Московский государственный технический университет им. Н.Э. Баумана

Динамическое документирование научно-образовательной деятельности

Студент группы РК6-81б Петричук А.О.
Научный руководитель: Соколов А.П.

15.06.2023

СОДЕРЖАНИЕ ДОКЛАДА

ОСНОВНЫЕ ОПРЕДЕЛЕНИЯ

НАУЧНО-ИССЛЕДОВАТЕЛЬСКАЯ ДЕЯТЕЛЬНОСТЬ (НИД)

Это работа научного характера, связанная с научным поиском, проведением исследований, экспериментами в целях расширения имеющихся и получения новых знаний, проверки научных гипотез, установления закономерностей, проявляющихся в природе и в обществе, научных обобщений, научного обоснования проектов.

ОБРАЗОВАТЕЛЬНАЯ ДЕЯТЕЛЬНОСТЬ (ОД)

Процесс целенаправленного, педагогически обоснованного, последовательного взаимодействия субъектов образования, в ходе которого решаются задачи обучения, развития и воспитания личности.

ОСНОВНЫЕ СОСТАВЛЯЮЩИЕ НИД И ОД

НИД	ОД
Поиск информации об объекте исследований, обзор литературы (статей, патентов и пр.)	Проведение учебных мероприятий для студентов
Проведение вычислительных экспериментов	Работа студентов над заданиями (постановка задач, консультирование и т.п.)
Подготовка и проведение испытаний	Проверка работ преподавателем, выдача обратной связи каждому студенту (оценка или развернутая обратная связь);
Документирование проведенных исследований	Создание рейтинга студентов, его доступность каждому участнику процесса
Публикация работ в официальных изданиях	
Разработка предметного ПО	
Подготовка заявок на гранты, договоры и т.д.	
Социальное взаимодействие исследователей друг с другом	

ИССЛЕДОВАННЫЕ СИСТЕМЫ

Более подробный анализ приведен в разделе ВВЕДЕНИЕ и в приложениях А и Б

Системы Управления Записями (Record Management System)

- Zotero
- EndNote
- Mendeley

- Каждый из пунктов обладает собственным набором преимуществ
- EndNote – необходима покупка лицензии

Инструменты для редактирования текстовых форматов файлов

- Overleaf
- Authorea
- LaTex + git

- Каждый из пунктов поддерживает коллективную работу над документом
- Authorea – часть предоставляемых функций платные

Системы управления обучением (Learning Management System)

- Moodle
- Sakai
- ATutor
- Blackboard
- SuccessFactor
- SumTotal

- Последние три – на коммерческой основе
- Каждый из пунктов предоставляет разный набор возможностей

ОСНОВНЫЕ ОПРЕДЕЛЕНИЯ

ИССЛЕДОВАНИЕ

Задокументированный процесс изучения заранее определённой конкретной темы.

ИССЛЕДОВАТЕЛЬ

Пользователь ИС, осуществляющий исследовательскую деятельность.

ЗАМЕТКА

Краткая запись о проделанной работе в рамках конкретного исследования.

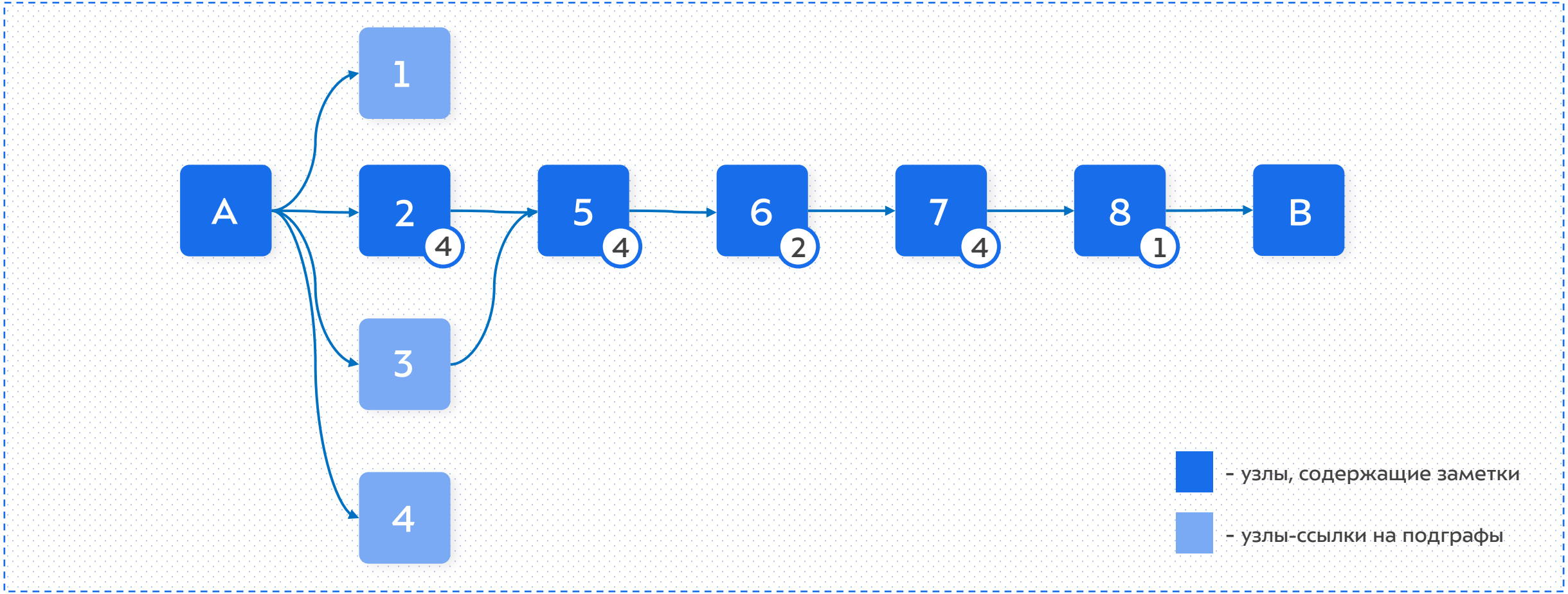
ГРАФ

Граф G – совокупность множества точек (вершин или узлов) $\{x_i\}_1^n$ и множества линий (далее рёбер) $\{a_j\}_1^m$, соединяющих между собой все или часть этих точек.

ОРГРАФ

Если рёбра графа G ориентированы (т.е. имеют направление, отмеченное стрелкой), то такой граф называется ориентированным (орграфом).

- A – начало
- 1-4 – обзор существующих инструментов
- 5 – постановка задачи
- 6 – проектирование архитектуры ИС
- 7 – разработка ИС
- 8 – модуль Latex2HTML
- B – итог работы



РОЛИ

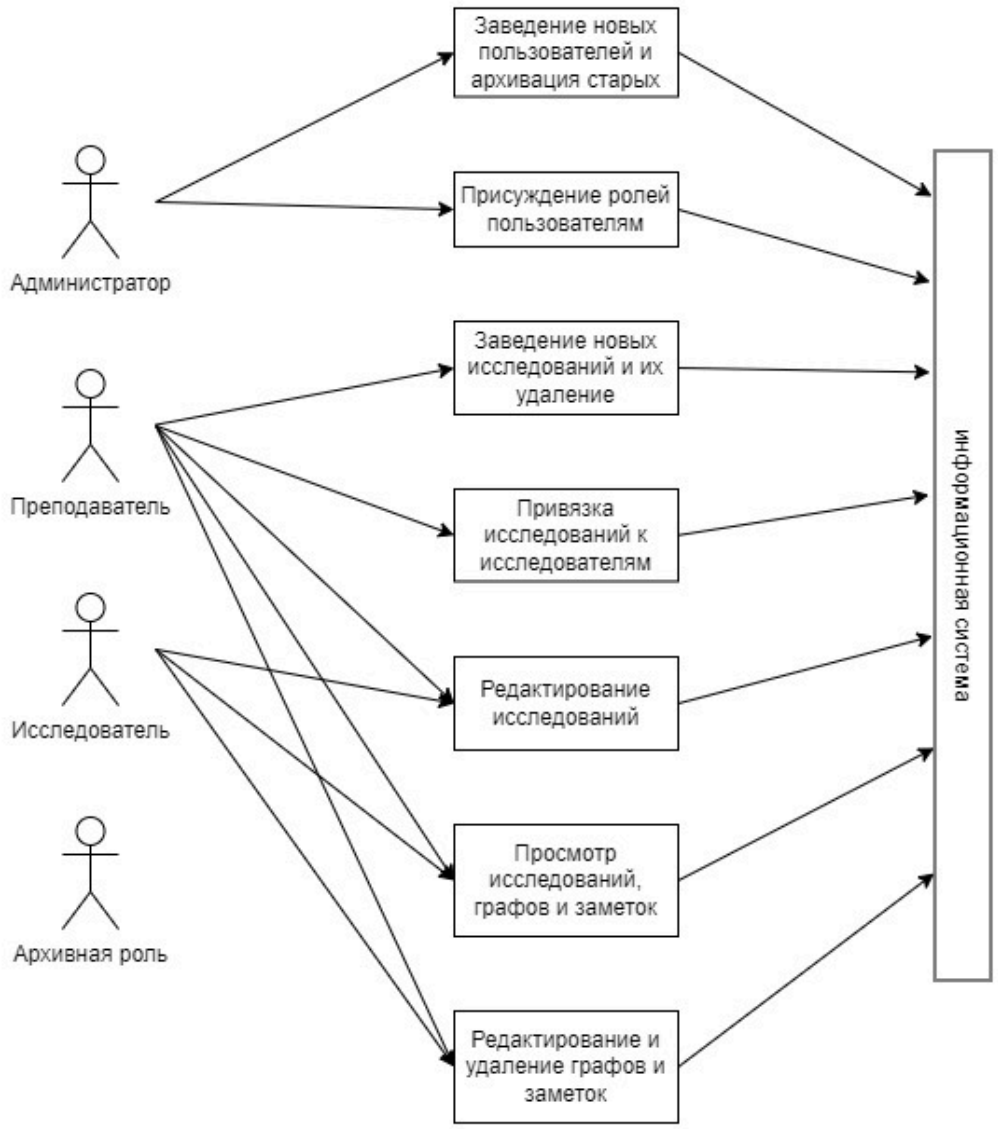
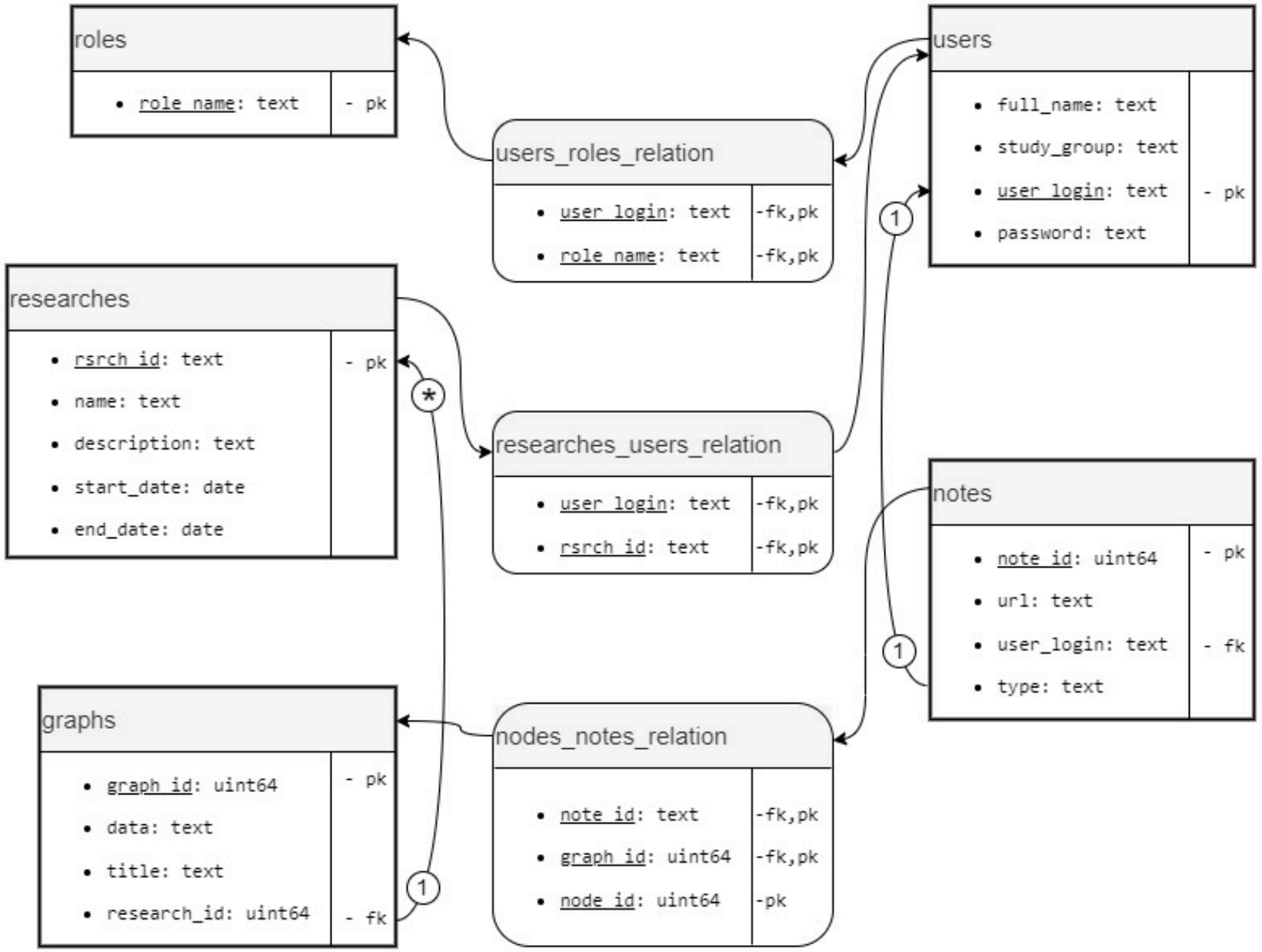

Заведение новых пользователей и архивация старых
Присуждение ролей пользователям
Администратор
Заведение новых исследований и их удаление
Преподаватель
Привязка исследований к исследователям
Редактирование исследований
Исследователь
Просмотр исследований, графов и заметок
Архивная роль
Редактирование и удаление графов и заметок
информационная система

СХЕМА БАЗЫ ДАННЫХ



СПРОЕКТИРОВАННЫЙ СПИСОК МЕТОДОВ API

Auth

- POST login
- POST logout
- POST logout всех сессий

Note

- GET список заметок
- GET детальная инф-я
- POST создание
- DELETE удаление

Graph

- GET детальная инф-я
- PATCH редактирование атрибутов
- DELETE удаление
- POST создание

User

- GET данные авторизованного пользователя
- GET список исследователей
- GET подсказки для автозаполнения

Research

- GET список исследований
- GET детальная инф-я
- PATCH редактирование атрибутов
- DELETE удаление
- POST создание

Node

- GET детальная инф-я

ЧТО БЫЛО СДЕЛАНО

- `Graph.data` – текстовое поле, которое хранит данные о графе в формате языка DOT.
- `Graph._dot` – хранит объект pydot.Dot. Этот объект содержит в себе те же данные, что поле data, но в формате объекта, работать с которым удобнее.
- `User.computed_full_name` – поле, содержащее ФИО и вычисляемое на основе других полей. Используется для создания индекса, с помощью которого происходит поиск пользователя по поисковому запросу.

ЧТО БЫЛО СДЕЛАНО

Реализация аутентификации

выполнена на основе библиотеки DjangoRestKnox и работает на основе JWT токенов.

Реализация авторизации

выполнена с помощью запрограммированных классов разрешений:

- `IsOwnerObjectOrIsProfessorOrReadOnly` – объект «принадлежит» пользователю, либо пользователь имеет роль Преподаватель. Иначе только чтение;
- `IsProfessorOrReadOnly` – пользователь имеет роль Преподаватель. Иначе только чтение.

ЧТО БЫЛО СДЕЛАНО

Реализованы методы

- Валидации графа
  - Проверка наличия начального и конечного узлов
  - Проверка отсутствия объявления повторения узлов
  - Проверка отсутствия циклов
  - Проверка целостности графа
  - Проверка наличия всех объявленных вершин в структуре
- Представления данных из формата DOT в формат levels
- Редактирования метаданных узла
- Редактирования структуры графа
- Проверки существования узла

Пример простого графа в представлении DOT

```
digraph {
    A;
    1;
    B [title=SomeNode];
    A -> 1;
    1 -> B;
}
```

Пример объекта levels

```
{
    0: {'A': []},
    1: {'1': ['A']},
    2: {'B': ['1']},
},
```

ЧТО БЫЛО СДЕЛАНО

1. Для каждого из методов спроектированы тестовые сценарии.

2. Написаны юнит-тесты.

Подробнее процесс проектирования тестовых сценариев описан в главе 3 в разделе «Юнит-тестирование методов работы с данными графа»

ЧТО БЫЛО СДЕЛАНО

- Классы моделей для каждой из таблиц БД 1
- Модуль аутентификации и авторизации 2
- Методы работы с данными графа в формате DOT 3
- Юнит-тесты для методов работы с данными графа 4
- **Модуль-клиент для API Gitlab** 5

Атрибуты заметки

можно получить, распарсив её URL.

Класс GLClient

реализован и имеет метод get_note_raw_text_by_url, который получает на вход url и возвращает текст заметки и формат ее файла.

ОБЗОР

Основные трудности разбора синтаксиса LaTex

1. TeX – полный по Тьюрингу, следовательно практически невозможно провести разбор LaTeX ничем, кроме самого LaTeX.
2. Большинство написанных конвертеров написаны давно и устарели. Они «пытаются» разбирать уравнения (что сложно) и быть универсальными (что невозможно из-за п. 1).

Проанализированы инструменты

- Doconce
- Quarto

Для преобразования требуется весь проект, включая преамбулу и настройки пакетов

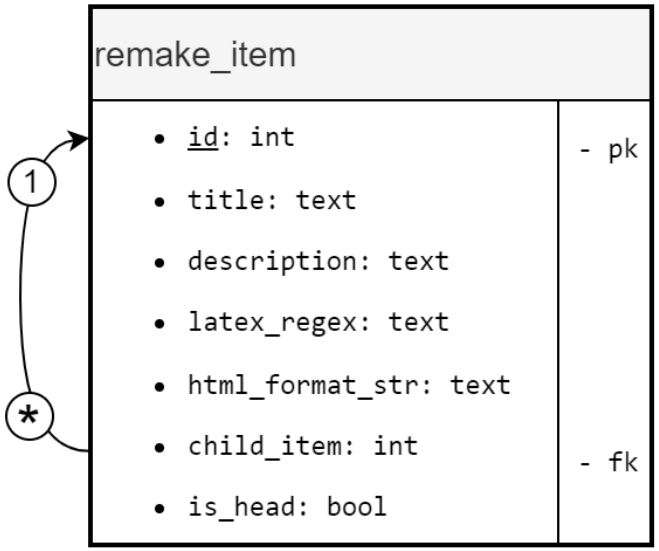
- MathJax

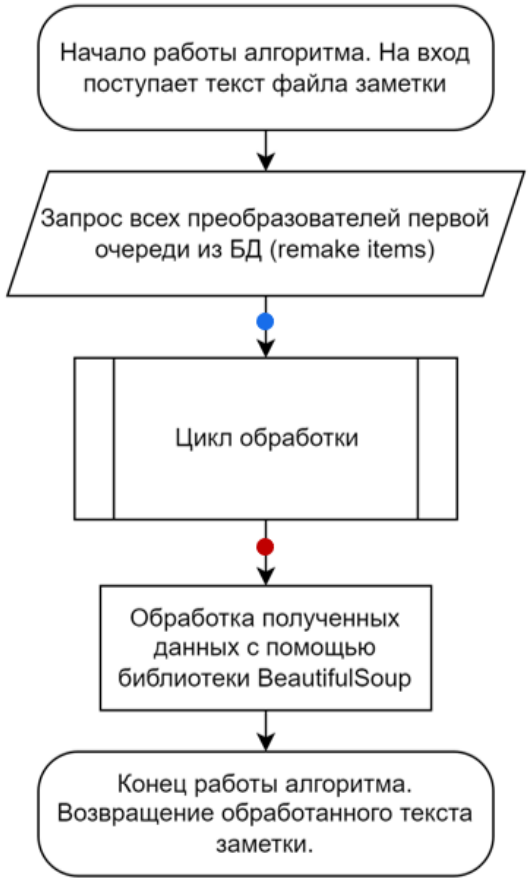
- Инструмент преобразования уравнений LaTeX в HTML
- Работает на стороне клиента
- Поддерживает все основные современные браузеры
- Не требует сложной настройки

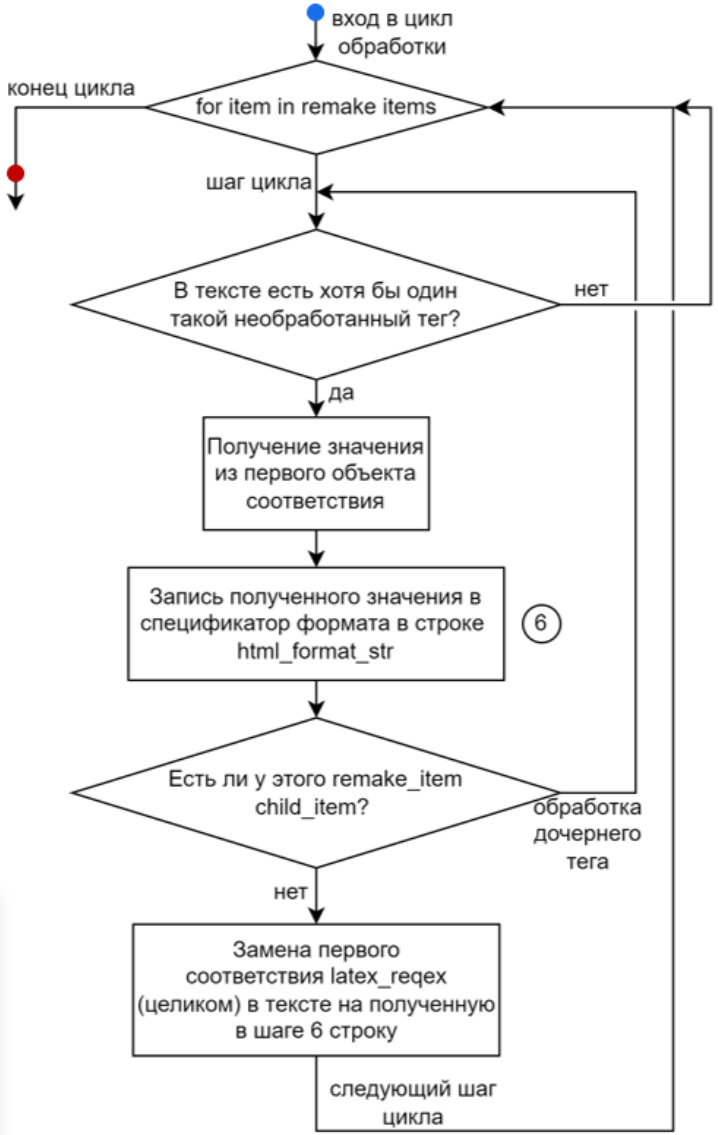
Подходы модуля Latex2HTML

- Не разбирает уравнения, предоставляя это MathJax
- Не претендует на звание универсального инструмента
- Множество ЕП расширяемо и изменяемо
- Может накладывать дополнительные ограничения на синтаксис текста заметок
- Накладывает ограничения на предоставляемую функциональность некоторых тегов

РАЗРАБОТКА







- GET запрос всех ЕП
- POST создание
- PATCH обновление атрибутов
- DELETE удаление

$$O\left(\sum_{i=1}^{n} m_i\right),$$

где n – кол-во ЕП в БД, m_i - кол-во встреченных в тексте единиц, подходящих под i-ое регулярное выражение

ПРИМЕР ЕДИНИЦ ПРЕОБРАЗОВАТЕЛЯ

ИСХОДНЫЙ ТЕКСТ	РЕГУЛЯРНОЕ ВЫРАЖЕНИЕ	СТРОКА ФОРМАТА	ИТОГОВАЯ СТРОКА
\underline{Подчеркнутый текст}	\\underline{(.*)}	<u>%s</u>	<u>Подчеркнутый текст</u>
\gls{URL}	\\gls{(URL)}	%s	URL
\begin{enumerate} \item Один; \item Два; \item Три. \end{enumerate}	\\begin{enumerate}([\s\S]*?)\\end{enumerate} + \\item\s*(.*)	%s + %s	 - Один; - Два; - Три.

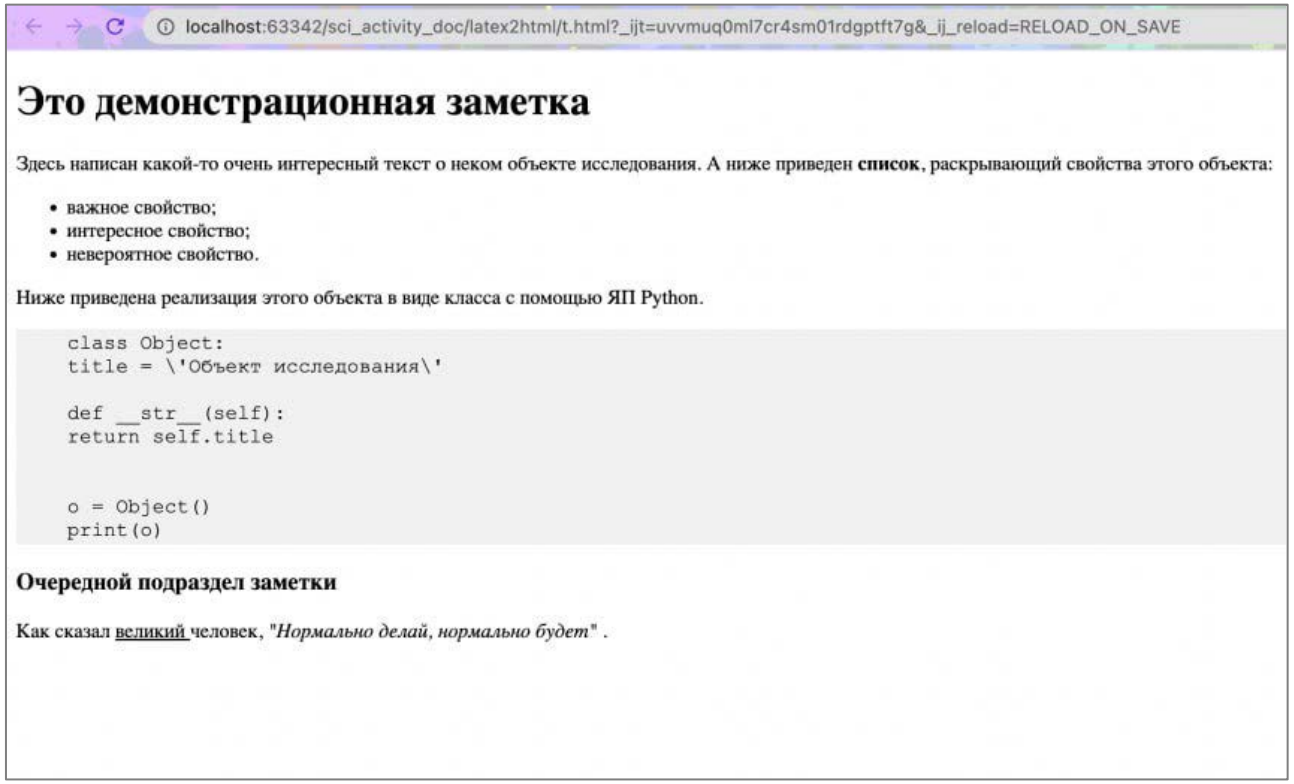
Отправляемый запрос

```
curl ––location 'http://0.0.0.0:8000/api/note/3' \
––header 'Authorization: Token 688080c6238e69e42987cff1e8c5476d17948bb4987ab0afc9a2dc915483735a' \
––data ''
```

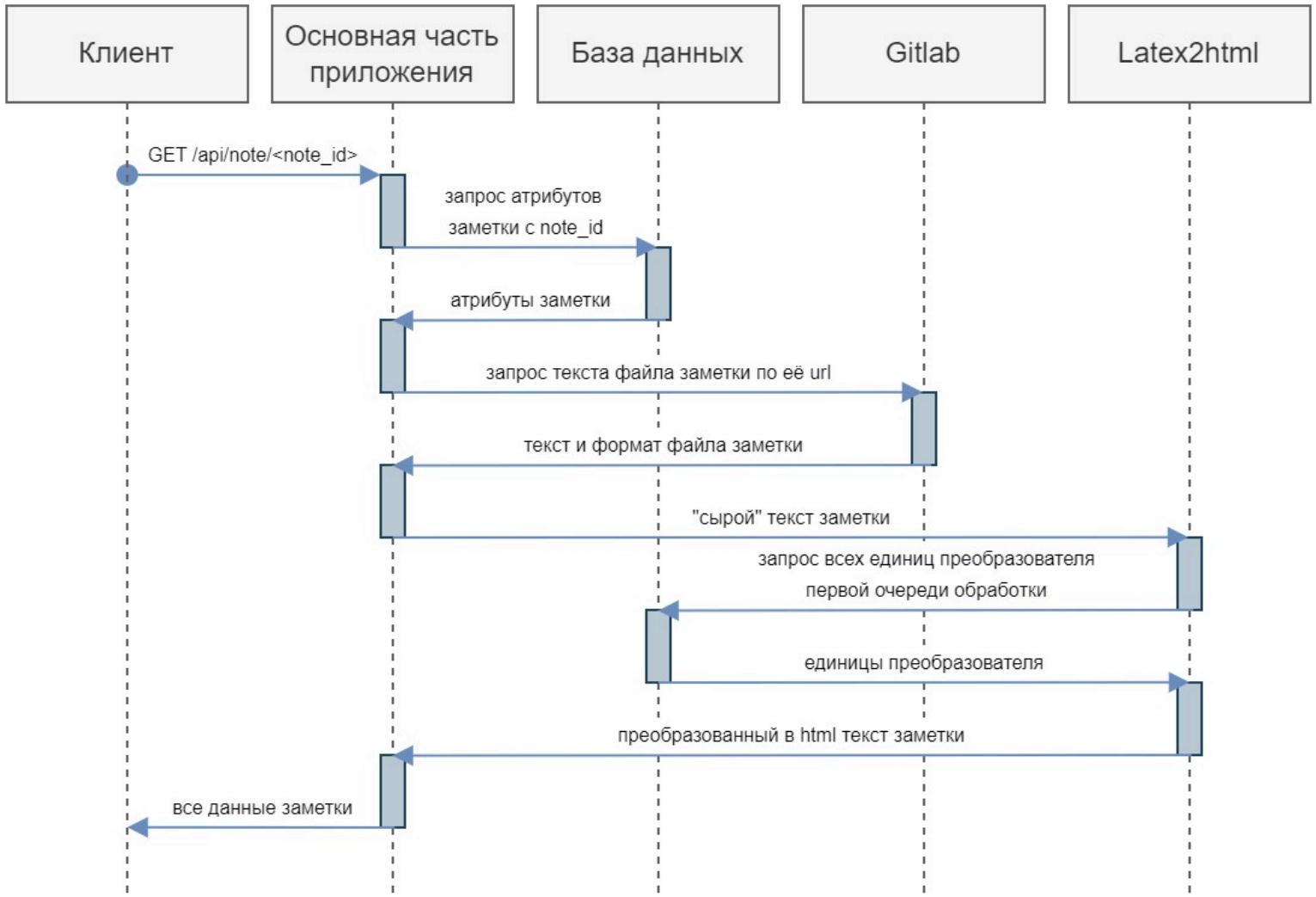
Исходный текст файла заметки в LaTeX

```
\notestatement{rndcsedoc}{Это демонстрационная заметка}

Здесь написан какой-то очень интересный текст о неком объекте исследования.
А ниже приведен \textbf{список}, раскрывающий свойства этого объекта:

\begin{itemize}
    \item важное свойство;
    \item интересное свойство;
    \item невероятное свойство.
\end{itemize}

Ниже приведена реализация этого объекта в виде класса с помощью \gls{PL} Python.

\begin{lstlisting}[language=Python]
    class Object:
        title = 'Объект исследования'

        def __str__(self):
            return self.title

    o = Object()
    print(o)
\end{lstlisting}


\subsection{Очередной подраздел заметки}

Как сказал \underline{великий} человек, \textit{<<Нормально делай, нормально будет>>}.
```

Отображение преобразованного текста в браузере



UML-диаграмма взаимодействия



Спасибо за внимание!

Контакты: petrichuk.nastya@yandex.ru

Исходный код проекта: