

|  |  |
| --- | --- |
| Задача  02 | **Сервис для прогнозирования и формирования закупок** |

Сопроводительная документация

Команда Khazir kasher:

* **Георгий Закала**
* Глеб Зотов
* Артём Маканов
* Андрей Путков
* Анна Денисова

Оглавление

[**Методы обработки данных** 3](#_Toc169441478)

[**Условия и ограничения внутри решения** 8](#_Toc169441479)

[**Инструкции по компиляции, сборке и установке** 9](#_Toc169441480)

[**Описание функциональной и компонентной архитектуры** 10](#_Toc169441481)

# 

# **Методы обработки данных**

Все решение задачи реализовано на сервере StatisticsServer.

Были созданы следующие элементы системы поддержки сервиса:

* реляционная база данных PostgreSQL 12 (далее - БД);
* веб-интерфейс в формате BI-системы Apache SuperSet (далее – SuperSet);
* веб-интерфейс в формате телеграм-бота с возможностью авторизации;
* возможность использования масштабированного сервиса OpenSearch, выполняющего функции полнотекстового поиска;
* единая точка регистрации, авторизации и аутентификации для использования обоих веб-интерфейсов на Keycloak.
* docker-контейнеры для организации на сервере BI-системы, телеграм-бота, сервисов для анализа данных и прогноза.

Для обеспечения системы обработки данных были созданы:

* скрипты загрузки и предобработки исходных таблиц в БД (SQL);
* скрипты предобработки и визуализации для телеграм-бота (Python – pandas, seaborn, prophet);
* скрипты для реализации прогноза потребности в закупках и анализ остатков, основанный на статистических методах (Python – pandas, prophet, statsmodels);
* надстройка интеллектуального поиска закупки по названию товара/работы/услуги с помощью системы полнотекстового поиска OpenSearch;
* комплекс дашбордов с визуальной аналитикой на SuperSet;
* система логирования;
* скрипт генерации и передачи json-файла с описанием закупки с использованием API.

Репозиторий с рабочим кодом решения находится по [ссылке](https://github.com/GZakala/lct_hack).

Рассмотрим последовательно решения, которые были применены для обеспечения связности данных, анализа, построения прогноза, генерации json-файла с описанием закупки.

1. Веб-интерфейс приложения сознательно был реализован в двух вариантах: телеграм-бот и SuperSet. Разработчики предусмотрели, что работа только через телеграм-бот может быть неудобной, особенно при необходимости анализа больших данных. По этой причине часть функций телеграм-бота была продублирована, а потом расширена за счет модулей глубокой аналитики в формате веб-сервиса на базе BI-системы SuperSet. Если пользователь хочет исследовать какой-то вопрос более подробно, возможности SuperSet подходят ему гораздо больше за счет своего функционала, средств визуализации, свободной масштабируемости и быстродействия. Вместе с тем, Keycloak обеспечивает единый вход пользователей в обе системы.
2. В БД загружены:
   * история закупок (контрактов) заказчика procurement\_contracts;
   * финансовые ведомости (бухгалтерские балансы по материальным активам) – собраны в одну таблицу и размечены по счету и отчетной дате financial\_data;
   * складские ведомости – собраны в одну таблицу и размечены по счету и отчетной дате balances\_account;
   * настройки для каждого товара по идентификатору СПГЗ, в последствии используемая для формирования json-файла с описанием закупки restrictions;
   * справочник спецификаций и характеристик товаров, работ и услуг kpgz\_spgz;
   * таблица с информацией о зарегистрированных пользователях users.
3. Для мэтчинга таблиц, в которых нет идентификаторов СПГЗ и КПГЗ был применен полнотекстовый поиск OpenSearch по наименованиям.
4. В телеграм-боте пользователь вводит название товара, по которому также с помощью OpenSearch ищутся наиболее релевантные варианты, далее из которых пользователь выбирает нужный.
5. Регулярные закупки размечены в БД по следующим правилам.
   * Для каждого уникального объекта считался набор значений – разницы во времени между соседними закупками.
   * Проверяется, можно ли считать полученный ряд значений для каждого уникального объекта однородным. С этой целью оценивается коэффициент вариации: традиционно считается, что если его значение меньше 33%, то ряд данных является однородным. Более продвинутые статистические методы здесь применять нецелесообразно, поскольку для абсолютного большинства объектов в выборке получится мало наблюдений.
   * На основе значения коэффициента вариации делается вывод: если ряд однороден, то закупка регулярна, если нет – то нерегулярна.
   * Далее оцениваем разницу между датой последней закупки соответствующего объекта и условной сегодняшней датой (исходя из исходных данных считаем, что это - 1 января 2023 года).
   * Если последняя разница превышает среднее значение больше, чем в два раза, то:

(а) в случае, если ранее закупка была идентифицирована как «регулярная», то сообщаем пользователю об этом, а также о том, когда эта регулярность в последний раз фиксировалась;

(б) в случае, если ранее закупка была идентифицирована как «нерегулярная», то пересчитываем коэффициент вариации разницы во времени между последней закупкой и условной сегодняшней датой. Делаем вывод по тому же правилу: если ряд значений можно считать однородным, то идентифицируем закупку как «регулярную», в противном случае – как «нерегулярную».

1. Прогнозирование потребности в закупке разделено на три части: (а) через телеграм-бот - прогноз потребности в товаре на складе; б) через телеграм-бот - прогноз потребности в работе/услуге.

Прогноз по варианту (а) строился по алгоритму.

* Для каждого уникального объекта, соответствующему варианту (а), оценивалась средняя интенсивность потребления за 2022 год – то есть сколько кварталов потребуется для потребления условной единицы каждого товара.
* С учетом складских остатков оценивается количество периодов, через которое товар на складе закончится. С учетом того, что от публикации закупки до заключения контракта проходит в среднем месяц, оценивается рекомендуемый срок проведения закупки.
* Считаем объем закупки по цене.
* По запросу через телеграм-бот возвращается прогноз по конкретному товару.

Прогноз по варианту (б) строился по алгоритму.

* Для каждого уникального объекта, соответствующему варианту (б), оценивалась средняя интенсивность потребления относительно НМЦ – то есть на сколько в среднем денежных средств расходуется в один период времени между соседними закупками этого объекта.
* Исходя из того, на какую сумму была закупка соответствующего объекта в последний раз (с поправкой на средний уровень инфоляции) можно понять, через сколько времени может понадобиться следующая закупка.
* Считаем объем закупки по цене.
* По запросу через телеграм-бот возвращается прогноз по конкретному товару.

По каждому из вариантов пользователь может выбрать период прогнозирования: на месяц, квартал, или год. Например, при выборе периода «месяц» будет выведена информация по всем товарам и прогноз их расхода на заданный месяц.

1. Модуль визуальной аналитики доступен через телеграм-бот и реализован средствами Python. По запросу он предоставляет графики складских остатков и прогноза потребности.
2. Система дашбордов на SuperSet повторяет и расширяет аналитические возможности телеграм-бота, а также закрывает многие бизнес-потребности. Пользователь может ознакомиться с дашбордами по темам:
   * общая статистика по контрактам, где представлена информация о динамике закупок и частоте заключения контрактов, способах выбора поставщика, стоимости контрактов и их исполнении, категориях закупаемых товаров, работ, услуг;
   * финансовый отчет, где представлена информация о сальдо, разнообразии товарных позиций и о том, какое количество позиций остается на балансе в начале и конце 2022 года, динамике поступления и потребления товаров;
   * складские остатки, где представлена информация об актуальных остатках, состоянии амортизации, введении товаров в эксплуатацию.

В дашбордах можно настроить фильтры по заказчику, дате, категории товаров и др. показателям. Кроме того, туда интегрирован модуль прогнозирования потребности в закупках и остатков товаров на складе.

1. Генерация json-файла с описанием закупки осуществляется только через телеграм-бот, в соответствии с техническим заданием. Пользователю предлагается ввести название товара/работы/услуги для оформления закупки. По названию с помощью OpenSearch определяется релевантный идентификатор СПГЗ. На основе средней цены за единицу товара, которая содержится в финансовых ведомостях, оценивается начальная (максимальная) цена контракта (НМЦ). Для услуг считается средняя НМЦ по истории закупок.

# **Условия и ограничения внутри решения**

1. Все расчеты представлены для единственного заказчика «Заказчик 1», данные о деятельности которого есть в исходных файлах.
2. Соответственно, все товары, работы и услуги, которые пользователь может искать, это только те товары и услуги, которые либо хотя бы раз закупались заказчиком, либо стояли у него на балансе в 2022 году (согласно исходным данным). Формально в БД решения добавлена таблица с общим справочником, однако в функционале она не использовалась, так как не для всех товаров, работ и услуг будет возможность реализовать поставленные задачи.
3. Основной идентификатор товара/работы/услуги – идентификатор СПГЗ или сочетание КПГЗ-СПГЗ. Для обеспечения связности складских и балансовых ведомостей с остальными таблицами, прежде всего со статистикой закупок, использовался OpenSearch для определения схожести названий товаров.
4. Системы визуализации статистики в телеграм-боте и в SuperSet технически между собой не связаны, однако данные для их работы берутся из одного источника и проходят те же шаги предобработки с помощью тех же модулей. Ключевая разница заключается в том, что SuperSet предоставляет пользователю расширенный доступ к данным и аналитике по ним.

# **Инструкции по компиляции, сборке и установке**

Телеграм-бот доступен с любого устройства по адресу: <https://t.me/KhazirKasherBot>.

Там реализована система авторизации и аутентификации c двумя ролями: администратор и обычный пользователь. Новых пользователей регистрирует администратор. Он также управляет разрешениями на получение прогноза, генерацию json-файла о закупке, реализует создание и удалении новых учетных записей пользователей. В свою очередь, обычный пользователь при наличии соответствующих разрешений может осуществлять поиск товара, просмотр аналитики по остаткам и прогноз потребности, генерацию json-файла с описанием закупки. Пользователю в чате с ботом предлагается перейти в другой веб-интерфейс – SuperSet, для более подробного анализа. Повторно авторизовываться не надо – пользователь будет переведен туда под своей учетной записью (реализовано на Keycloak).

Доступ вSuperSet также возможен с любого устройства по [ссылке](https://superset.cosahack.ru/).

Для тестирования предлагается следующая учетная запись:

**логин** admin,

**пароль** admin.

Для запуска системы на любом компьютере подготовлены docker-файлы, создан специальный [makefile](https://github.com/GZakala/lct_hack/blob/master/Makefile) со скриптами загрузки. Для установки необходимо скачать рабочий репозиторий и далее просто запустить makefile.

# **Описание функциональной и компонентной архитектуры**

Перечень всех использованных библиотек и компонентов:

Docker, PostgreSQL, Apache SuperSet, Keycloak, Python (sys, pathlib, telebot, json, requests, io, flask, keycloak, functools, typing, opensearchpy, psycopg2, utils, jinja2, pandas, numpy, statsmodels, prophet, seaborn).

Файл спецификации для StatisticsServer API описывает два метода:

* PUT /user: для добавления нового пользователя с параметрами username, md5(parol), department, permission\_admin, permission\_forecast, и permission\_json. Возвращает true в случае успешного добавления пользователя.
* GET /user: для получения md5 пароля пользователя и его сравнения с переданным паролем. Возвращает bool, указывающий, авторизован ли пользователь, и если да, то возвращает словарь с правами доступа и отделом.

Если пользователь хочет выйти из своей учетной записи, он должен ввести команду /start.

Файл доступен в общем репозитории.

