

Приложение-помощник в учебе «UniHelp»

Авторы: ученики 10 «Б» класса школы № 1501 г. Москва,
Чагаев Егор Алексеевич
Ильясов Тимур Ильгизович

Руководитель: Дральщиков Никита Сергеевич

Актуальность и проблема проекта

Несмотря на наличие множества учебных инструментов, они не систематизированы, что вынуждает учащихся использовать несколько программ одновременно, снижая эффективность обучения.

Цель

Создать интегрированную программу-помощник на Python, которая объединит функции калькулятора, построения графиков, решения уравнений и предоставит доступ к учебным материалам, с целью улучшения и упрощения процесса обучения

Задачи

1. Проанализировать потребности учащихся в учебных инструментах.
2. Спроектировать архитектуру программы с нужным функционалом.
3. Реализовать следующие модули программы:
 - Калькулятор для выполнения различных вычислений;
 - Инструмент для построения графиков математических функций;
 - Модуль для решения алгебраических уравнений;
 - Библиотека для хранения и доступа к электронным учебникам;
4. Провести тестирование программы и оптимизировать её работу.

Анализ потребностей учащихся:

- 45% испытывают трудности с расчётами ежедневно
- 62% испытывают сложности с доступом к учебным материалам
- 67% нуждаются в специализированных калькуляторах

Какие инструменты вам чаще всего не хватает для эффективного изучения сложных тем?

- Инструменты для пошагового решения задач 45%
- Графическое представление функций (графики) 25%
- Доступ к библиотеке учебников и справочникам 30%

Анализ источников информации:

Название библиотеки/фреймворка	Преимущества	Недостатки
Tkinter	<ul style="list-style-type: none">•Интегрирована в Python•Кроссплатформенность•Большое сообщество	<ul style="list-style-type: none">•Устаревший интерфейс•Ограниченный функционал
Kivy	<ul style="list-style-type: none">•Простота•Кроссплатформенность	<ul style="list-style-type: none">•Ограниченность функционала•Собственный язык разметки

Название библиотеки/фреймворка	Преимущества	Недостатки
PySimpleGUI	<ul style="list-style-type: none">•Наличие многих цветовых тем для приложений•Кроссплатформенность	<ul style="list-style-type: none">•Необходимость регистрации для разработки приложений•Сложно интегрировать дизайн

Название библиотеки/фреймворка	Преимущества	Недостатки
PyQt6	<ul style="list-style-type: none">•Простота использования•Встроенное решение для создания дизайна приложений•Поддержка CSS и XML•Кроссплатформенность•Большое сообщество•Высокая производительность	<ul style="list-style-type: none">•Сложность обучения•Несовместимость с некоторыми модулями Python

Анализ аналогов:

- Desmos – онлайн-приложение для построения сложных графиков и визуализации функций, но требует подключения к интернету и не имеет оффлайн-версии.
- Photomath – мобильное приложение для пошагового решения уравнений с объяснением, однако полные функции доступны только по подписке.
- Библиотека МЭШ – онлайн-платформа для доступа к учебным материалам, но требует интернет-соединения и не позволяет загружать материалы для оффлайн-доступа.

Промежуточный вывод :

Для разработки была выбрана библиотека **PyQt6** благодаря её удобству, функциональности и производительности. Программа включает модули:

- **Калькулятор** с высокой производительностью;
- **Построение графиков**, доступное оффлайн;
- **Решение уравнений** с удобным интерфейсом;
- **Библиотека учебников** для оффлайн-доступа.

Все модули имеют простой интерфейс и приятную цветовую гамму, что важно для учащихся, проводящих много времени за компьютером.

> .venv library root

▼ data

- classes.py
- functions.py

▼ db

- textBookDB.db

▼ forms

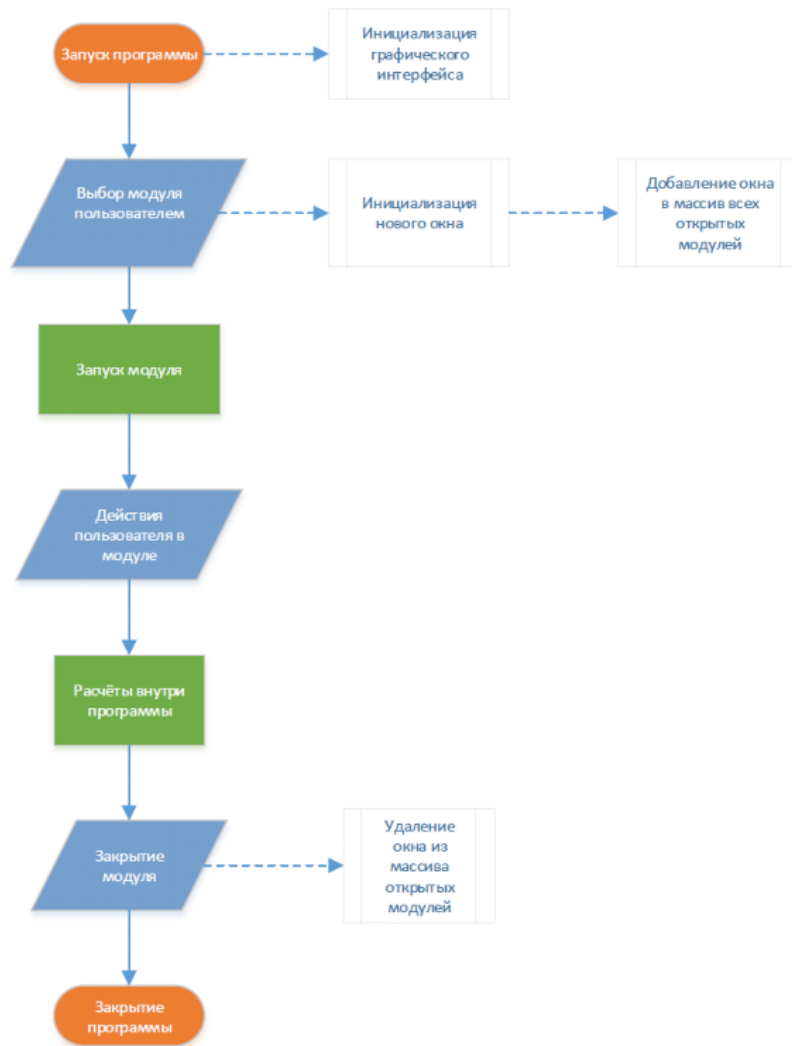
- AboutWindow.ui
- AboutWindow_ui.py
- BookLibraryWindow.ui
- BookLibraryWindow_ui.py
- CalcWindow.ui
- CalcWindow_ui.py
- EquationsWindow.ui
- EquationsWindow_ui.py
- GraphWindow.ui
- GraphWindow_ui.py
- MainWindow.ui
- MainWindow_ui.py
- ReadMeGraphWindow.ui
- ReadMeGraphWindow_ui.py

> img

> materials

- main.py

Ход работы: создание структуры папок проекта



Ход работы:
разработка схемы
работы
приложения.

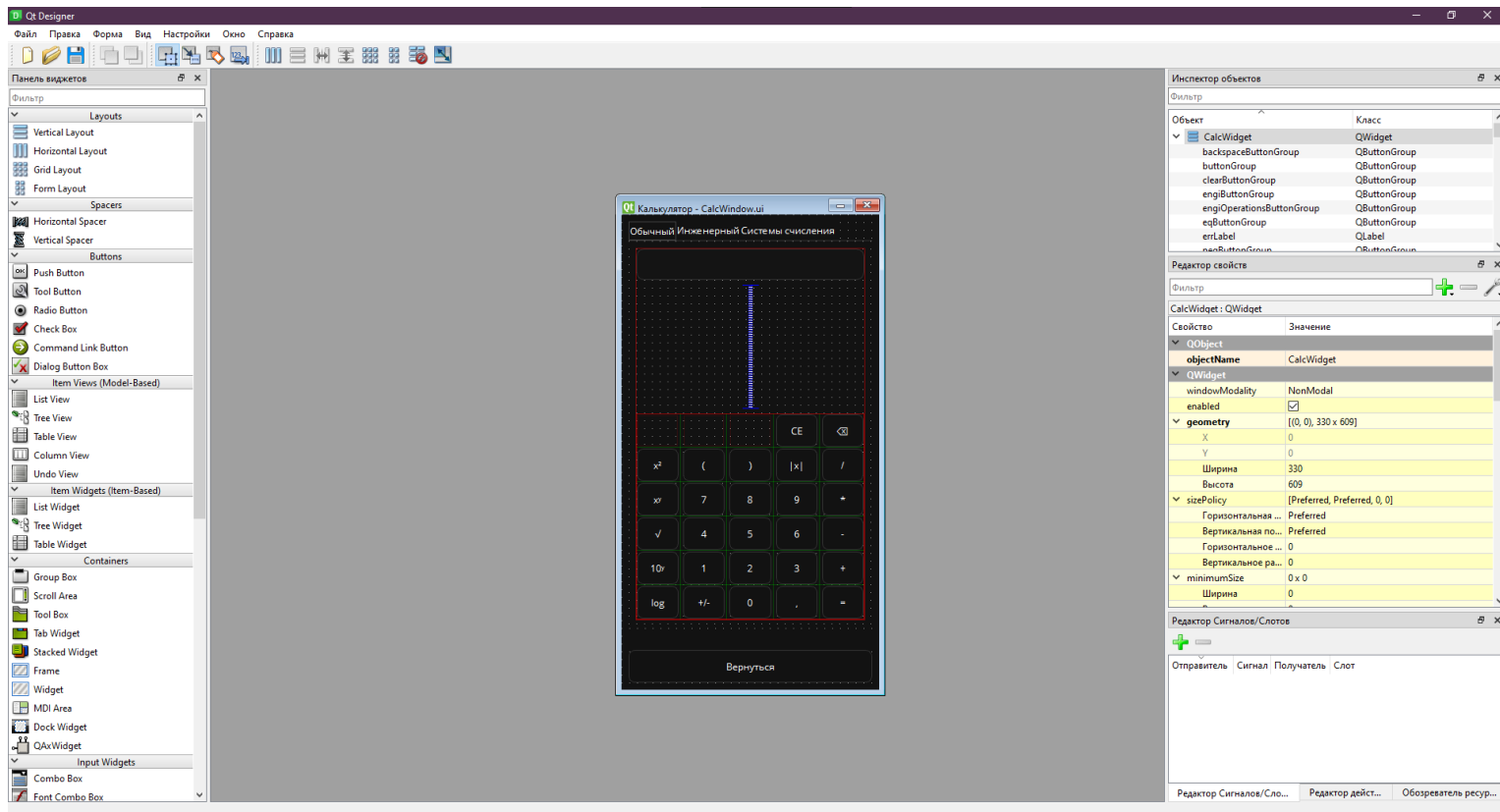
Ход работы: выбор среды разработки.

Критерий	Pycharm Community Edition	Visual Studio Code
Опыт работы	Егор: 2 года Тимур: 1.5 года	Егор: менее полугода Тимур: менее полугода
Работа с языком Python	<ul style="list-style-type: none">•Подсветка синтаксиса•Подсказки типов переменных•Подсказки при написании кода•Удобная отладка	<ul style="list-style-type: none">•Подсветка синтаксиса•Подсказки типов переменных

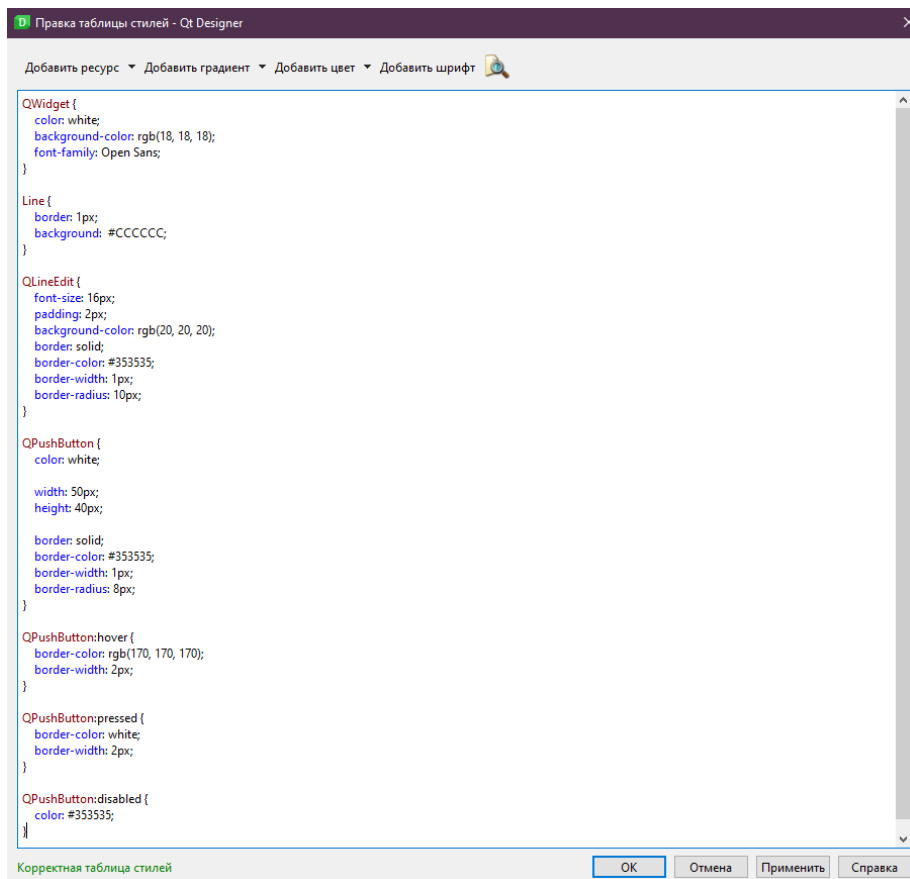
Ход работы: выбор среды разработки.

Критерий	Pycharm Community Edition	Visual Studio Code
Работа с git и github	Интуитивно понятный интерфейс git	Наличие инструментов работы git
Удобные функции	Наличие функции «external tools». Нам это пригодилось для конвертации форм из файлов с «.ui» расширением в файлы с расширением «.py».	

Ход работы: создание форм.



Ход работы: изменение дизайна форм.



Ход работы: Добавление функций к кнопкам на формах.

```
# Обработчики нажатий кнопок
self.eqBtn.clicked.connect(self.add_widget)
self.calcBtn.clicked.connect(self.add_widget)
self.bookLibButton.clicked.connect(self.add_widget)
self.graphBtn.clicked.connect(self.add_widget)
```

Испытание работоспособности прототипа :

В ходе отладки приложения были выявлены и устранены недочёты. Программа работает стабильно и завершает работу только по желанию пользователя.

Все модули функционируют корректно: калькулятор правильно обрабатывает выражения и переводит их в разные системы счисления, модуль графиков строит функции на заданном отрезке, уравнения решаются быстро, а поиск учебников осуществляется точно и быстро.

Выводы

В результате работы над проектом получилось готовое полностью функциональное Desktop-приложение. Все задачи были реализованы полностью. Все модули приложения-помощника работают правильно: быстро и без зависаний. Приложение не вызывает фатальный сбой и закрывается только после соответствующего действия пользователя.

Спасибо за внимание!

Авторы: Чагаев Егор Алексеевич
Ильясов Тимур Ильгизович

Обратная связь:
gerodann@gmail.com