**Описание архитектуры сервиса**

Сервис позволяет на основе заданных параметров сгенерировать тайл на основе тайлов более высокого или низкого уровня.

**Стек технологий:**

* Python 3.10
* Mercantile
* Pillow
* Flask
* Flask-RESTX
* Docker
* Docker Swarm

Файл сервиса tile\_provider.py отвечает за генерацию шаблона ссылки для обращения к серверам провайдеров.

Файл retiler.py содержит основную логику генерации тайлов.

Файл api.py обеспечивает для сервиса API формата:

GET /retile/{provider}/{level}/{resolution}/{z}/{x}/{y}.png

**Основные классы:**

* TileProvider
* Retiler

**Классы исключений:**

* InvalidInputParamException

**Классы маршрутизации:**

* GetRetile

На рисунке 1 показана диаграмма классов.

Класс TileProvider инициализируется параметрами из файла конфигурации (url\_template, tile\_size, max\_zoom, token). Метод download\_tile с помощью метода create\_url формирует ссылку для запроса исходного тайла и возвращает его в виде изображения.

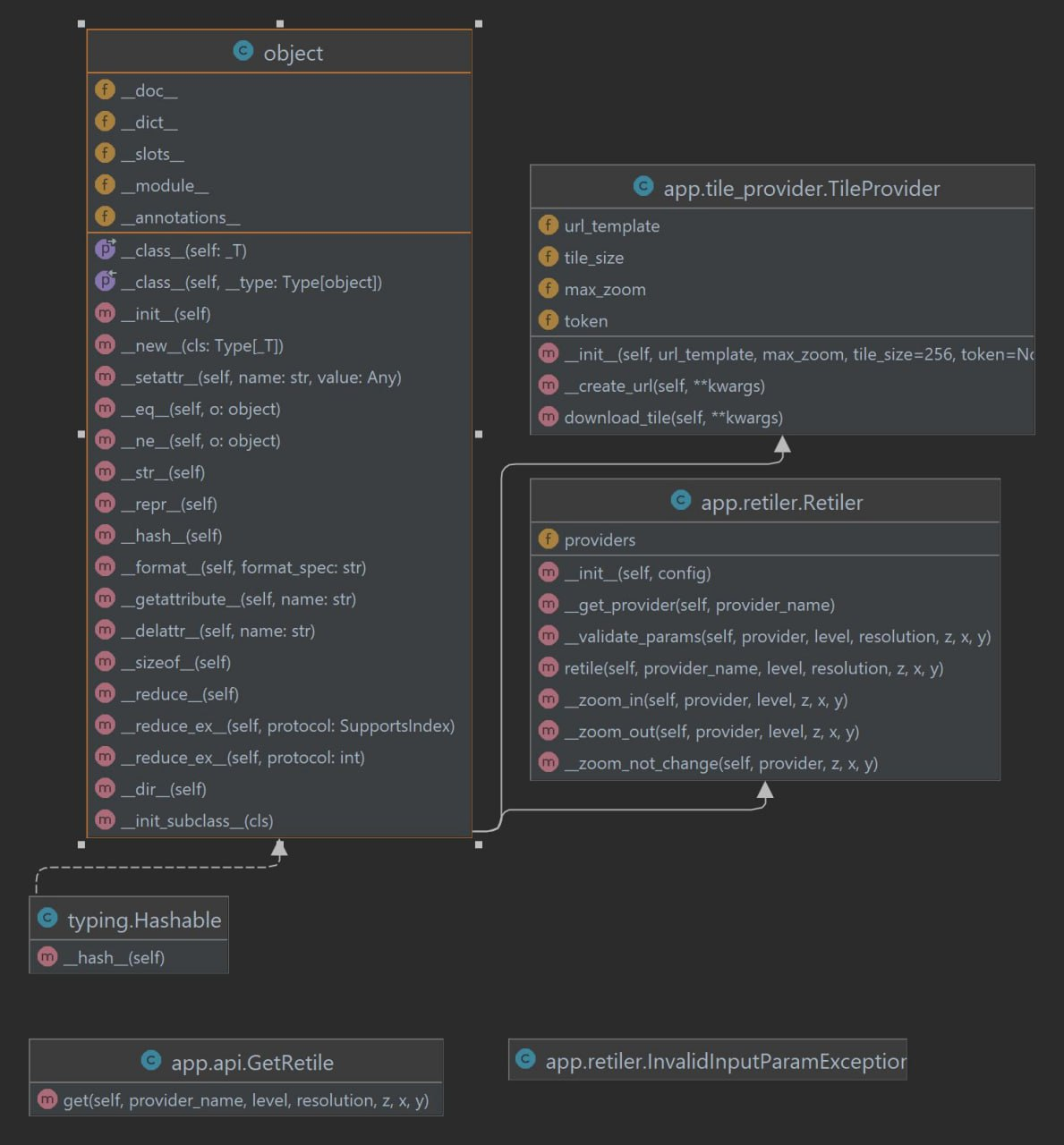


Рисунок 1 – Диаграмма классов

Класс Retiler инициализируется файлом конфигурации и на основе класса TileProvider формирует и заносит в поле providers список доступных провайдеров в виде объектов класса TileProvider.

Данный класс содержит метод \_\_get\_provider, который принимает название провайдера и проверяет его наличие в списке providers. В случае успеха возвращаемым значением будет соответсвующий элемент списка providers, в противном случае будет вызвано исключение типа InvalidInputParamException.

Метод validate\_params класса Retiler выполняет проверку входных параметров (provider, resolution, z, x, y), в случае некорректных входных данных будет вызвано исключение типа InvalidInputParamException.

Метод retile в зависимости от входного параметра level вызывает соответсвующий метод \_\_zoom\_in, \_\_zoom\_out или \_\_zoom\_not\_change.

\_\_zoom\_in отвечает за генерацию нового тайла на основе тайлов более высокого уровня при положительном смещении зума (координата z). В первую очередь на основе входных параметров создается исходный тайл. Затем формируется список дочерних тайлов уровня z + level, после чего они загружаются с сервера и объединяются в результирующий тайл. Полученый тайл в виде изображения – возвращаемое значение метода.

\_\_zoom\_out отвечает за генерацию нового тайла на основе тайлов более низкого уровня при отрицательном смещении зума. Аналогично методу \_\_zoom\_in сперва создается исходный тайл. После чего формируется его родительский тайл уровня z + level (level < 0), который, в свою очередь, подгружается с сервера и необходим для выделения из него результирующего тайла. Список дочерних тайлов уровня z родительского тайла используются для определения вырезаемой области.

В случае нулевого смещения зума метод \_\_zoom\_not\_change вернет исходный тайл.

Пользовательский класс исключения InvalidInputParamException наследуется от базового класса Exception и используется для обработки ошибок, связанных с некорректными входными параметрами.

Класс маршрутизации GetRetile содержит метод get соответсвующий HTTP-методу GET. Внутри него вызывается метод retile класса Retiler, котрый возвращает сгенерированный тайл. Возвращаемым значением метода get является файл с расширением png. Также сервис возвращает следующие коды ответа:

* 200 – когда тайл успешно сгенерирован;
* 422 – переданы неверные параметры запрос;
* 500 – произошла внутренняя ошибка сервиса.