**МИНИСТЕРСТВО ОБРАЗОВАНИЯ КРАСНОЯРСКОГО КРАЯ**

**КРАЕВОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ПРОФЕССИОНАЛЬНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ**

**«КРАСНОЯРСКИЙ КОЛЛЕДЖ ОТРАСЛЕВЫХ ТЕХНОЛОГИЙ И ПРЕДПРИНИМАТЕЛЬСТВА»**

****ОТЧЕТ ПО МОДУЛЮ №3****

**Преподаватель: \_\_\_\_\_\_\_ \_\_\_\_\_\_\_ Глухов С.Ю.**

подпись

дата

**Исполнитель**

**студент: 2 курса группы ИС23-01 \_\_\_\_\_\_ \_\_\_\_\_ Гвоздева Д.Д.**

подпись

дата

**Красноярск, 2025 г.**

****Введение****

Проект представляет собой интерактивную настольную энциклопедию, созданную на Python с использованием библиотеки Tkinter. Приложение имеет удобный графический интерфейс, где вся информация систематизирована по алфавитному принципу: для каждой буквы предусмотрен отдельный раздел со списком соответствующих терминов и их подробными определениями.

Данный проект был реализован в качестве учебной работы, основной целью которой являлось освоение ключевых аспектов разработки графических интерфейсов с помощью Tkinter. В процессе создания энциклопедии были изучены принципы работы с виджетами, организация навигации между разделами, а также реализация дополнительных функций для повышения удобства использования приложения.

**Цель проекта:**

* Получить практический опыт работы с библиотекой Tkinter для создания современных графических интерфейсов.
* Разработать интуитивно понятное приложение с удобной системой навигации по алфавитному каталогу.

**Задачи:**

1. Разработка интерфейса:

* Создать основной GUI с использованием Tkinter.
* Реализовать систему тематического оформления (светлая/тёмная темы).

1. Организация данных:

* Структурировать информацию по алфавитному принципу.
* Обеспечить хранение данных в JSON-формате.

1. Функциональность:

* Реализовать многоуровневую навигацию (буквы → слова → определения).
* Организовать просмотр определений в отдельных модальных окнах.

1. Дополнительные возможности:

* Обеспечить адаптивность интерфейса.
* Реализовать компиляцию в исполняемый файл для кроссплатформенного использования.

**Описание реализации**

1. **Используемые технологии.**

В проекте использовались следующие технологии:

* Python - основной язык разработки.
* Tkinter - библиотека для создания кроссплатформенного графического интерфейса.
* JSON - формат для хранения и организации словарных статей.
* PyInstaller - инструмент для сборки конечного приложения.

1. **Структура проекта.**

Приложение состоит из трех основных компонентов:

* **Главное окно:**
* Содержит панель с кнопками алфавита для навигации.
* Включает переключатель цветовых тем (светлая/тёмная).
* Выполняет роль центрального узла управления приложением.
* **Окно выбора слов:**
* Активируется при выборе конкретной буквы.
* Отображает прокручиваемый список соответствующих терминов.
* Реализована адаптивная верстка под различное количество элементов.
* **Окно словарной статьи:**
* Открывается для просмотра полного определения выбранного термина.
* Имеет четкую визуальную иерархию информации.

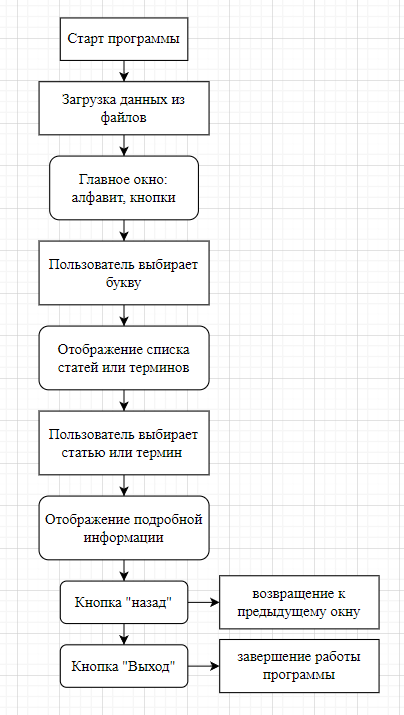
****

Рис 1.

1. **Особенности реализации.**

* Система хранения данных:
* Словарная база организована в виде набора JSON-файлов.
* Каждой букве алфавита соответствует отдельный файл данных.
* Используется человекочитаемый формат хранения информации.
* Пользовательский интерфейс:
* Интуитивная навигация между уровнями информации.
* Единообразное оформление всех элементов:
* Кнопки букв центрированы
* Элементы списка слов выровнены по левому краю
* Адаптивные размеры интерактивных элементов
* Реализована поддержка прокрутки для объемных списков
* Система тем оформления:
* Механизм переключения между светлой и тёмной палитрами.
* Единая точка управления цветовой схемой через глобальные переменные.
* Автоматическое применение изменений ко всем элементам интерфейса.
* Сохранение выбранной темы между сеансами работы.

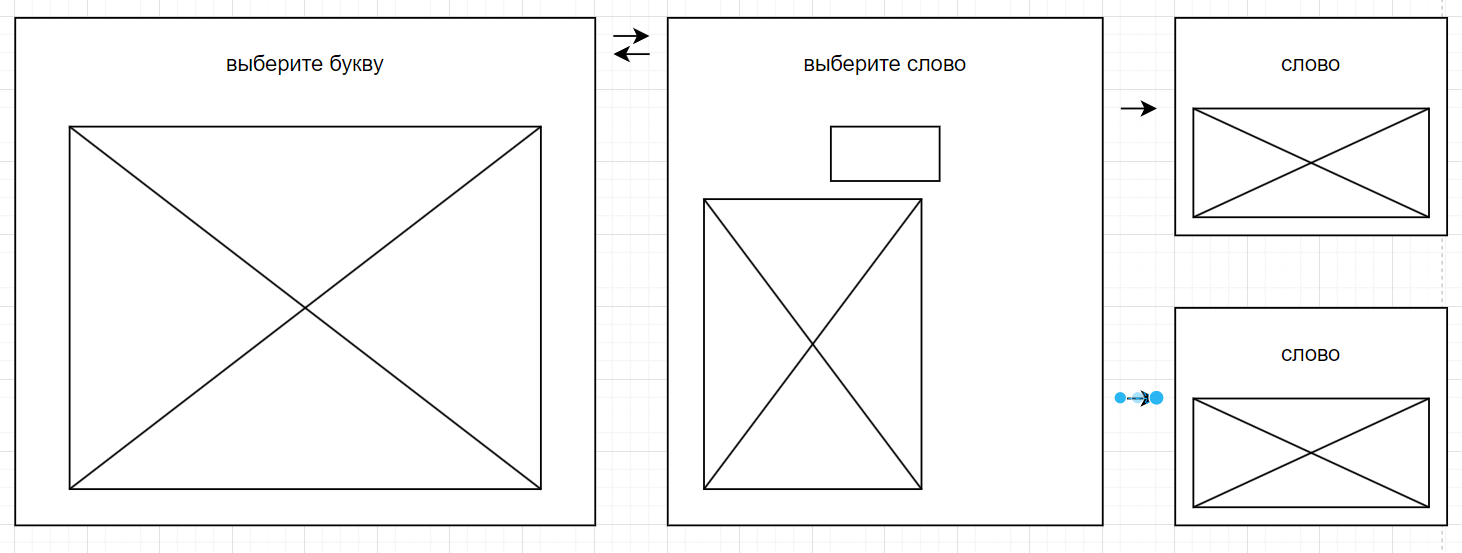
****

Рис 2.

**Трудности и их решение.**

Реализация прокрутки списка слов

* Проблема: При большом количестве слов в разделе содержимое не умещалось в окне стандартного размера, что делало навигацию неудобной.
* Решение: Была разработана система прокрутки с использованием связки Canvas и Scrollbar.

Система смены тем оформления

* Проблема: Требовалось обеспечить синхронное изменение цветовой схемы всех элементов интерфейса при переключении между темами.
* Решение: Создана функция apply\_theme() для управления цветовыми параметрами.

Компиляция в исполняемый файл

* Проблема: После сборки приложения возникали ошибки доступа к файлам данных (JSON), так как исполняемый файл искал их по некорректным путям.
* Решение: Реализована система проверки доступности файлов данных.

**Результаты**

В результате разработки было создано полнофункциональное настольное приложение – интерактивная энциклопедия, соответствующее всем поставленным требованиям.

Реализованы следующие задачи:

* Удобная трехуровневая навигация: алфавит → слова → определения.
* Система просмотра словарных статей в отдельных модальных окнах.
* Механизм смены цветовых тем (светлая/тёмная) с мгновенным применением изменений.
* Поддержка прокрутки для работы с объемными списками терминов.

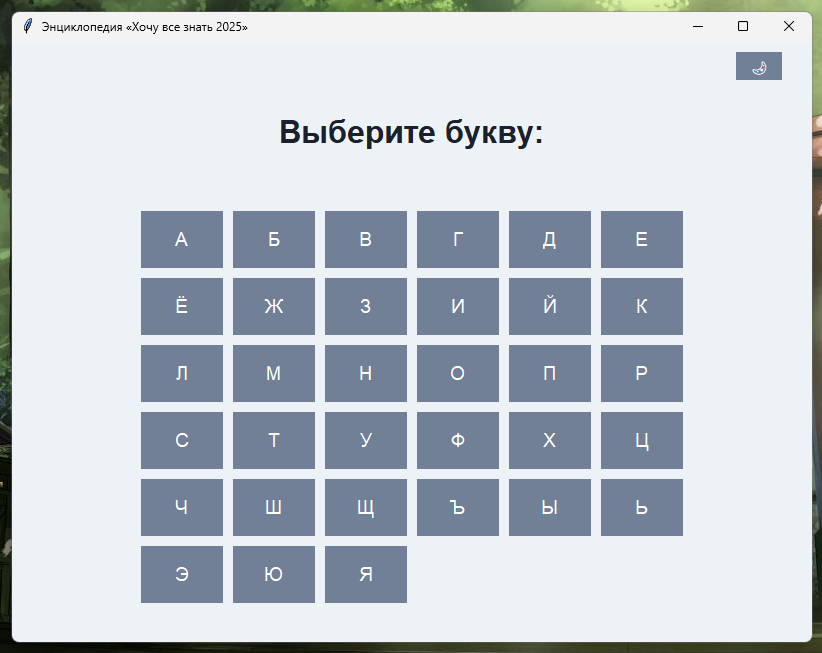


Рис 3.

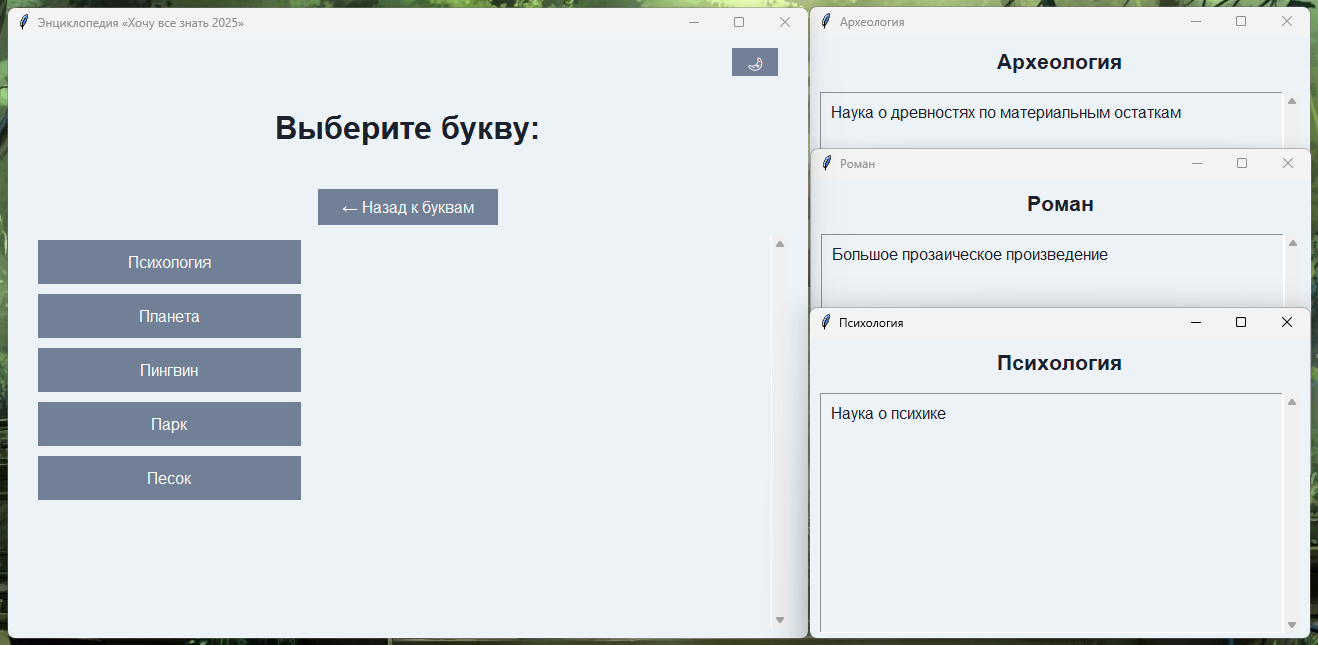


Рис 4.

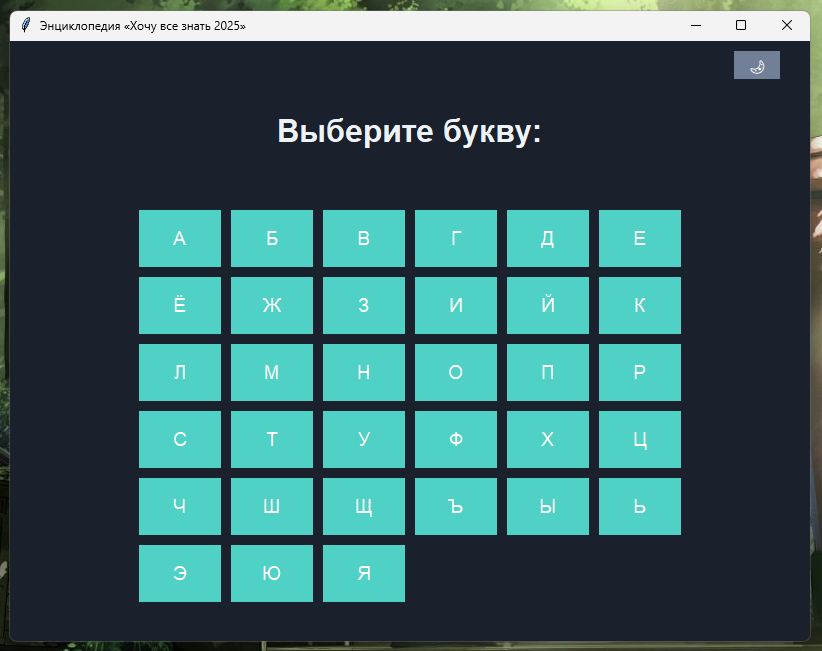


Рис 5.

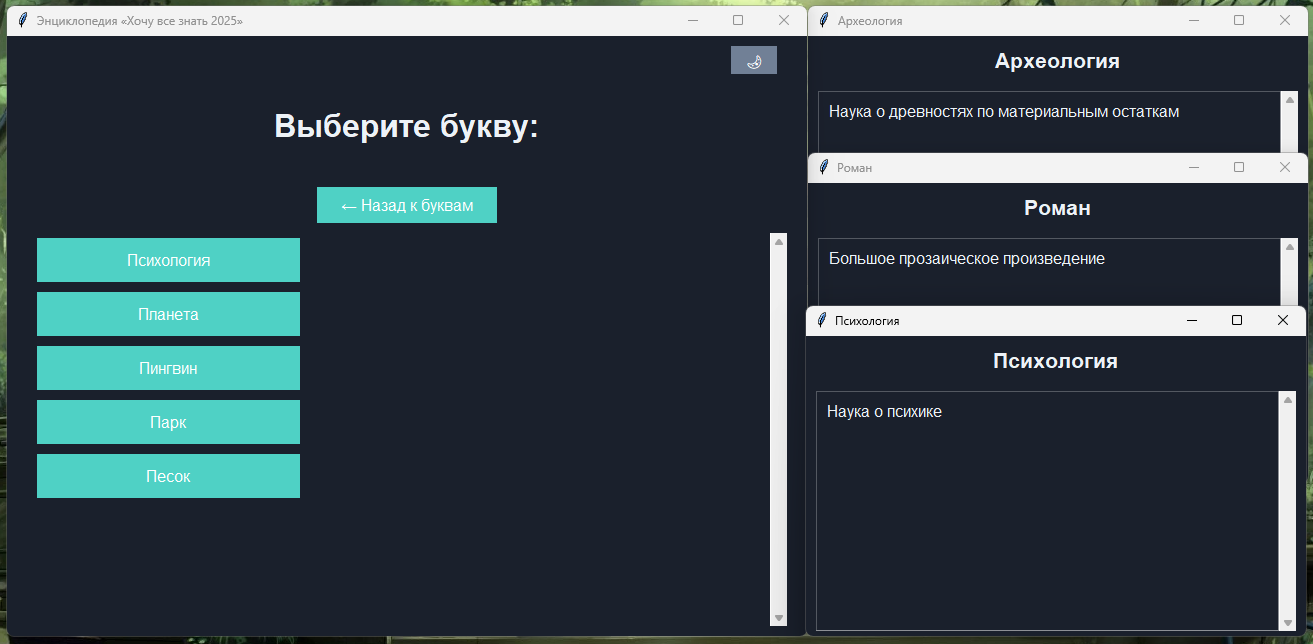


Рис 6.

**Выводы**

Основные достижения проекта:

1. Получен практический опыт работы с Tkinter:

* Создание многооконного интерфейса.
* Управление виджетами и обработка событий.
* Динамическое обновление элементов GUI.

1. Освоены ключевые технологии:

* Организация структурированного хранения данных в JSON-формате.
* Реализация прокрутки и адаптивного интерфейса.
* Управление цветовыми схемами приложения.
* Компиляция Python-приложений в исполняемые файлы.

Перспективные направления для улучшения:

1. Расширение функциональности:

* Внедрение системы полнотекстового поиска по словам и определениям.
* Добавление возможности редактирования и дополнения энциклопедии.

1. Технологические улучшения:

* Переход на систему хранения данных в SQLite или другой БД.
* Реализация системы резервного копирования словарной базы.

1. Визуальные усовершенствования:

* Добавление анимации переходов между окнами.
* Разработка современного набора иконок.
* Создание дополнительных тем оформления.

Проект успешно выполнил свою учебную задачу, заложив основу для дальнейшего развития в полноценное приложение с расширенным функционалом.