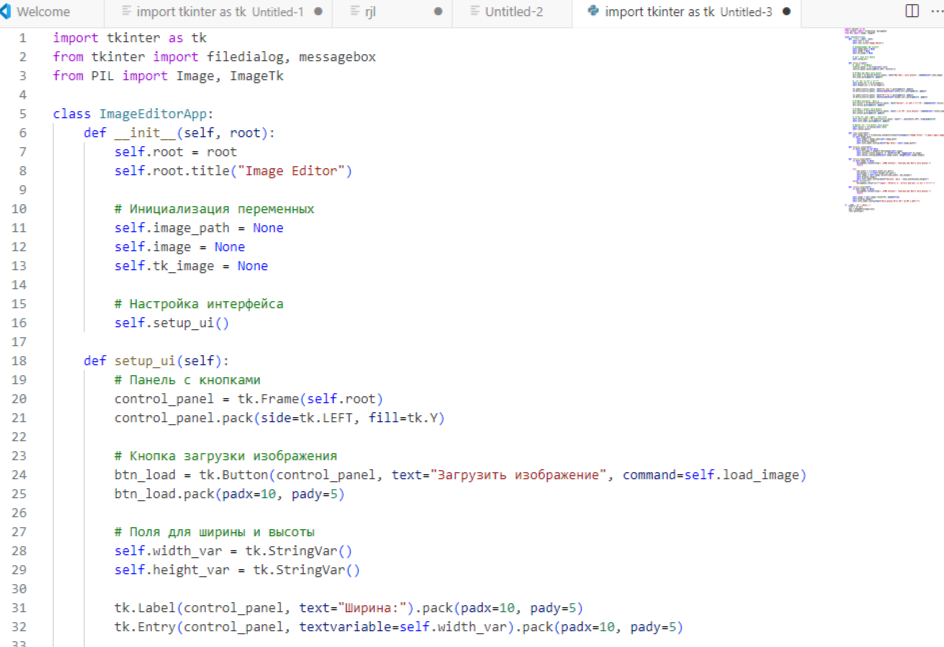


Рисунок 11 – вид приложения VisualStudio

э

Приложение Г

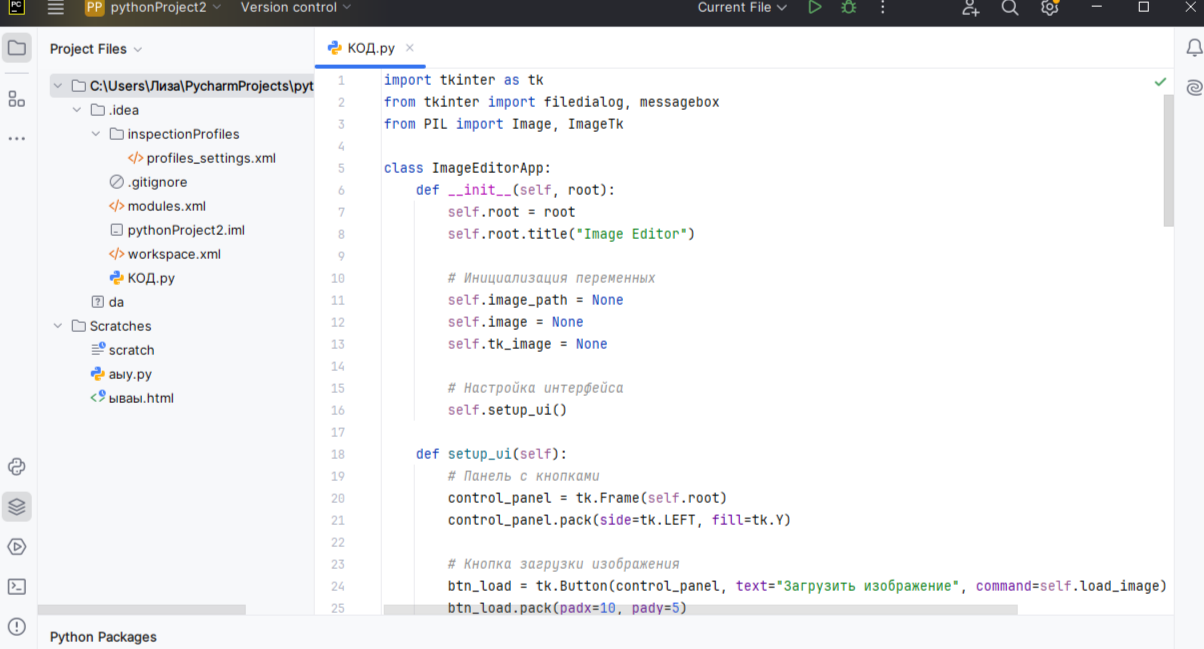


Рисунок 12 – вид приложения PyCharm

Приложение Д

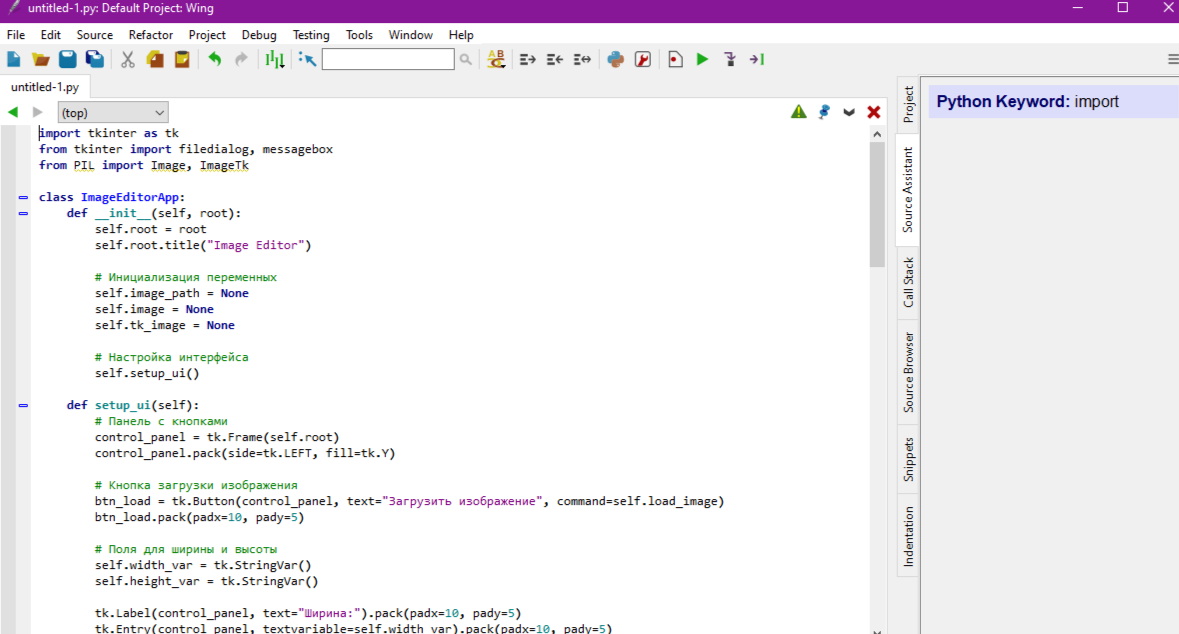


Рисунок 13 – вид приложения WING IDE

Приложение Е

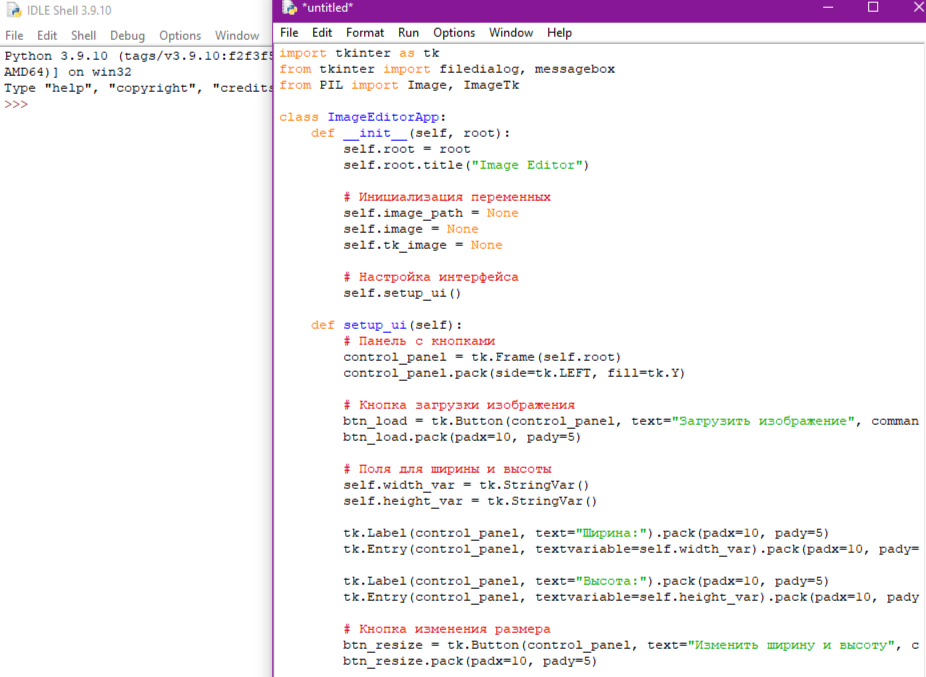


Рисунок 14 – вид приложения IDLE