

ТЕХНИЧЕСКОЕ ЗАДАНИЕ НА РАЗРАБОТКУ СБОРЩИКА ТЕНДЕРОВ С ПЛОЩАДКИ «ПАО Полюс»

1. Введение

Данное техническое задание разработано для сборщика с площадки «ПАО Полюс».

Программа предназначена для сбора тендеров с сайта/платформы «ПАО Полюс». Ниже перечислены следующие функции программы:

- 1) Отправка и обработка HTTP – запроса к серверу площадки с заданными входными параметрами;
- 2) Обработка ответа от сервера площадки;
- 3) Парсинг в модель;
- 4) Проверка на существующий тендер с таким же номером закупки/извещения и изменения внутри него;
- 5) Добавление минимизированной модели тендера в базу данных или обновление уже существующей в базе;
- 6) Отправка модели в очередь.

1.1. Функциональные и технические требования

Язык разработки: Python 3.x

База данных: MongoDB (библиотека pymongo)

Очередь для передачи конечных данных: RabbitMq (библиотека pika)

Требования к потреблению оперативной памяти: не более 100 Мб

Формат даты: UTC Unix timestamp epoch (мс)

Ссылка на сайт: <http://tenders.polyusgold.com/purchases/>

2. Описание площадки

На площадке «ПАО Полюс» размещаются коммерческие тендеры.

Примечание:

На площадке размещаются коммерческие тендеры, собирать нужно все.

Поля для сбора

- 1) Наименование тендера (Наименование Лота, если их несколько):

Текущие закупки

[RM-2018-0128] Тендер на поставку компьютерного оборудования и комплектующих

Дата публикации: 13.04.2018 | Прием заявок до: 23.04.2018 | Статус закупки: Рассмотрение заявок
АО "РиМ" | Оборудование ИТ и ПО

Предмет закупок

Контактное лицо: Казакова, Надежда
8 (4132) 691-100, доб. 7-45-07

Ссылка для загрузки ТКП: <https://supplier.polyusgold.com/Upload/?campaignId=1c65e4a4-d83e-e811-a1c3-00155d0141b2>

2) Статус закупки

Текущие закупки

[RM-2018-0128] Тендер на поставку компьютерного оборудования и комплектующих

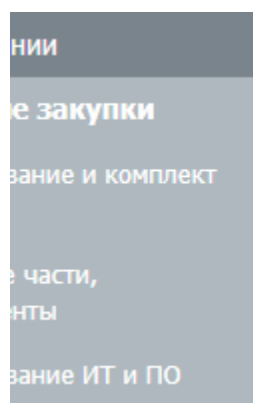
Дата публикации: 13.04.2018 | Прием заявок до: 23.04.2018 | Статус закупки: Рассмотрение заявок
АО "РиМ" | Оборудование ИТ и ПО

Предмет закупок

3) Регион

4) Площадка: <http://tenders.polyusgold.com/>

5) Заказчик



Текущие закупки

[RM-2018-0128] Тендер на поставку компьютерного оборудования и комплектующих

Дата публикации: 13.04.2018 | Прием заявок до: 23.04.2018 | Статус закупки: Рассмотрение заявок
АО "РиМ" | Оборудование ИТ и ПО

Предмет закупок

6) Информация об объекте закупки:

6.1) Наименование



Текущие закупки

[RM-2018-0128] Тендер на поставку компьютерного оборудования и комплектующих

Дата публикации: 13.04.2018 | Прием заявок до: 23.04.2018 | Статус закупки: Рассмотрение заявок
АО "РиМ" | Оборудование ИТ и ПО

Предмет закупок

7) Порядок размещения заказа

7.1) Завершение подачи заявок

Текущие закупки

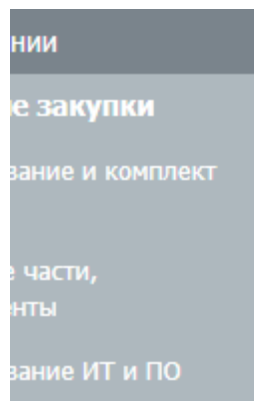
[RM-2018-0128] Тендер на поставку компьютерного оборудования и комплектующих

Дата публикации: 13.04.2018 | Прием заявок до: 23.04.2018 | Статус закупки: Рассмотрение заявок
АО "РиМ" | Оборудование ИТ и ПО

Предмет закупок

8) Контактная информация

8.1) Организация



Текущие закупки

[RM-2018-0128] Тендер на поставку компьютерного оборудования и комплектующих

Дата публикации: 13.04.2018 | Прием заявок до: 23.04.2018 | Статус закупки: Рассмотрение заявок
АО "РиМ" | Оборудование ИТ и ПО

Предмет закупок

8.2) Контактное лицо

АО "РиМ" | Оборудование ИТ и ПО

Предмет закупок

Контактное лицо: Казакова, Надежда

8 (4132) 691-100, доб. 7-45-07

Ссылка для загрузки ТКП: <https://supplier.polyusgold.com/Upload/?campaignId=1c65e4a4-d83e-e811-a1c3-00155d0141b2>

Документы по текущей закупке

8.3) Телефон:

АО "РиМ" | Оборудование ИТ и ПО

Предмет закупок

Контактное лицо: Казакова, Надежда

8 (4132) 691-100, доб. 7-45-07

Ссылка для загрузки ТКП: <https://supplier.polyusgold.com/Upload/?campaignId=1c65e4a4-d83e-e811-a1c3-00155d0141b2>

Документы по текущей закупке

9) Дата публикации

Текущие закупки

[RM-2018-0128] Тендер на поставку компьютерного оборудования и комплектующих

Дата публикации: 13.04.2018 | Прием заявок до: 23.04.2018 | Статус закупки: Рассмотрение заявок
АО "РиМ" | Оборудование ИТ и ПО

Предмет закупок

10) Ссылка на площадку: <http://tenders.polyusgold.com/purchases/>

11) Номер закупки / Ссылка на закупку

Текущие закупки

[RM-2018-0128] Тендер на поставку компьютерного оборудования и комплектующих

Дата публикации: 13.04.2018 | Прием заявок до: 23.04.2018 | Статус закупки: Рассмотрение заявок
АО "РиМ" | Оборудование ИТ и ПО

Предмет закупок

12) Документы закупки

Ссылка для загрузки ТКП: <https://supplier.polyusgold.com/Upload/?campaignId=1c65e4a4-d83e-e811-a1c3-00155d0141b2>

Документы по текущей закупке

Документация.zip

Приглашение на участие в тендере.pdf

3. Описание программного продукта

3.1. Структура сборщика.

Сборщик тендеров с площадки должен иметь следующую нижеприведенную структуру:

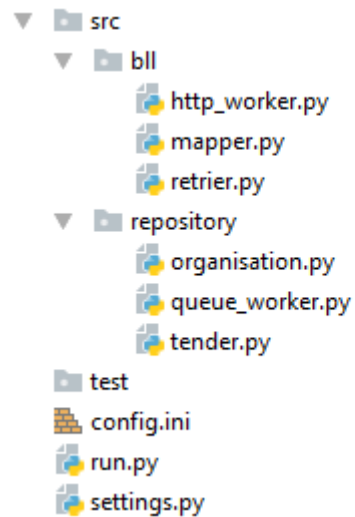


Рис.1. Структура проекта

В папке src должны располагаться директории bll и repository. В repository содержатся модули по работе с базой данных, очередью и api организаций, в bll — .py – модули по отправке и обработке HTTP – запроса к площадке, парсинг данных по тендеру в модель. Модель, необходимая нашей системе, прилагается к техническому заданию в архиве вместе с JSON – схемой, содержащейся в модели.

Модуль run.py в проекте сборщика является основным: именно в нем должна быть реализация логики программного продукта по сбору тендеров с площадки. Конфигурации подключения к очереди, базе данных и api организаций должны храниться в config.json, чтение и парсинг последующего файла конфигурации происходит в settings.py.

В директории test должны присутствовать модули с тестами данного сборщика. Рекомендуемая структура модулей находится в приложении к техническому заданию.

3.2. Описание модели.

В процессе работы сборщика, данные, пришедшие в качестве ответа с сервера площадки, преобразуются в модель, пример которой лежит в архиве в файле CommonModel.json. Саму структуру модели можно посмотреть на схеме 1 в приложении.

Как видно по схеме, класс Attachment, содержащий в себе информацию о файле, Customers с информацией о заказчике, и Platform, где указывается название площадки и ссылка на ее основную страницу, являются частью CommonModel.

Для получения данных по организации, в модели Customers, следует создать заглушку следующего формата запроса:

host/organization?inn=инн&kpp=кпп, где в качестве query параметра используются инн и кпп организации.

Более подробную информацию о моделях и их полях смотреть в табл.1 — 6 соответственно.

Таблица 1. Данные модели CommonModel

Название	Описание
id	Номер тендера+ ”_” + номер лота
customers	Создать заглушку - на получение данных из стороннего сервиса, формат запроса: host/organization?inn=инн&kpp=кпп Вставлять в модель название и регион (см. Табл. 4)
globalSearch	Соединённые через пробел: okdp, okpd, okpd2, number,orderName, customers.name
guaranteeApp	Обеспечение заявки
status	см. Табл. 4
href	Ссылка на тендер
json	см. п. 3.3
maxPrice	Максимальная цена
multilot	Многолотовый тендер или нет
number	Номер тендера
okdp, okpd, okpd2	Классификаторы
orderName	Название тендера
organisationsSearch	Соединённые через пробел: inn + kpp + customers.name + customers.region
platform	см. Табл. 3
publicationDateTime	Дата публикации тендера на площадке
region	Место проведения
submissionCloseDateTime	Дата окончания приёма заявок
submissionStartDateTime	Дата начала подачи заявок
tenderSearch	Соединённые через пробел: number,orderName, customers.name
timestamp	Дата прихода тендера на сборщик
version	Версия тендера (если есть изменения)

Таблица 2. Данные модели Platform

Название	Описание
href	Ссылка на страницу площадки
name	Название площадки

Таблица 3. Данные модели Customer

Название	Описание
guid	GUID организации (берется значение из JsonModel)
name	Наименование организации
region	Код региона (берется из ИНН)

Таблица 4. Значения полей status

Название	Описание
0	None
1	Active
2	Commission
3	Closed
4	Cancel
5	Abandoned

Таблица 5. Значение полей регионов

```
{
  "москва": 77,
  "московская область": 50,
  "ярославская область": 76,
  "ивановская область": 37,
  "костромская область": 44,
  "вологодская область": 35,
  "архангельская область": 29,
  "ненецкий автономный округ": 89,
  "коми республика": 11,
  "тверская область": 69,
  "новгородская область": 53,
  "псковская область": 60,
  "мурманская область": 51,
  "карелия республика": 10,
  "ленинградская область": 47,
  "санкт-петербург": 78,
  "смоленская область": 67,
  "калининградская область": 39,
  "брянская область": 32,
  "калужская область": 40,
  "крым республика": 91,
  "севастополь": 92,
  "тульская область": 71,
  "орловская область": 57,
```


"курская область": 46,
"белгородская область": 31,
"ростовская область": 61,
"краснодарский край": 23,
"ставропольский край": 26,
"калмыкия республика": 8,
"кабардино-балкарская республика": 7,
"северная осетия-алания республика": 15,
"чеченская республика": 95,
"дагестан республика": 5,
"карачаево-черкесская республика": 9,
"адыгея республика": 1,
"ингушетия республика": 6,
"рязанская область": 62,
"тамбовская область": 68,
"воронежская область": 36,
"липецкая область": 48,
"волгоградская область": 34,
"саратовская область": 64,
"астраханская область": 30,
"татарстан республика": 16,
"марий эл республика": 12,
"удмуртская республика": 18,
"чувашская республика": 21,
"мордовия республика": 13,
"ульяновская область": 73,
"пензенская область": 58,
"самарская область": 63,
"башкортостан республика": 2,
"челябинская область": 74,
"оренбургская область": 56,
"казахстан": 99,
"германия": 0,
"южная осетия республика": 0,
"владимирская область": 33,
"нижегородская область": 52,
"кировская область": 43,
"пермский край": 59,
"свердловская область": 66,
"тюменская область": 72,
"ханты-мансийский-югра автономный округ": 86,
"ямало-ненецкий автономный округ": 89,
"новосибирская область": 54,
"томская область": 70,
"курганская область": 45,
"омская область": 55,
"красноярский край": 24,
"алтай республика": 4,
"кемеровская область": 42,
"хакасия республика": 19,
"алтайский край": 22,
"иркутская область": 38,
"тыва республика": 17,

```
"бурятия республика": 3,  
"забайкальский край": 75,  
"амурская область": 28,  
"саха (якутия) республика": 14,  
"еврейская автономная область": 79,  
"хабаровский край": 27,  
"камчатский край": 41,  
"магаданская область": 49,  
"чукотский автономный округ": 87,  
"приморский край": 25,  
"сахалинская область": 65  
}
```

3.3. Описание json, хранящегося в качестве строки в CommonModel.

Данные, интерпретируемые в модель CommonModel, также входят в json – схему. Ее пример, равно как и поля, которые должны присутствовать в ней, можно посмотреть в файле JsonModel.json находящийся в архиве с техническим заданием. Подобная json – схема хранится в модели, указанной выше, в качестве строки в поле *json*.

JSON – схема включает следующие разделы:

- информация об объекте закупки — наименование тендера, его тип (запрос предложений/аукцион/проч.), начальная цена и информация по торговой единице тендера
- контактная информация — информация об организации, выставившей тендер;
- контакты — контактное (ответственное) лицо;
- порядок размещения заказа — раздел, в котором указываются основные даты этапов по тендеру;
- general — остальная не вошедшая информация, такая как: ставка НДС, альтернативные предложения и проч.;
- customers — включает в себя информацию по ценовому обеспечению контракта тендера и поле заказчика.

Каждое поле раздела состоит из следующих наименований атрибутов и их свойств, указанных на рис.2.

```
"ch": {},  
"fdn": "Альтернативные предложения",  
"fn": "",  
"ft": "String",  
"fv": "Принимается не более 1 альтернативных предложений от одного участника",  
"md": []
```

Рис.2. Пример построения

Атрибуты:

- fdn — field display name — здесь указывается название поля для json модели;
- fn — field name — наименование поля (на англ.);
- ft — field type — Тип поля. Может быть следующим: None, Integer, String, DateTime, Date, Price, Array, Bool, Object;

- fv — field value — значение, берётся с данных по тендеру;
- md — modification — следующие типы: Total, Calendar, Help, CurUSD, CurRUB, CurEUR, HiddenLabel, ProductSearch, Email.

3.4. Функциональная диаграмма сборщика

Как примерно должен работать сбор тендеров с площадки указано в приложении на схеме 2.

Входные параметры, передаваемые сборщику, определяются для формирования последующего HTTP – запроса, и зависят от критерий поиска/фильтраций тендеров на площадке. *Например:* площадке для сбора тендеров требуется задавать query – строку с количеством тендеров и датой публикации. В качестве входных параметров для сбора будет указываться количество тендеров, дата публикации, т.е. в `def collect(quantity: int, publish_date: datetime)`, где `quantity` — это количество тендеров, `publish_date` — дата публикации.

Для обращения к серверу сборщика следует прописать адреса прокси.

При успешном выполнении запроса в качестве ответа приходят данные.

Проходимся по каждому тендеру и формируем данные по нему в следующие модели:

- модель (см. п. 3.2) для дальнейшей отправки в очередь;
- минимизированная модель для хранения в базе данных. Например, можно хранить в базе номер тендера, дату его публикации на сайте сборщика, и текущий статус (опубликован/принятие заявок/рассмотрение комиссии и т.п.).

Есть случаи, когда по тендеру была обновлена информация. В таком случае, если он уже хранится в базе, то проверяем на изменения и обновляем его, как и в базе, так и заново приводим к модели (см. п. 3.2) для отправки ее в очередь.

Так цикл повторяется для каждого тендера в списке (HTTP – запрос для получения данных по тендеру —> обработка в модель для очереди и отправка в очередь —> обработка в модель для хранения в бд), и в случае обработки последнего из него, снова формируется запрос для получения очередных тендеров и так далее, покуда сборщик не достигнет последнего тендера, опубликованного на площадке по заданным параметрам.

После сбора тендеров программа должна уходить в спящий режим на промежуток времени, заданный в конфигурации. По окончании режима сборщик повторяет весь цикл, описанный выше.

3.4. Диаграмма активности

На диаграмме активности (приложение, схема 3) можно видеть, что в сборщике задействованы три внешних источника:

- площадка — первый источник, задействован для отправки запроса и получения ответа в форме данных;
- хранилище данных — база данных, где будут храниться тендеры. В случае обновления или вставки информации по нему, база принимает эти данные;
- очередь — туда отправляются данные по тендеру, сформированные в модель CommonModel (см. п. 3.2.)

С площадкой в сборщике связаны две активности: формирование данных для запроса, где используются входные параметры, а также определяется использование прокси, и обработка ответа с сервера площадки.

Для работы со вторым внешним источником — базой данных — данные по тендеру преобразуются в минимизированную модель, используемую для проверки на наличие или изменения, и далее вставки или обновления.

Отправка в очередь требует использование последнего источника — очереди — куда сборщик отсылает информацию по тендеру, приведенную к модели CommonModel (см. п.3.2.).

Весь цикл сборщика оканчивается в случае прохождения по последнему тендеру, и уже затем уходит в спящий режим на заданный промежуток времени.

3.5. Диаграмма классов

Приведенная в приложении на схеме 4 диаграмма дана в пример.

В результате построения функциональной диаграммы программа по сбору тендеров с площадки выводится приблизительная диаграмма классов, в которых осуществляется следующее:

- Repository — работа с базой данных. В ней реализуется подключение к хранилищу, вставка тендера в базу и проверка на уже существующий;
- HttpWorker и Retrier — классы, отвечающие за выполнение HTTP – запроса и повторного его выполнения в случае ошибки выполнения запроса сервером площадки;
- Mapper — отвечает за парсинг данных по тендеру/извещению в модель CommonModel, отправляемая в очередь;
- QueueWorker — класс отправки тендера, преобразованного в CommonModel, в очередь. Прилагается к техническому заданию в архиве;
- Organization — предназначен для работы со сторонним сервисом, к которому отсылается запрос следующего типа: `host/organization?inn=инн&kpp=кпп`;
- Settings — парсинг файла конфигурации;
- Collector — основной класс сборщика.

Приложение

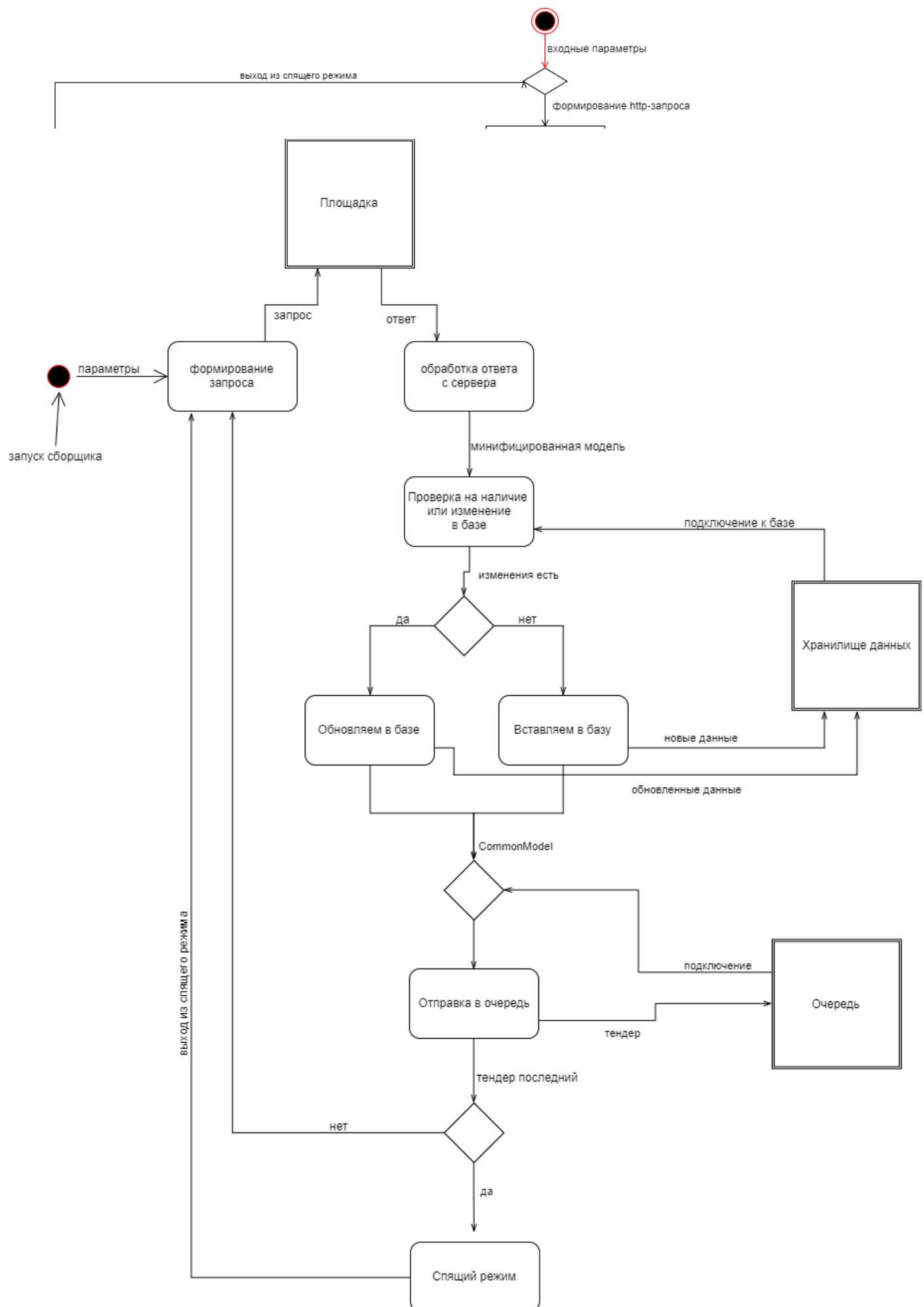


Схема 2. Диаграмма активности

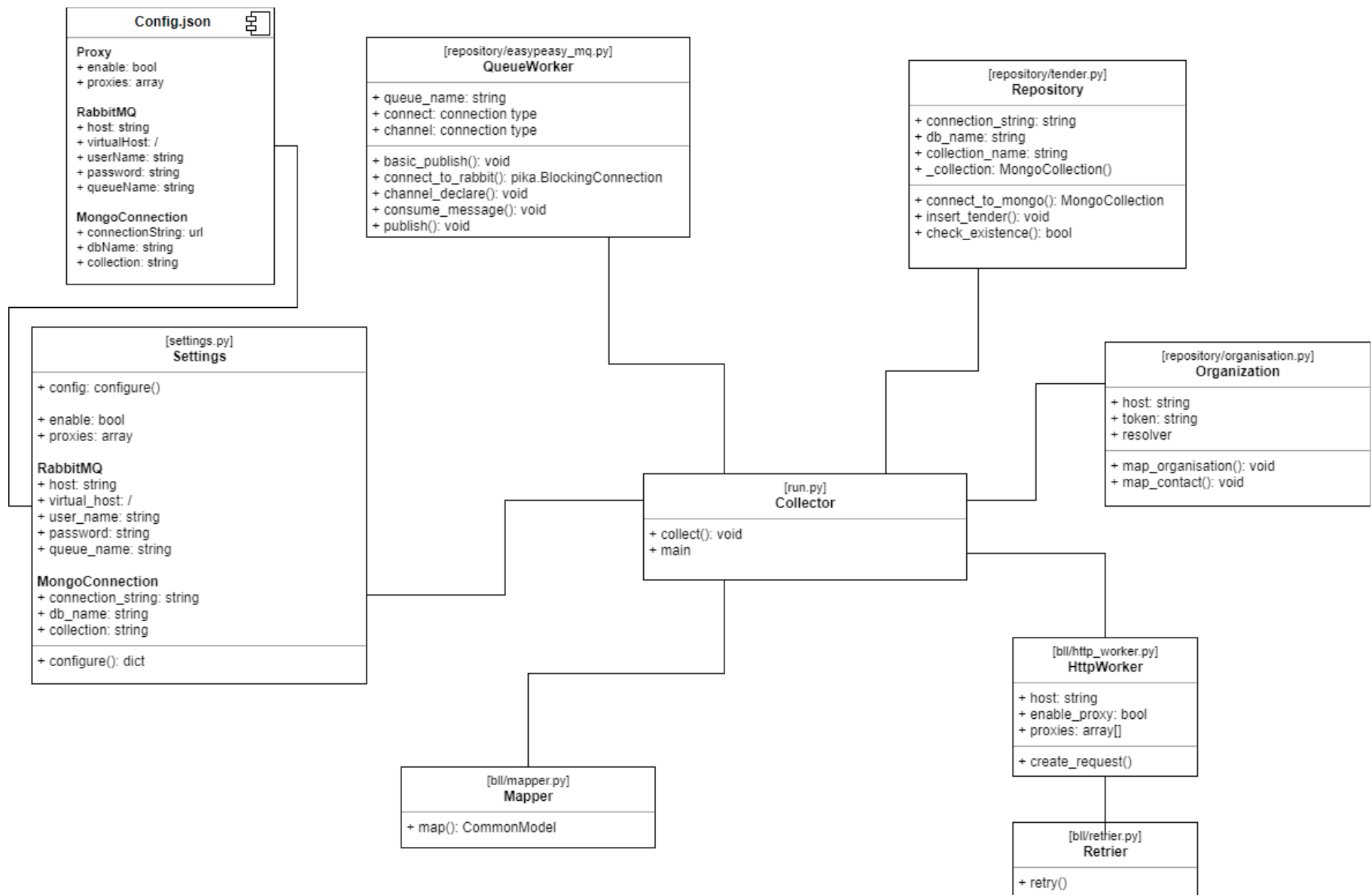


Схема 3. Диаграмма классов