

Министерство образования Республики Беларусь

Учреждение образования
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет Информационных технологий и управления
Кафедра Интеллектуальных информационных технологий

ОТЧЁТ

по лабораторной работе №4

по дисциплине “Проектирование программного обеспечения в интеллектуальных системах”

Выполнили:

Р. В. Липский, гр. 121701

Проверил:

С. В. Бутрин

Цель и постановка задачи

Цель: Изучить архитектурные приемы построения пользовательского интерфейса

Задание:

За основу берется результат лабораторной работы №1. Необходимо разработать графический пользовательский интерфейс к программной системе, которая получилось в л/р №1.

Необходимо сохранить/выделить общий код для л/р №1 и л/р №4, т.е. в результирующем проекте должна быть возможность запустить приложение как с GUI, так и CLI.

GUI

```
from cli2gui import Cli2Gui

from src.pyslang import main, parse_args

generate_gui = Cli2Gui(
    run_function=main,
    auto_enable=True,
    gui="pysimplegui",
)

gui = generate_gui(parse_args)
if __name__ == "__main__":
    gui()
```

Первая строка импортирует класс Cli2Gui из модуля cli2gui.

Вторая строка импортирует функции main и parse_args из модуля src.pyslang.

Третья строка создает экземпляр класса Cli2Gui с передачей аргументов:

run_function=main - указывает функцию, которая будет выполняться при запуске в GUI.
auto_enable=True - автоматически включает графический интерфейс, если он доступен.

gui="pysimplegui" - указывает, что используется библиотека PySimpleGUI для создания GUI.

Четвертая строка вызывает метод generate_gui и передает ему функцию parse_args. Этот метод генерирует GUI на основе информации, полученной из функции parse_args. Функция parse_args используется для разбора аргументов командной строки.

Пятая строка проверяет, является ли модуль основным (то есть запущен ли скрипт напрямую, а не импортирован). Если это так, то выполняется шестая строка, которая запускает GUI.