Учреждение образования

«БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Кафедра интеллектуальных информационных технологий

Отчёт

# по курсу «Естественно-языковой интерфейс интеллектуальных систем»

Лабораторная работа №1

«Разработка автоматизированной системы формирования словаря естественного языка»

|  |  |
| --- | --- |
| Выполнили:  студенты гр. 221701: | Крупский А. В. Глёза Е. Д. |
| Проверил: | Крапивин Ю.Б. |

Минск, 2025

# Цель работы:

Освоить принципы разработки прикладных сервисных программ для решения задачи автоматического лексического и лексико-грамматического анализа текста естественного языка.

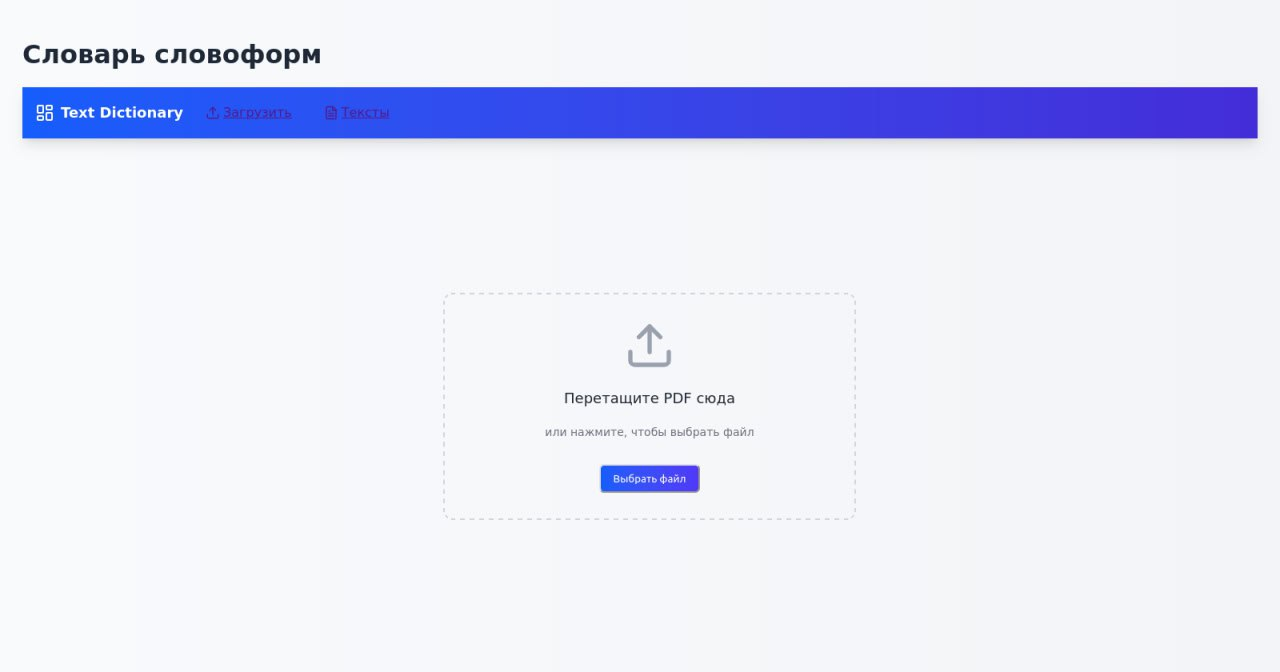
# Задание:

Список слов, упорядоченный по алфавиту и включающий как лексемы, так и словоформы, с указанием частоты встречаемости каждой из форм. Для словоформ пользователю должна быть предоставлена возможность вводить дополнительную морфологическую информацию, а именно, отнесение слова к соответствующей части речи, указание рода, числа, падежа и т.п. При этом морфологическая информация может быть оформлена как отдельная неформатированная запись, т.е. это просто текст, который пользователь может оформлять произвольным образом.

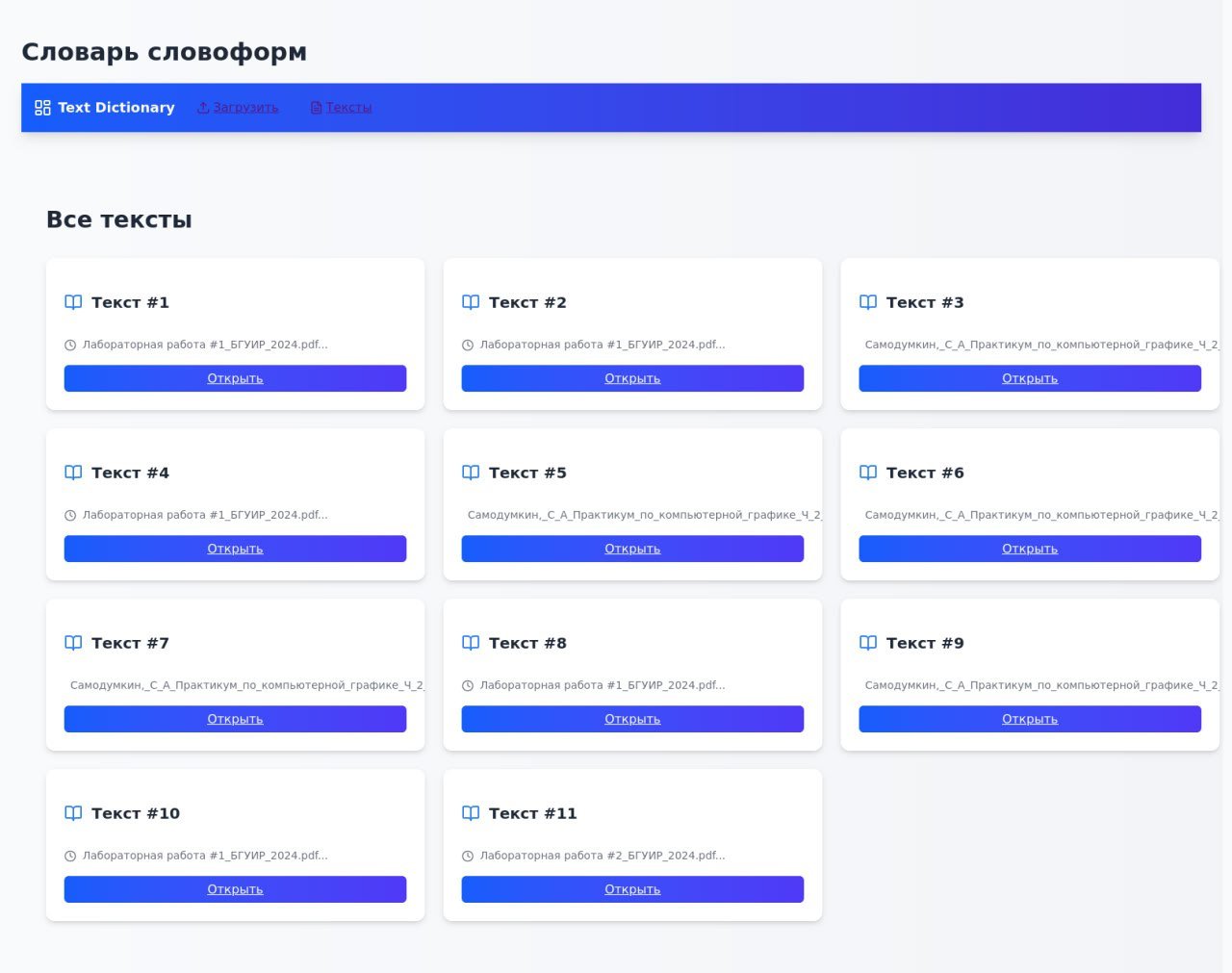
**Используемые бибилиотеки:**

1. **FastAPI**  
   **Описание:** Современный высокопроизводительный веб-фреймворк для создания API на Python, основанный на типизации.  
   **Использование:** Разработка RESTful API, асинхронных веб-приложений и микросервисов с автоматической документацией (Swagger, ReDoc).
2. **Natasha**  
   **Описание:** Библиотека для обработки естественного языка (NLP) на русском языке.  
   **Использование:** Разбор и анализ текстов на русском языке: токенизация, лемматизация, определение частей речи, извлечение именованных сущностей.
3. **setuptools**  
   **Описание:** Стандартный инструмент для сборки, упаковки и распространения Python-пакетов.  
   **Использование:** Создание setup-файлов для установки библиотек, управление зависимостями, публикация на PyPI.
4. **SQLAlchemy**  
   **Описание:** Библиотека для работы с базами данных в Python, реализующая ORM (Object-Relational Mapping) и Core-интерфейс для SQL-запросов.  
   **Использование:** Определение моделей данных, выполнение SQL-запросов, взаимодействие с различными СУБД через единый интерфейс.
5. **os**  
   **Описание**: Стандартная библиотека Python для взаимодействия с операционной системой.  
   **Использование**: Проверка существования файлов, работа с путями и файловой системой.
6. **sys**  
   **Описание**: Стандартная библиотека Python для взаимодействия с интерпретатором Python.  
   **Использование**: Обработка аргументов командной строки и управление выполнением программы.
7. **PyCryptodome**  
   **Описание:** Криптографическая библиотека Python, совместимая с PyCrypto, но с активной поддержкой и улучшенной безопасностью.  
   **Использование:** Шифрование и дешифрование данных, хэш-функции, работа с RSA, AES, генерация ключей.
8. **PyPDF2**  
   **Описание:** Библиотека Python для работы с PDF-документами.  
   **Использование:** Чтение, извлечение текста, объединение, разбиение, поворот страниц и метаданные PDF-файлов.

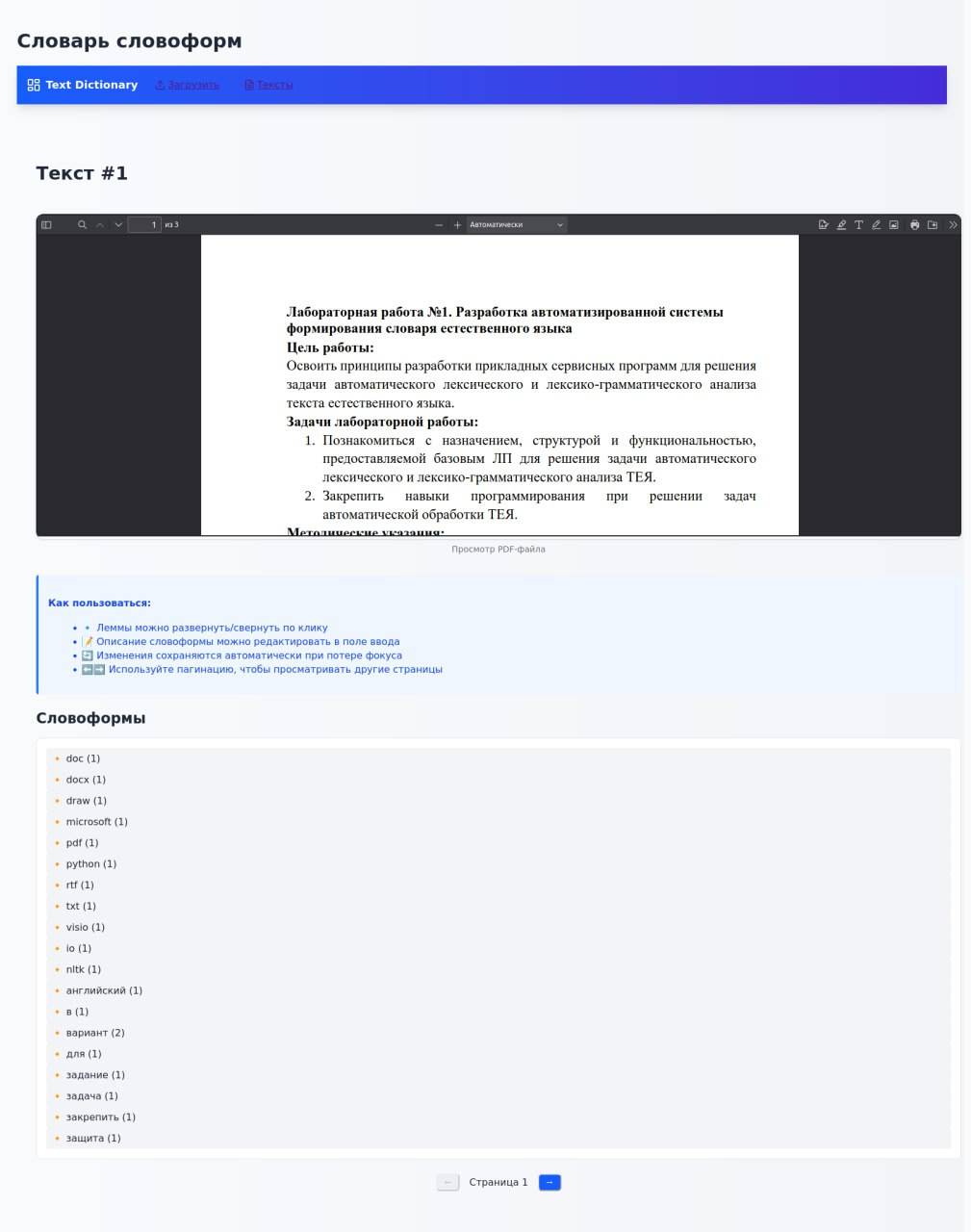
**Интерфейс:**



Страница загрузки файла

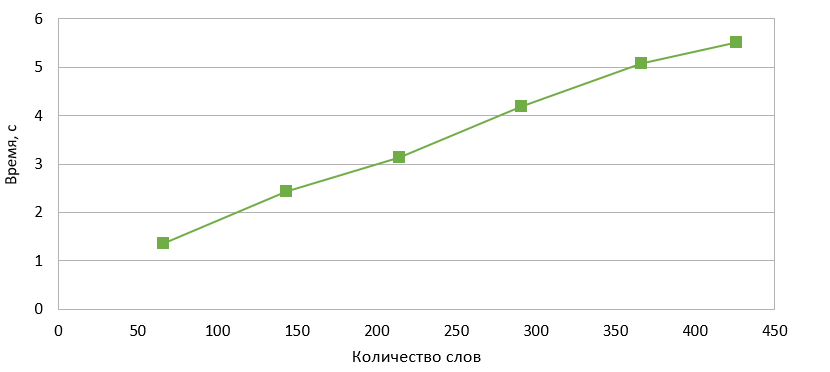


Страница выбора текста

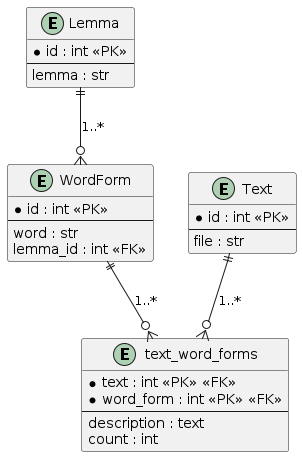


Страница просмотра текста

# Тест производительности

****

**Структура хранения данных**

******

**Структурная схема приложения**



## Описание алгоритмов:

#### Обработка текста и построение словаря лексем

1. **Начало**
2. Получаем путь к файлу документа (.pdf).
3. Проверяем существование файла и его формат:
4. Очищаем текст: заменяем непечатаемые символы пробелами, убираем лишние пробелы.
5. Если текст пустой, возвращаем ошибку.
6. Создаем пустой словарь лексем.
7. Используем библиотеку Natasha для анализа текста:
   * Сегментируем текст на токены.
   * Применяем морфологический анализ для определения частей речи.
   * Лемматизируем токены, получая нормализованные формы слов.
8. Фильтруем токены, исключая стоп-слова, пунктуацию и некорректные леммы.
9. Для каждого токена находим или создаем лексему.
10. Сохраняем словоформы в базе данных.
11. **Конец**

#### Добавление описания к словоформе

1. **Начало**
2. Проверяем, что словарь лексем не пуст. Если пуст, показываем ошибку и завершаем.
3. Выбираем лексему.
4. Выбираем словоформу к заданной лексеме.
5. Заполняем поле описания.
6. **Конец**

#### Просмотр сохраненных текстов

1. **Начало**
2. Пользователь кликает на вкладку “тексты”.
3. Пользователь выбирает текст из списка текстов.
4. Выбранный текст загружается из базы данных и отображается.
5. **Конец**

## Вывод:

Данная программа представляет собой инструмент для создания списка лексем и словоформ. Она также предоставляет возможность пользователю добавлять морфологическую информацию к словоформам. Данная программа имеет перспективы в области анализа текстов и книг для создания аналитики по различным лексемам.