

РАЗРАБОТКА ГРАФИЧЕСКОГО РЕДАКТОРА С ИСПОЛЬЗОВАНИЕМ ОБЪЕКТНО-ОРИЕНТИРОВАННОГО ПОДХОДА

Введение. В современном мире графика играет важную роль в компьютерной технологии. Разработка графических редакторов и программ для работы с графикой является сложным и увлекательным процессом, который требует от разработчиков глубоких знаний в области программирования, дизайна и эргономики. Эти программы позволяют пользователям выразить свою креативность и создавать уникальные произведения искусства. Однако, разработка качественного и интуитивно понятного приложения для рисования является сложной задачей, требующей глубоких знаний в области программирования и дизайна пользовательских интерфейсов.

Графический редактор — программное обеспечение, которое позволяет пользователям создавать и редактировать графические изображения, используя различные инструменты, такие как кисть, ластик, заливка цветом, выбор цвета и т. д. С помощью графического редактора пользователи могут создавать простые рисунки, диаграммы, аннотации и т. д.

Основная часть. Целью исследования является создание графического редактора на объектно-ориентированном языке программирования C# используя возможностей пользовательского интерфейса Windows Forms.

При разработке ставились задачи по созданию приложения предназначенного для развлечения, развития творческих способностей и воображения.

В качестве средств реализации проекта была использована среда разработки Visual Studio с использованием языка программирования C#. Visual Studio является богатой инструментальными средствами, средой программирования, предоставляющей функциональные возможности, необходимые для создания больших или малых проектов, запускаемых под управлением Windows.

Для понимания архитектуры системы на рисунке 1 представлена диаграмма классов разрабатываемого приложения, которая отображает структуру объектов и их атрибутов, а также показывает взаимодействие между объектами и их методами.

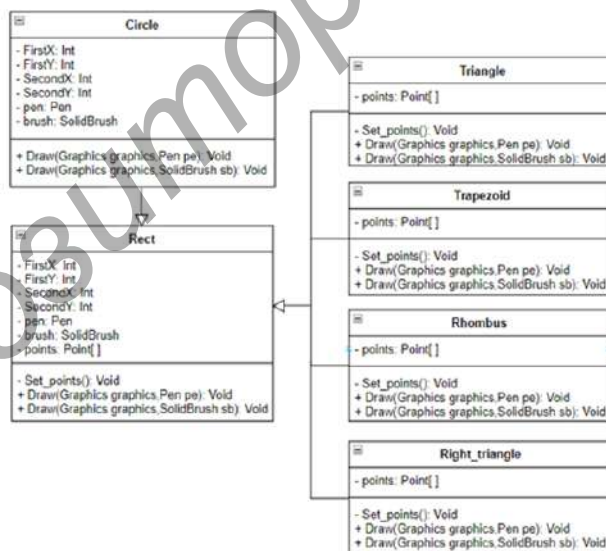


Рисунок 1 — Диаграмма классов разрабатываемого приложения

В качестве приемов основных принципов объектно-ориентированного программирования использованы:

1. Наследование — это механизм, который позволяет создавать новые классы на основе уже существующих классов. Новые классы наследуют свойства и методы базового класса и могут добавлять свои собственные свойства и методы [1]. Это позволяет избежать дублирования кода и упрощает структуру программы, что улучшает ее поддержку и развитие. Классы «Треугольник», «Прямоугольный треугольник», «Ромб», «Окружность» и «Трапеция» наследуют поля и методы от класса «Фигура».

2. Виртуальные функции — это функции в языке программирования, которые могут быть переопределены в производных классах. Они используются там, где объекты могут принимать множество форм и действовать по-разному в зависимости от их конкретного типа и контекста выполнения [2]. В базовом классе присутствуют чисто виртуальные функции установки точек и прорисовки, которые переопределены в производных классах, что делает базовый класс полиморфным и абстрактным.

3. Инкапсуляция — позволяет объединить внутреннее состояние объекта и методы работы с этим состоянием, скрыть внутреннюю реализацию объекта и предоставить только интерфейс для работы с ним [3].

Основные задачи приложения:

- рисование геометрических фигур;
- изменение размеров геометрических фигур путем их растягивания/сжатия;
- просмотр параметров нарисованной фигуры (рисунок 2);
- выбор цвета и заливка;
- использование ластика и пипетки;
- сохранение изображения в файл;
- загрузка изображения из файла.
- рисование кистью, карандашом (рисунок 3).

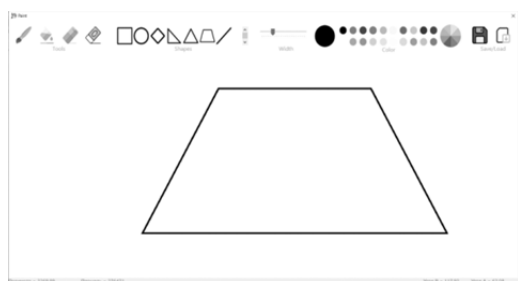


Рисунок 2 — Рисование с просмотром параметров фигуры

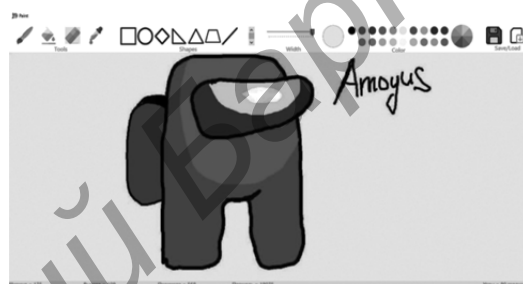


Рисунок 3 — Процесс рисования

Преимущества разрабатываемого приложения:

1. Удобство использования. Графический редактор облегчает работу с изображениями и позволяет быстро редактировать их.

2. Широкий функционал. Редактор обеспечивает широкий спектр функций и инструментов, которые позволяют решать многие задачи, связанные с графикой.

3. Быстрота обработки изображений. Графический редактор позволяет быстро редактировать и изменять любые параметры изображения.

4. Возможность работать с различными форматами изображений.

Заключение. В результате выполнения проекта было создано приложение, которое можно использовать для развития воображения и творческих способностей. Приложение обладает интуитивно понятным и приятным интерфейсом. Использование объектно-ориентированного подхода позволяет упростить разработку и сделать ее более гибкой, а также облегчить поддержку и доработку приложения в будущем. Кроме того, объектно-ориентированный подход позволяет более полно учесть требования заказчика и целевую аудиторию, что повышает качество разработки.

Список цитируемых источников

1. Буч, Г. Объектно-ориентированный анализ и проектирование с примерами приложений на C++ / Г. Буч ; пер. с англ. И. Романовский, Ф. Андреев. — М. : Бинум, 2012. — 560 с.
2. Фридман, А. Л. Основы объектно-ориентированной разработки программных систем / А. Л. Фридман. — М. : Финансы и статистика, 2010. — 97с.
3. Бадд, Т. Объектно-ориентированное программирование в действии / Т. Бадд. — СПб. : Питер, 2011. — 464 с.