Комитет по образованию Правительства Санкт-Петербурга

**САНКТ-ПЕТЕРБРУГСКИЙ КОЛЛЕДЖ ИНФОРМАЦИОННЫХ   
ТЕХНОЛОГИЙ**

**Отчет по практической работе  
МДК 01.02 «Разработка мобильных приложений»  
Разработка интерактивного приложения карт  
c использованием REST API**

Выполнил

студент группы 493:

Лукьянов И. А.

Преподаватель: Фомин А.В.

Санкт-Петербург 2022

Структура базы данных

База данных состоит из 5 таблиц:

1. Tile – хранит загруженные тайлы.
2. Layer – хранит настройки дополнительных слоев.
3. CacheTime – хранит время кэширования.
4. LastPosition – хранит последнее место на карте.
5. APIEndPoint – хранит конечную точку API.

ER диаграмма представлена на рисунке 1.

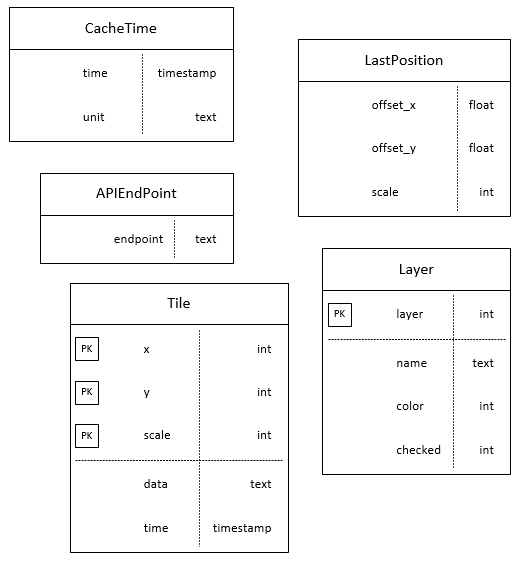


Рисунок 1 – ER диаграмма базы данных

**Таблица Tile**

Содержит сведения о загруженных тайлах. Таблица состоит из пяти столбцов:

1. x – координата тайла по горизонтали.
2. y – координата тайла по вертикали.
3. scale – уровень масштабирования.
4. data – изображение в формате строки.
5. time – время создания тайла.

Подробное описание столбцов представлено на рисунке 2.

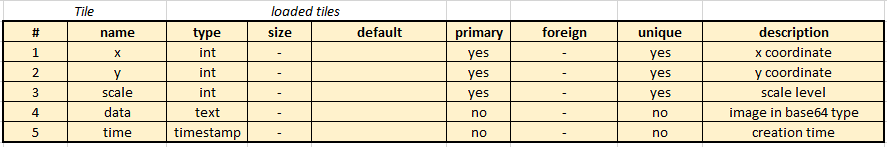


Рисунок 2 – Описание столбцов таблицы Tile

**Таблица Layer**

Содержит сведения о настройках дополнительных слоев. Таблица состоит из четырех столбцов:

1. layer – уникальный идентификатор слоя.
2. name – название слоя.
3. color – цвет слоя в численном выражении.
4. checked – статус слоя: выбран или нет.

Подробное описание столбцов представлено на рисунке 3.

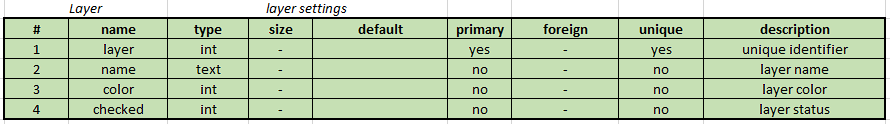


Рисунок 3 – Описание столбцов таблицы Layer

**Таблица LastPosition**

Содержит сведения о последнем месте на карте. Таблица состоит из трёх столбцов:

1. offset\_x – координата по горизонтали.
2. offset\_y – координата по вертикали.
3. scale – уровень масштабирования.

Подробное описание столбцов представлено на рисунке 4.

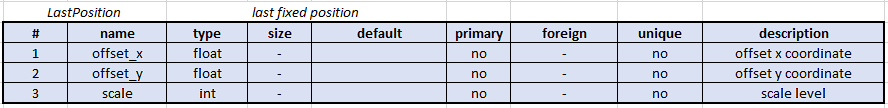


Рисунок 4 – Описание столбцов таблицы LastPosition

**Таблица CacheTime**

Содержит сведения о времени кэширования тайлов. Таблица состоит из двух столбцов:

1. time – время кэширования.
2. unit – единица измерения: секунды/минуты/часы/дни.

Подробное описание столбцов представлено на рисунке 5.

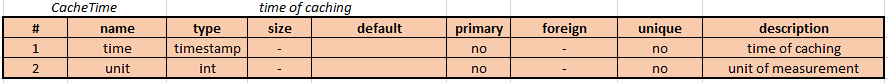


Рисунок 5 – Описание столбцов таблицы CacheTime

**Таблица APIEndPoint**

Содержит сведения о конечной точке API. Таблица состоит из одного столбца:

1. endpoint – конечная точка API.

Подробное описание столбцов представлено на рисунке 6.



Рисунок 6 – Описание столбцов таблицы APIEndPoint

Интерфейс приложения

Приложение состоит из 2 форм:

1. Main Form: служит для отображения карты и изменения масштаба, а также перехода к форме настроек.
2. Settings Form: форма настроек, для задания конечно точки API, времени кэширования, а также выбора активных дополнительных слоев и их цветов.

**Форма Main Form**

На рисунке 7 показан макет внешнего вида основной формы.

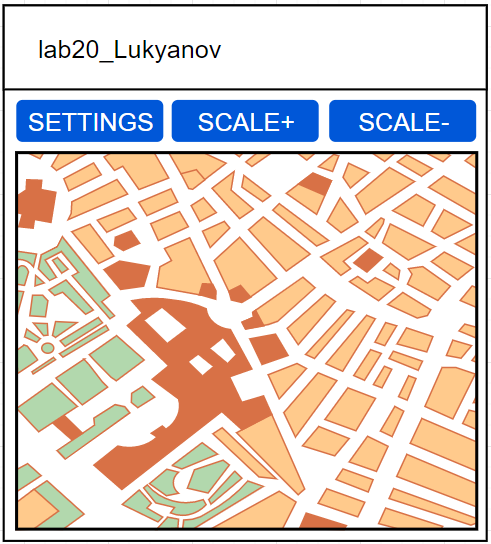


Рисунок 7 – Макет формы Main Form

На рисунке 8 показан внешний вид основной формы в приложении.

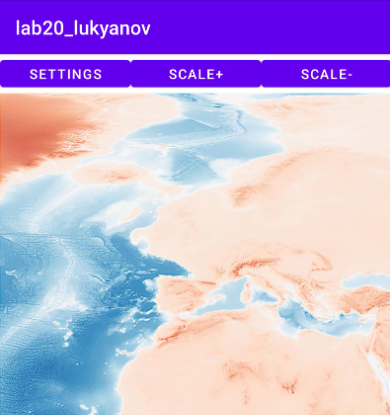


Рисунок 8 – Форма Main Form в приложении

**Форма Settings Form**

На рисунке 9 показан макет внешнего вида формы настроек.

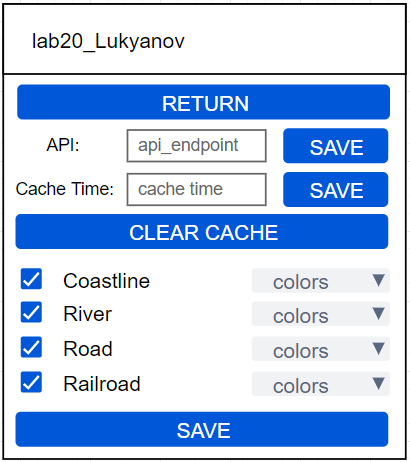


Рисунок 9 – Макет формы Settings Form

На рисунке 10 показан внешний вид основной формы в приложении.

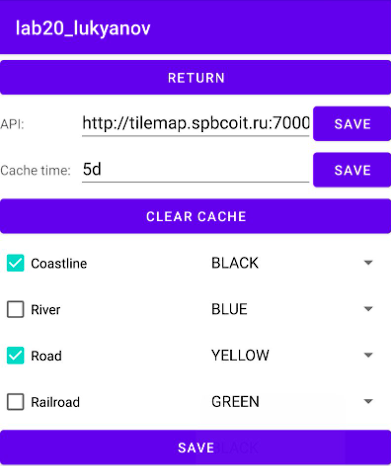


Рисунок 10 – Форма Settings Form в приложении

Описание протокола взаимодействия

Для взаимодействия с базой данных мобильное приложение использует API на основе хранимых процедур и сервера, который организует вызов процедур по протоколу HTTP и обмен данными.

Всего доступно 6 функций, список которых представлен на рисунке 11.

1. /raster – получение уровней масштабирования с характеристиками.
2. /raster/level/{x}-{y} – получение изображения тайла.
3. /coastline/{level} – получение списка точек для рисования берегов.
4. /river/{level} – получение списка точек для рисования рек.
5. /road/{level} – получение списка точек для рисования дорог.
6. /railroad/{level} – получение списка точек для рисования железных дорог.

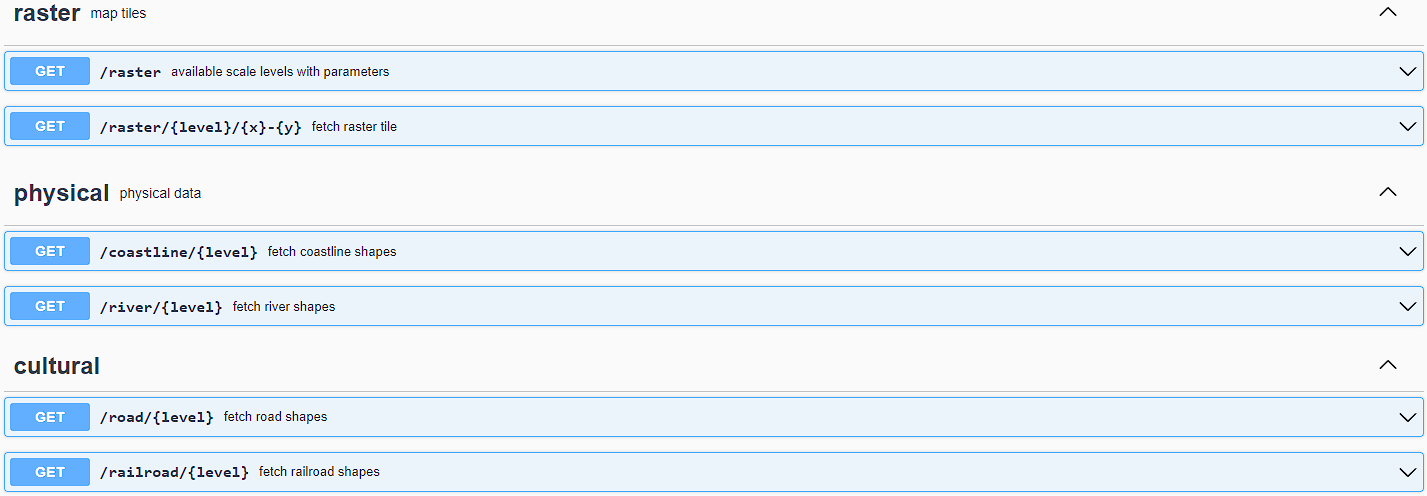


Рисунок 11 – Список функций

1. **Растровые**

Данный набор функций позволяет получить данные о тайлах карты.

* 1. **Функция /raster**

Обеспечивает возможность получения сведений о уровнях масштабирования карты.

Входных параметров нет.

Выходными параметрами будет массив уровней со следующими характеристиками:

* level – уровень масштабирования.
* xtiles – количество тайлов по горизонтали.
* ytiles – количество тайлов по вертикали.
* width – ширина тайлов.
* height – высота тайлов.
* resolution – коэффициент разрешения.

Пример вызова функции показан на рисунке 12.

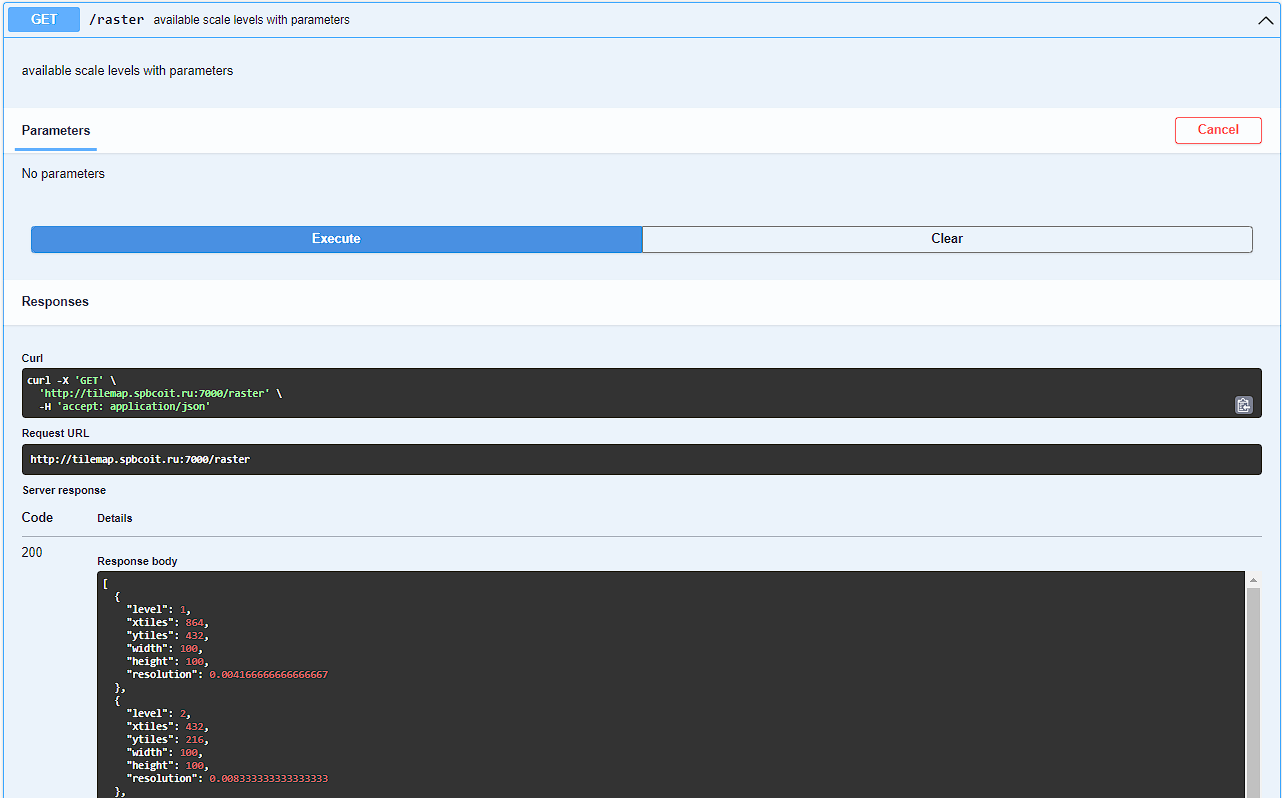


Рисунок 12 – Вызов функции /raster

* 1. **Функция /raster/level/{x}-{y}**

Обеспечивает возможность получения параметров тайла с использованием его координат и уровня масштабирования.

Входные параметры:

* level – уровень масштабирования.
* x – координата по горизонтали.
* y – координата по вертикали.

Выходные параметры:

* lat – ширина.
* lon – долгота.
* data – изображение в формате текста.

Пример вызова функции показан на рисунке 13.

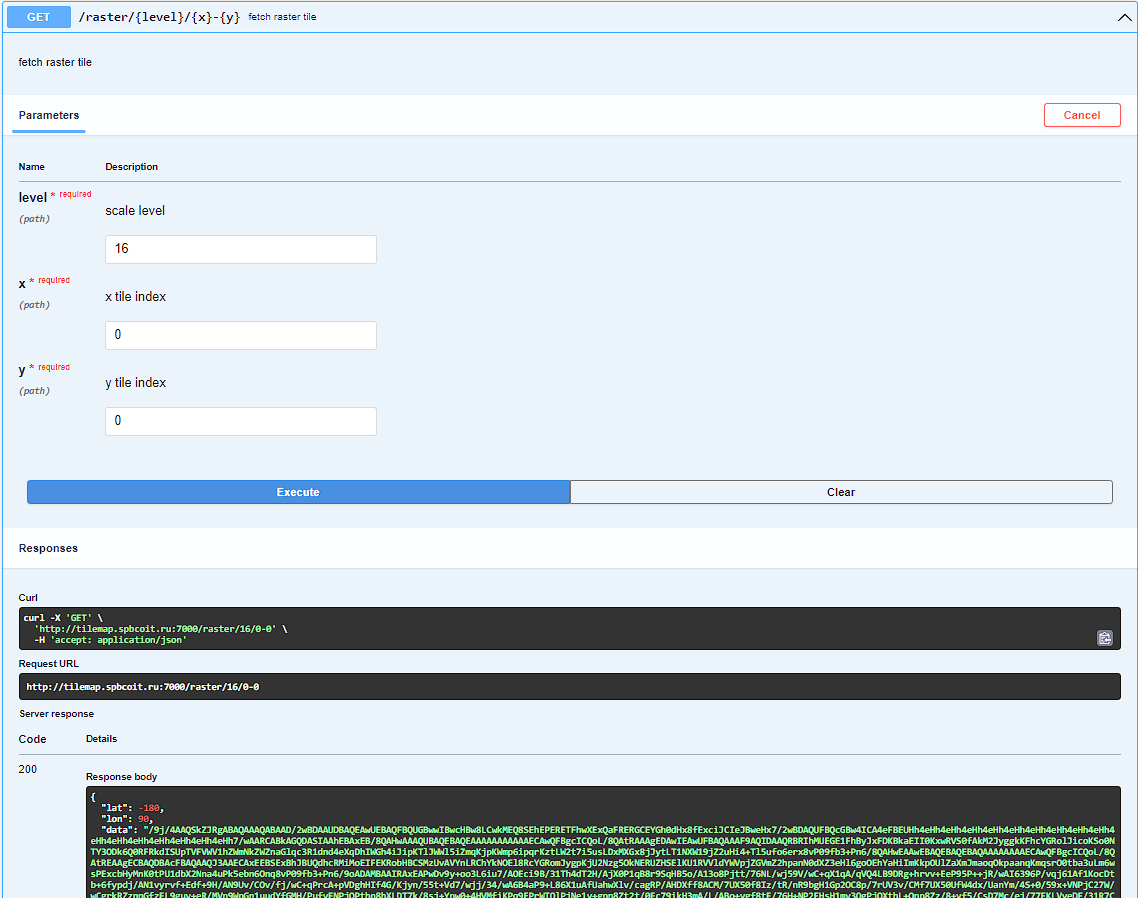


Рисунок 13 – Вызов функции /raster/level/{x}-{y}

1. **Физические**

Данный набор функций позволяет получить физические географические данные.

* 1. **Функция /coastline/{level}**

Обеспечивает возможность получения точек для рисования линий берегов.

Входные параметры:

* level – уровень масштабирования.
* lat0 – левая-верхняя ширина.
* lon0 – левая-верхняя долгота.
* lat1 – правая-нижняя ширина.
* lon1 – правая-нижняя долгота.

Выходные параметры:

* массив массивов точек по горизонтали и вертикали.

Пример вызова функции показан на рисунке 14.

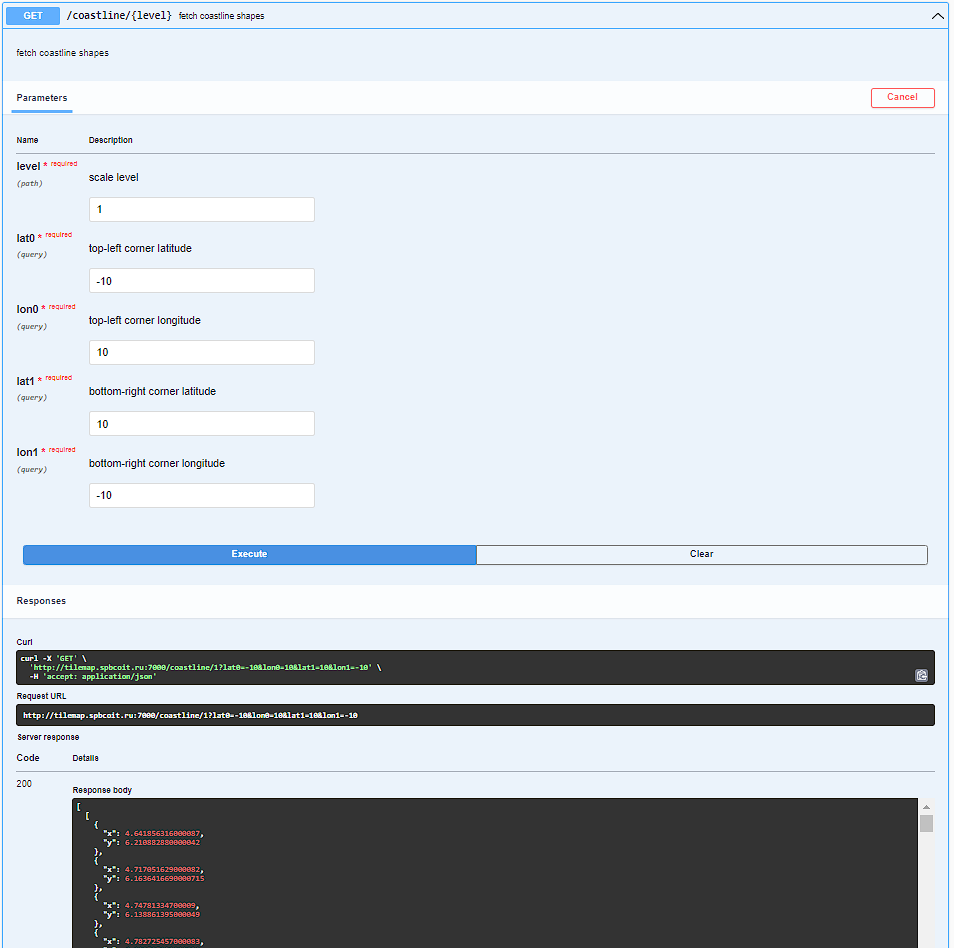


Рисунок 14 – Вызов функции /coastline/{level}

* 1. **Функция /river/{level}**

Обеспечивает возможность получения точек для рисования линий рек.

Входные параметры:

* level – уровень масштабирования.
* lat0 – левая-верхняя ширина.
* lon0 – левая-верхняя долгота.
* lat1 – правая-нижняя ширина.
* lon1 – правая-нижняя долгота.

Выходные параметры:

* массив массивов точек по горизонтали и вертикали.

Пример вызова функции показан на рисунке 15.

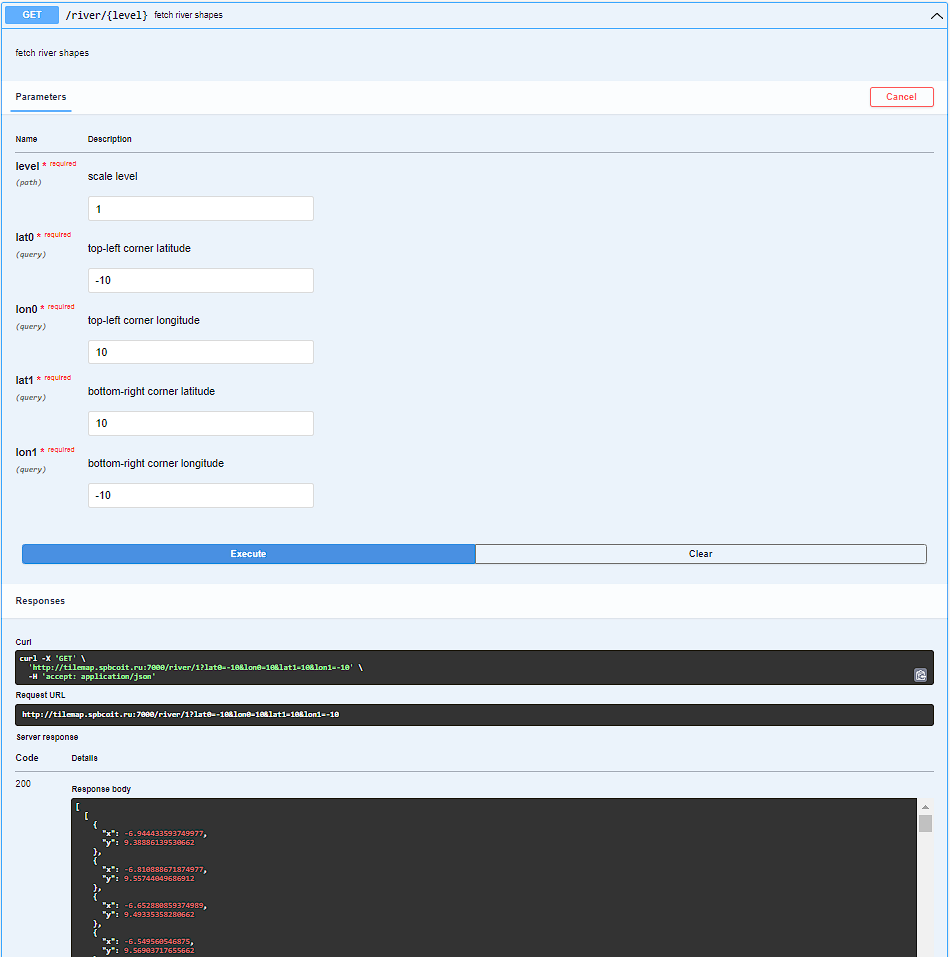


Рисунок 15 – Вызов функции /river/{level}

1. **Рукотворные**

Данный набор функций позволяет получить рукотворные географические данные.

* 1. **Функция /road/{level}**

Обеспечивает возможность получения точек для рисования линий дорог.

Входные параметры:

* level – уровень масштабирования.
* lat0 – левая-верхняя ширина.
* lon0 – левая-верхняя долгота.
* lat1 – правая-нижняя ширина.
* lon1 – правая-нижняя долгота.

Выходные параметры:

* массив массивов точек по горизонтали и вертикали.

Пример вызова функции показан на рисунке 16.

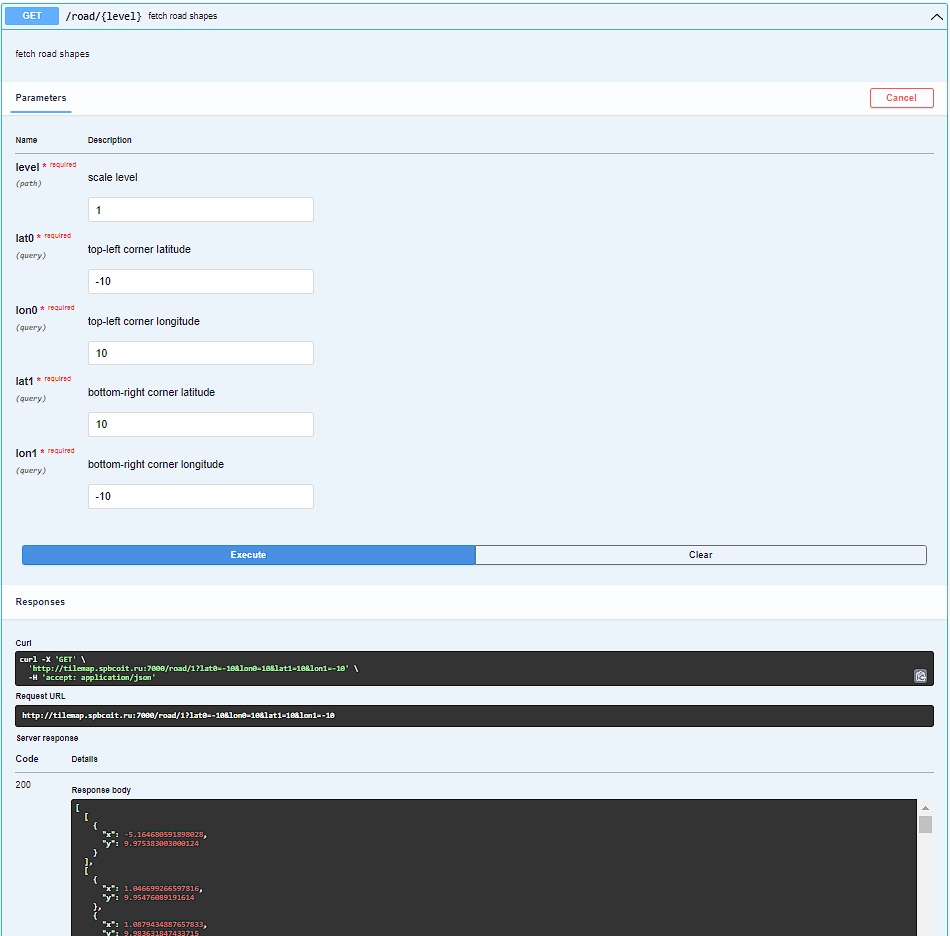


Рисунок 16 – Вызов функции /road/{level}

* 1. **Функция /railroad/{level}**

Обеспечивает возможность получения точек для рисования линий железных дорог.

Входные параметры:

* level – уровень масштабирования.
* lat0 – левая-верхняя ширина.
* lon0 – левая-верхняя долгота.
* lat1 – правая-нижняя ширина.
* lon1 – правая-нижняя долгота.

Выходные параметры:

* массив массивов точек по горизонтали и вертикали

Пример вызова функции показан на рисунке 17.

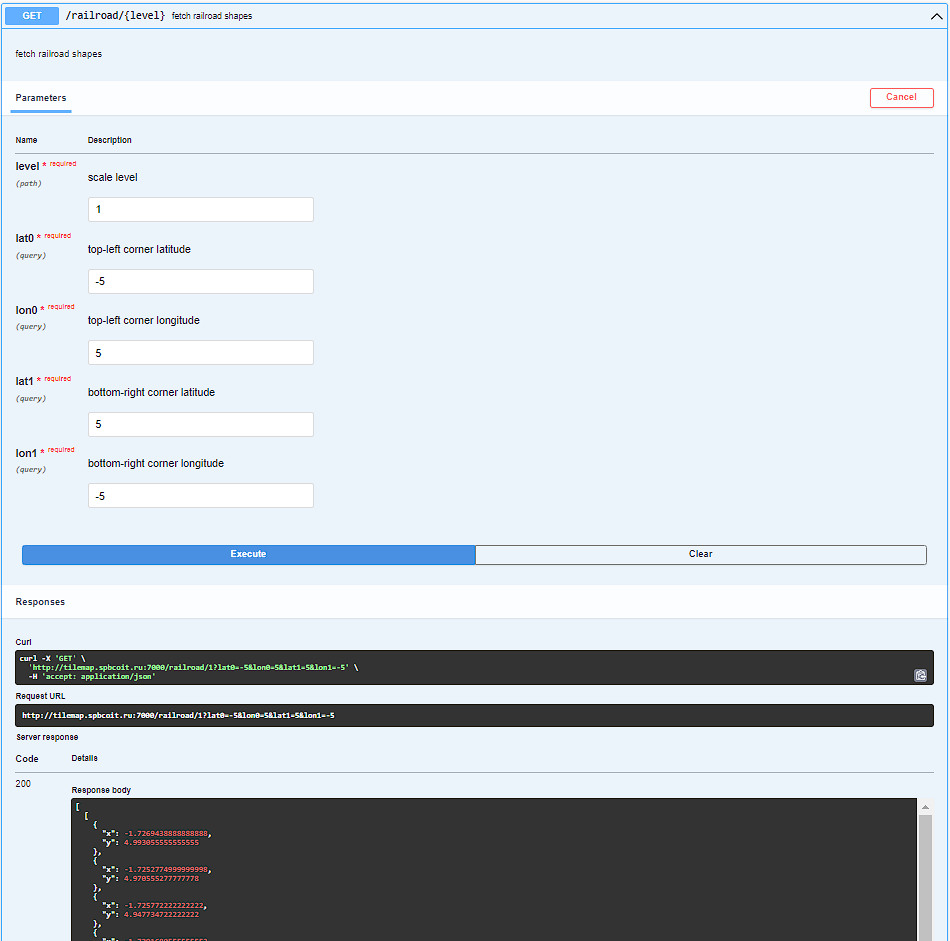


Рисунок 17 – Вызов функции /railroad/{level}

Демонстрация работы приложения

Изображения карты отображаются (рис. 18):

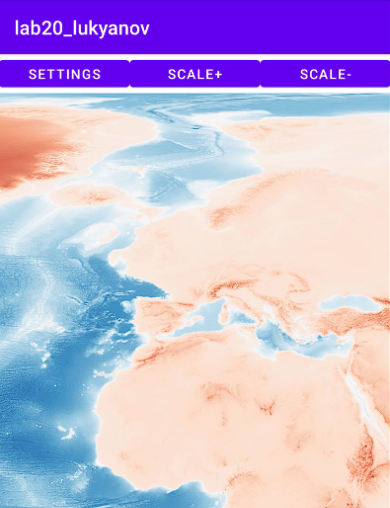


Рисунок 18 – Отображение карты

Реализовано перемещение по карте с загрузкой соответствующих тайлов (рис. 19):

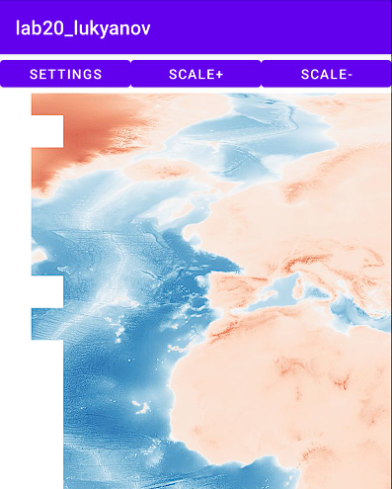


Рисунок 19 – Перемещение по карте

Имеется возможность увеличения и уменьшение масштаба, также при перезапуске приложения показывается последний уровень масштабирования и область обзора карты (рис. 20):

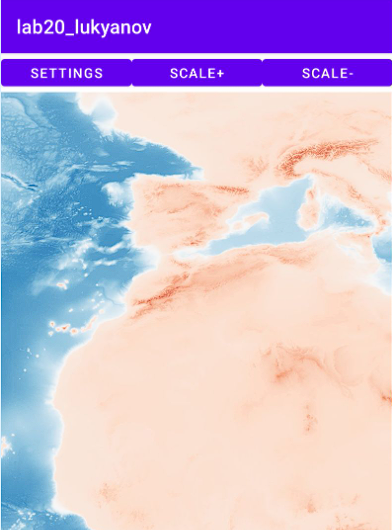


Рисунок 20 – Изменение масштаба

Имеется возможность настройки, а именно: задания точки API и времени кэширования, а также дополнительных слоев отображения (рис. 21):

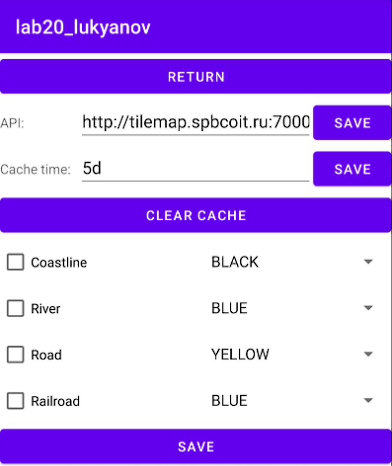


Рисунок 21 – Настройки

Изменим конечную точку API (рис. 22):

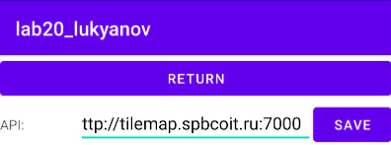


Рисунок 22 – Изменение конечной точки API

Теперь проверим загрузку новых тайлов (рис. 23):

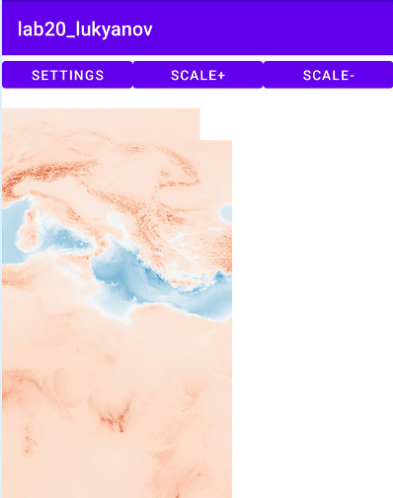


Рисунок 23 – Загрузка тайлов по новой конечной точке API

Имеется возможность настроить дополнительные слои отображения и задания им цветов (рис. 24):

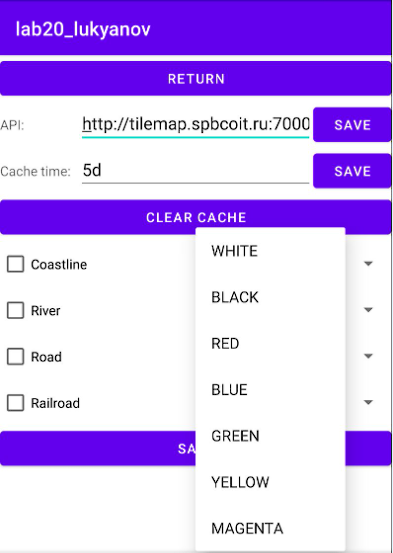


Рисунок 24 – Список доступных цветов

Выберем все слои и зададим каждому свой цвет (рис. 25):

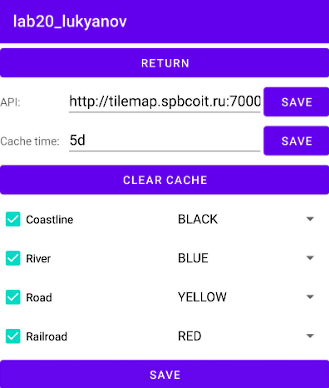


Рисунок 25 – Задание дополнительных слоев и их цветов

Теперь посмотрим их отображение на карте (рис. 26):

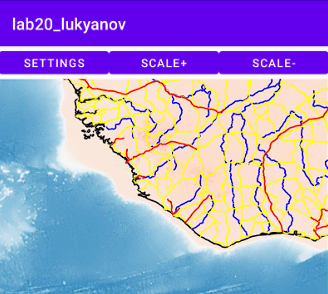


Рисунок 26 – Отображение дополнительных слоев

Приложение имеет собственную иконку (рис. 27):



Рисунок 27 – Иконка приложения