

КОНСТРУИРОВАНИЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Контрольная работа № 2 Синхронное межсервисное взаимодействие.

Дедлайн сдачи: 20 мая, 23:59

Формат сдачи КР определяется семинаристом. Задание выполняется на языке программирования являющимся основным для семинарских занятий (иной язык только по согласованию с семинаристом)

Предисловие

Преподавателям НИУ ВШЭ нужна ваша помощь!

В связи с большим количеством студенческих отчетов по письменным работам, сервис антиплагиат не справляется с нагрузкой. Вам поручено разработать инструмент для анализа отчетов на антиплагиат и их статистическую обработки.

Вся надежда на вас!

Ваш код — единственное оружие против цифрового хаоса.



В ходе общения с преподавателями выявлены следующие требования

Веб-приложение должно представлять функциональность «текстового сканера», который принимает на вход отчёт студента (в формате .txt) и выполняет его анализ, включая:

1. Подсчёт статистики: количество абзацев; слов; символов.
2. Сравнение файлов на схожесть (для выявления 100% плагиата среди ранее присланных отчетов).
3. Визуализация данных (облака слов) – дополнительно, не обязательное к выполнению.

Требуется: разработать микросервисную архитектуру серверной части веб-приложения

Архитектурный комитет предлагает следующее решение с четким разделением ответственности микросервисов (вы можете предложить свой вариант):

1. API Gateway – отвечает только за routing запросов
2. File Storing Service – отвечает только за хранение и выдачу файлов
3. File Analysis Service – отвечает только за проведение анализа, хранение результатов и их выдачу

Критерии оценки

1. Реализация основных требований к функциональности – **2 балла**
2. Распределение функциональности по микросервисам – **4 балла**
 - а. Серверная часть веб-приложения состоит минимум из двух микросервисов.
 - б. Обработка ошибок (один из микросервисов упал).
3. Реализация коллекции Postman / Swagger, которая должна демонстрировать функциональность реализованных микросервисов, охватывая все API – **1 балл**
4. Качество кода и документация – **2 балла**
 - а. Хорошо организованный, модульный и “чистый” код.
 - б. Документация, включая краткое описание архитектуры системы и спецификацию API.
5. Покрыто тестами более 65% кода (**1 балл**)

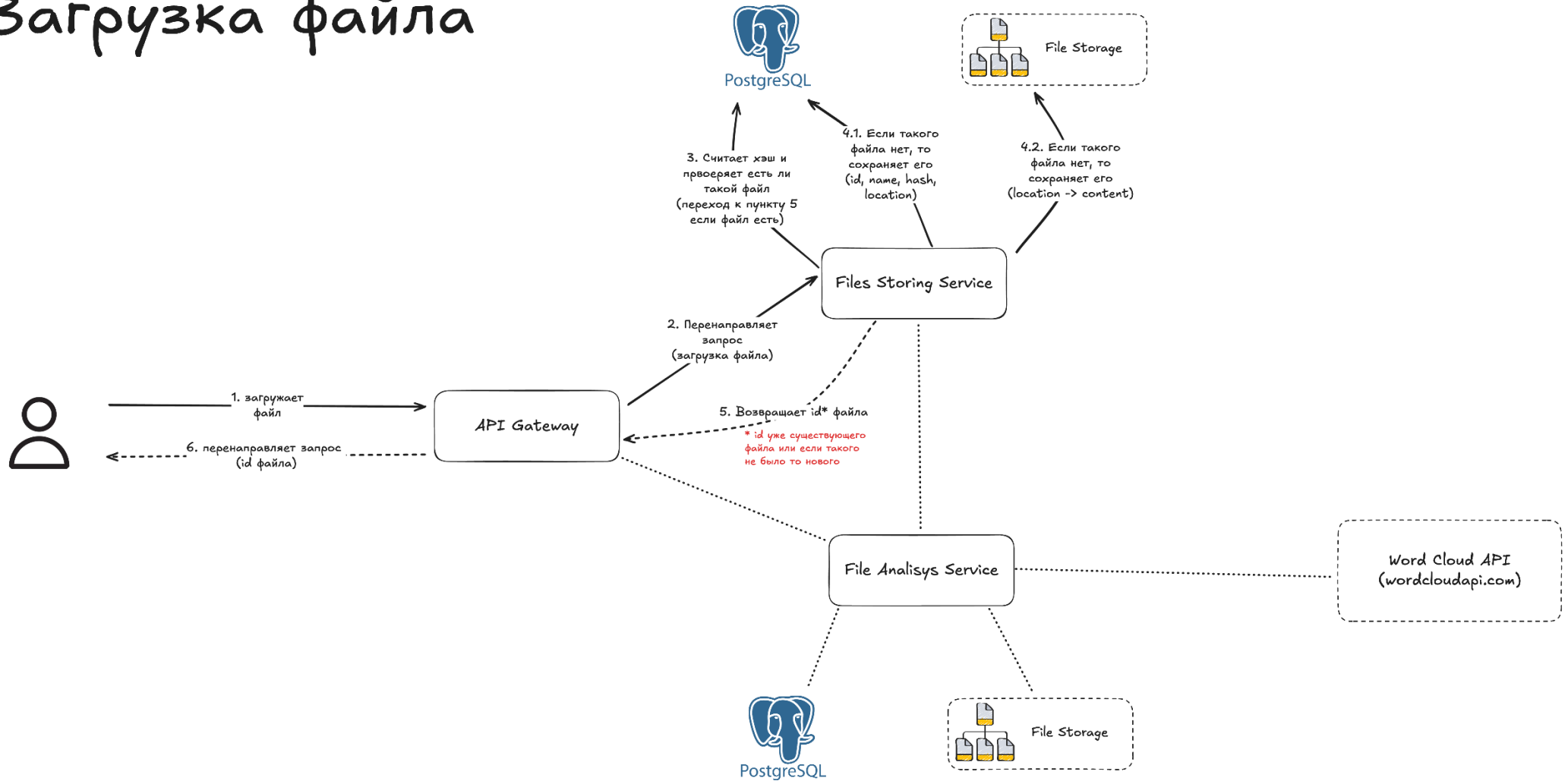
Штрафы

1. до – 2 баллов за наличие ошибки во время выполнения кода;
2. до – 5 баллов, если программа не собирается;
3. – 1 балл за каждый день просрочки дедлайна
4. – 1 балл за грубое игнорирование кодстайла.

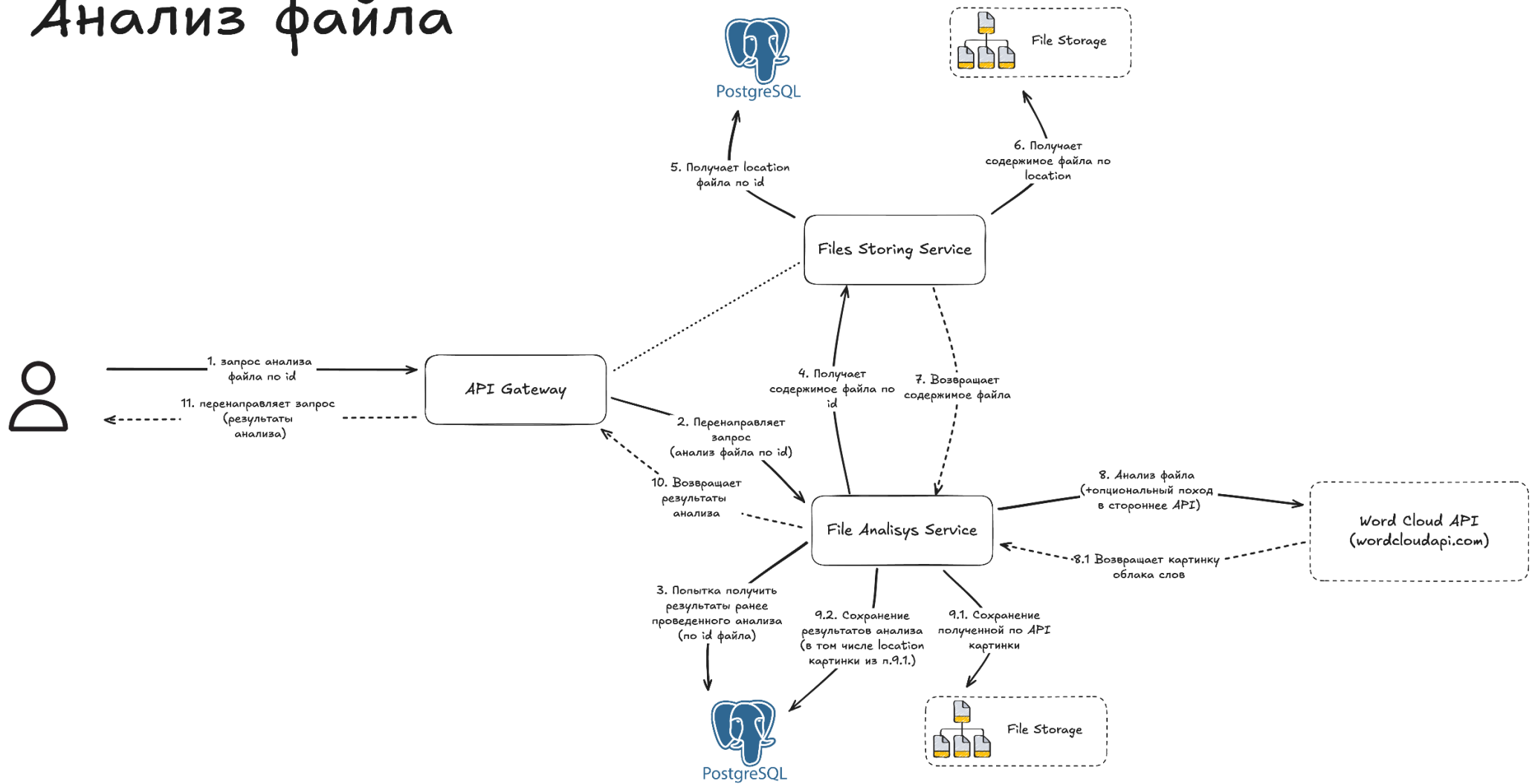
Документация к API построения облака слов: <https://quickchart.io/documentation/word-cloud-api/>

Пользовательские сценарии

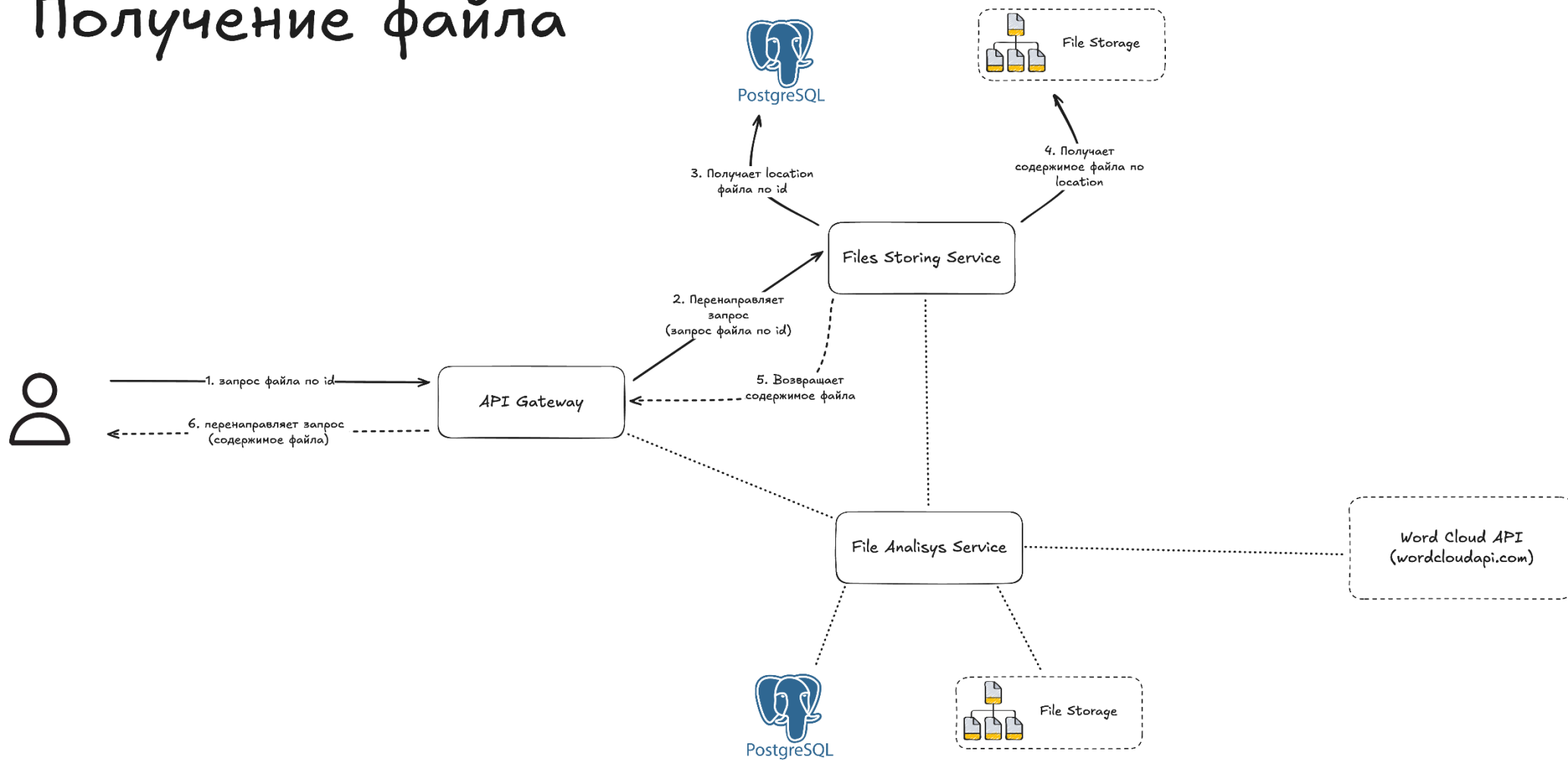
Загрузка файла



Анализ файла



Получение файла



Получение облака слов

