Федеральное государственное автономное образовательное учреждение высшего образования

«Пермский государственный национальный исследовательский университет»

**ДОКУМЕНТАЦИЯ**

по разработке проектных заданий для практики студентов

по теме

«**Умный словарь терминов БАС**»

в рамках реализации программы стратегического академического лидерства «Приоритет-2030» в ПГНИУ в 2023 году

Пермь, 2023 г.

**1.** **Введение**

1.1. Обзор программы

1.2. Назначение и основные характеристики

**2.** **Руководство пользователя**

2.1. Инструкциии по установке

2.2. Использование программы

**3.** **Техническое описание**

3.1. Внутреннее устройство программы

3.2. Используемые алгоритмы и структуры данных

**4.** **Руководство по разработке**

4.1. Инструкции для разработчиков

**5.** **Отчет**

5.1. Цели программы

5.2. Задачи

5.3. План и сроки

5.4. Предполагаемые риски и способы борьбы с ними

**6.** **Описание продукта**

6.1. Функциональные возможности

6.2. Способы тестирования

6.3. Скриншоты

**7.** **Разработчики**

**8.** **Ссылки**

1. ВВЕДЕНИЕ

1.1. Обзор программы

Программа "Умный словарь" предназначена для стандартизации терминологии в области беспилотных авиационных систем (БАС). Она помогает улучшить понимание и взаимодействие между специалистами, работающими в данной сфере, создавая единый и точный словарь ключевых терминов.

1.2. Назначение и основные характеристики

Программа автоматизирует определение и стандартизацию терминов в тексте, позволяет добавлять новые термины с их определениями, искать термины по словарю и предоставляет возможность экспорта и импорта словаря.

2. РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ

2.1. Инструкции по установке

·Скачайте проект с гитхаба.

·Выполните команду python manage.py runserver из директории, где находится файл manage.py

·После этого откройте сайт локально на адресе 127.0.0.1:8000

2.2. Использование программы

В окне поиска введите термин, описание термина или аббревиатуру и нажмите на иконку лупы

3. ТЕХНИЧЕСКОЕ ОПИСАНИЕ

3.1. Внутреннее устройство программы

Программа построена на основе клиент-серверной архитектуры. Клиентская часть отвечает за пользовательский интерфейс, а серверная часть обрабатывает запросы на добавление, поиск и стандартизацию терминов.

3.2. Используемые алгоритмы и структуры данных

· Алгоритмы поиска: Для поиска терминов используется библиотека Spacy.

· Структуры данных: Для хранения терминов и его составляющих используется база данных SQLite.

4. РУКОВОДСТВО ПО РАЗРАБОТКЕ

4.1. Инструкции для разработчиков

1. Исходный код программы написан на языке Python.

2. Для расширения функциональности необходимо соблюдать следующие шаги:

· Создать новую ветку в системе управления версиями.

· Внести изменения в код, следуя стандартам кодирования.

· Провести тестирование изменений.

· Сделать pull request для ревью кода.

3.Все изменения должны быть хорошо документированы и прокомментированы.

5. ОТЧЁТ

5.1. Цели программы

Цель программы заключается в создании единого словаря для стандартизации терминологии в области бизнес-аналитики, что улучшит коммуникацию и качество аналитических решений.

5.2. Задачи

· Разработка функционала для автоматической стандартизации терминов.

· Обеспечение возможности добавления и редактирования терминов.

· Создание удобного интерфейса для поиска терминов.

5.3. План и сроки

Ноябрь 2023: Начало разработки, сбор требований.

Декабрь 2023: Разработка дизайна

Январь 2024: Начало реализации базового функционала.

Май 2024: Тестирование и выпуск программы.

5.4. Предполагаемые риски и способы борьбы с ними

Риск: Некорректная работа программы из-за ошибок в коде.

Способ борьбы: Регулярное тестирование и ревью кода.

Риск: Низкое принятие пользователями из-за сложного интерфейса.

Способ борьбы: Проведение UX-исследований и улучшение интерфейса.

Риск: Замедление работы программы при большом количестве терминов в базе данных .

Способ борьбы: Нагрузочное тестирование и оптимизация алгоритма поиска терминов.

6. ОПИСАНИЕ ПРОДУКТА

6.1. Функциональные возможности

Поиск терминов в словаре по названию, описанию и по аббревиатуре; возможность поиска терминов с опечатками в вводимых данных.

6.2. Способы тестирования

Функциональное тестирование: Проверка всех функций программы на соответствие требованиям.

Регрессионное тестирование: Обеспечение того, что новые изменения не нарушают существующий функционал.

Тестирование производительности: Проверка программы на нагрузку и время отклика.

6.3. Скриншоты

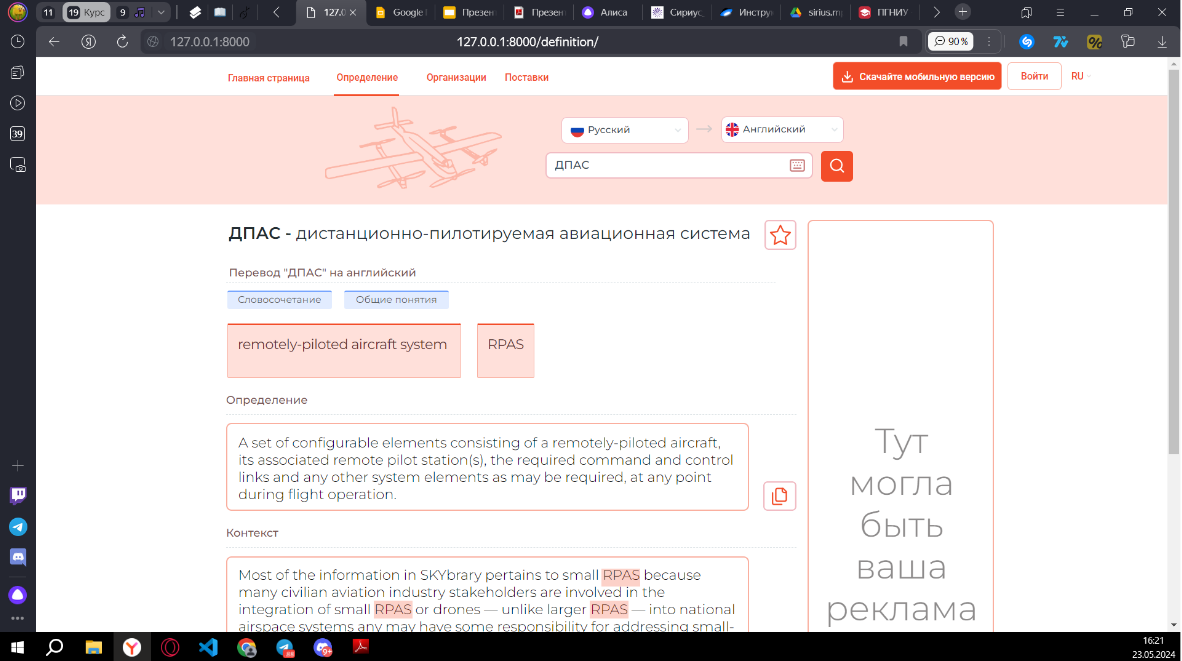


Рис. 1. Страница определения.

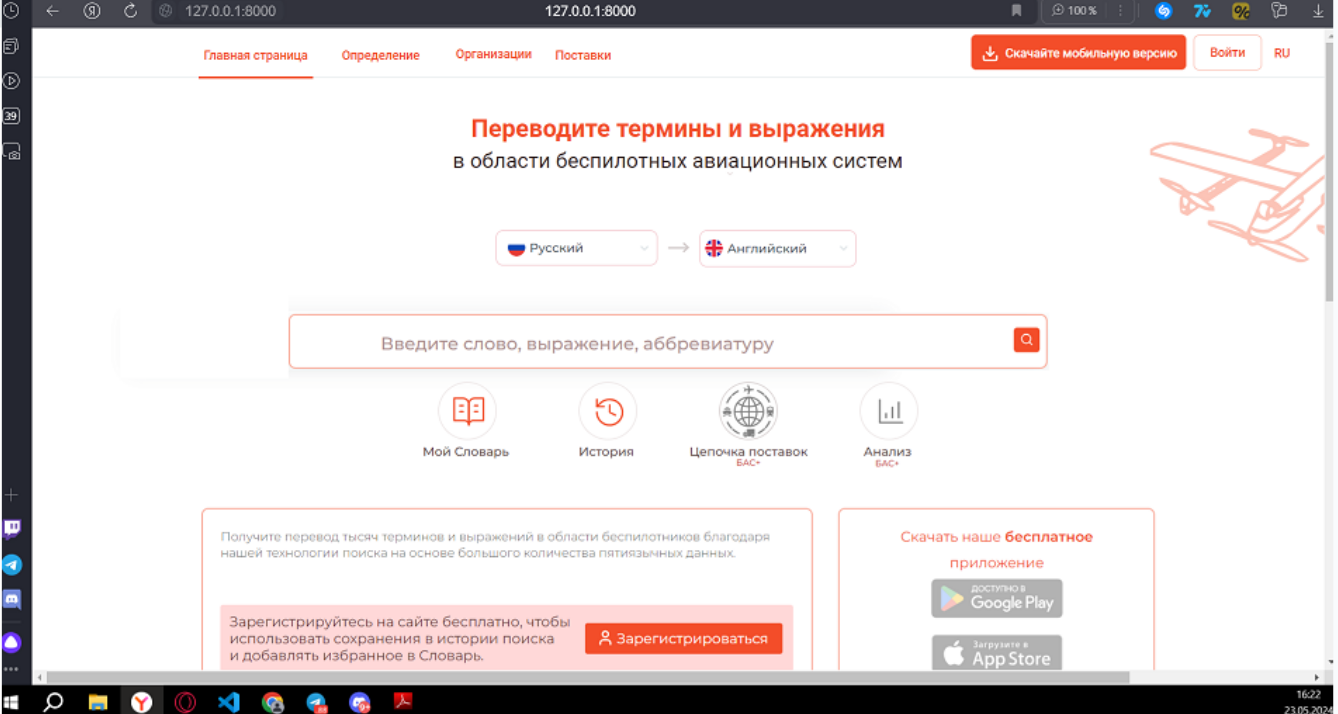


Рис. 2. Главная страница.

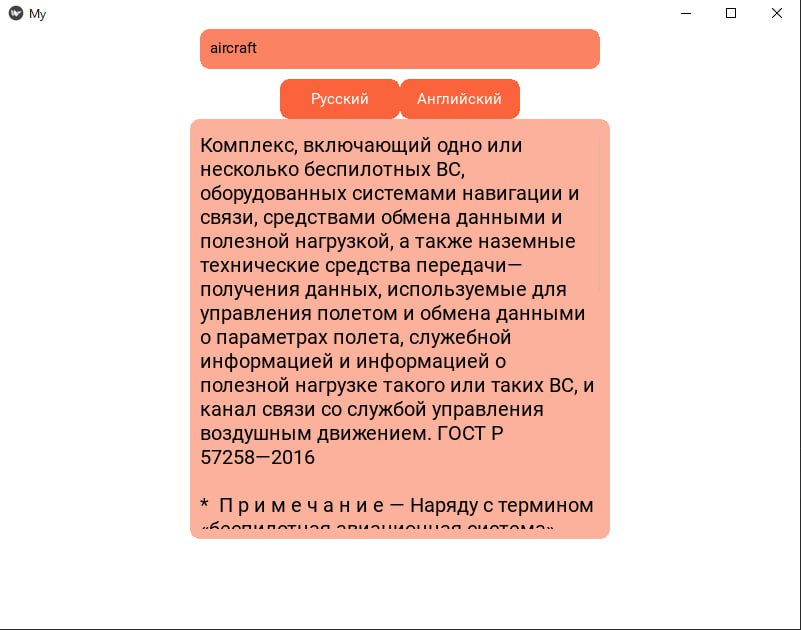


Рис. 3. Интерфейс мобильной версии.

7. РАЗРАБОТЧИКИ

1. Клименчук Илья: мобильный разработчик
2. Подавалов Глеб: back-end разработчик
3. Сахаутдинов Данил: разработчик интеллектуального поиска
4. Димиев Андрей: front-end разработчик
5. Гуляев Вадим: дизайнер сайта
6. Жуйков Артём: тестировщик

8. ССЫЛКИ

[Репозиторий на GitHub](https://github.com/Reez0n/UAS)