

позволяющими создавать более масштабируемые и гибкие архитектуры программ.

С учетом всего вышесказанного можно сказать, что использование абстрактных и виртуальных классов в C++ требует более тщательного и продуманного проектирования, однако предоставляет разработчику мощный инструмент для тонкой настройки работы с памятью и управления структурой программы.

Список использованной литературы

1. Virtual inheritance [Электронный ресурс] / Википедия. – Режим доступа: https://en.wikipedia.org/wiki/Virtual_inheritance. – Дата доступа: 12.03.2025.

2. Виртуальный базовый класс [Электронный ресурс] / Программирование на C и C++. – Режим доступа: <https://www.c-cpp.ru/books/virtualnyy-bazovyuy-klass>. – Дата доступа: 12.03.2025.

РАЗРАБОТКА ДЕСКТОПНЫХ ПРИЛОЖЕНИЙ НА PYTHON

Полын Серафим (УО МГПУ им. И.П. Шамякина, г. Мозырь)

Научный руководитель – В.В. Давыдовская, канд. физ.-мат. наук, доцент

Разработка десктопных приложений остаётся актуальной задачей для многих разработчиков. Такие программы широко применяются в различных сферах – от корпоративного программного обеспечения до личных утилит. Несмотря на рост популярности веб- и мобильных приложений, десктопные решения обладают рядом преимуществ, таких как высокая производительность, офлайн-доступ и более глубокая интеграция с операционной системой [1].

Python является одним из наиболее удобных языков для создания настольных приложений благодаря своей простоте, читаемости кода и огромному количеству библиотек. Он позволяет разрабатывать как простые утилиты, так и сложные кроссплатформенные системы с графическим интерфейсом.

Существует несколько популярных инструментов для создания GUI-приложений на Python: PyQt, Tkinter, Kivy, wxPython, PySide.

PyQt – это набор привязок к фреймворку Qt, который позволяет разрабатывать кроссплатформенные приложения с современным интерфейсом. Он поддерживает Windows, macOS и Linux, а также обладает богатым функционалом для работы с виджетами, окнами, меню и событиями. Qt предоставляет широкий набор компонентов, включая кнопки, текстовые поля, таблицы, вкладки и графики, а также позволяет создавать сложные макеты и настраивать внешний вид приложения. PyQt поддерживает систему сигналов и слотов, что упрощает обработку событий и взаимодействие между элементами интерфейса. Кроме того, PyQt позволяет использовать OpenGL для работы с 3D-графикой и поддерживает мультимедийные возможности, включая воспроизведение видео и аудио [2].

Для удобства работы с интерфейсом можно использовать программу Qt Designer. Это визуальный редактор, позволяющий быстро создавать формы, кнопки, поля ввода и другие элементы без необходимости писать код вручную. Готовый интерфейс сохраняется в файле .ui, который затем можно конвертировать в Python-код с помощью утилиты ruic5.

Разработка десктопных приложений на Python с использованием PyQt и других GUI-фреймворков предоставляет разработчикам мощные инструменты для создания удобного и функционального интерфейса. PyQt выделяется среди других решений благодаря своей гибкости, богатому набору компонентов и удобному механизму сигналов и слотов, упрощающему обработку событий [3].

Использование Qt Designer значительно ускоряет процесс проектирования интерфейса, позволяя сосредоточиться на бизнес-логике приложения. В результате разработка становится менее трудозатратной, а конечный продукт – более качественным.

Благодаря своей кроссплатформенности и широким возможностям PyQt остаётся одним из лучших решений для создания настольных приложений на Python. В дальнейшем разработчик может интегрировать в приложение базы данных, сетевые функции и мультимедийные элементы, расширяя его возможности и делая его более универсальным.

Список использованной литературы

1. Rossum, G. van. The Python Language Reference / G. van Rossum // Python Software Foundation. – 1995. – Режим доступа: <https://docs.python.org/3/reference/>. – Дата доступа: 15.03.2025.

2. Здоров, Д. Python GUI Programming with PyQt : [практ. руководство] / Д. Здоров. – 2-е изд. – М., 2019. – 359 с.

3. Гудредж, К. Python для сложных задач: наиболее эффективные подходы к разработке ваших программ / К. Гудредж. – М. : ДМК Пресс, 2018. – 220 с.

ФОРМИРОВАНИЕ ФУНКЦИОНАЛЬНОЙ ГРАМОТНОСТИ С ИСПОЛЬЗОВАНИЕМ КОМПЛЕКТОВ РОББО НА УРОКАХ ФИЗИКИ

Сапсалёва Елена (УО МГУ им. А.А. Кулешова, г. Могилёв)

Научный руководитель – Е.В. Тимощенко, канд. физ.-мат. наук, доцент

В современном обществе непрерывно происходят процессы цифровой трансформации, поэтому информационные технологии становятся все более востребованными в системе образования [2]. Особое внимание уделяется внедрению новых технологий, и робототехника в этом направлении занимает особое место. Использование робототехнических комплектов РОББО на уроках физики демонстрирует применение теоретических знаний на практике. Например, изучение принципов механики, электричества и магнетизма становится более наглядным и понятным. Учащиеся развивают навыки решения физических задач, анализируют результаты экспериментов и делают выводы. Это развивает их функциональную грамотность и критическое мышление.