




ОНЛАЙН-ОБРАЗОВАНИЕ

Онлайн-образование



Меня хорошо видно && слышно?

Ставьте  , если все хорошо
Напишите в чат, если есть проблемы

Выпускной проект курса OTUS Java Developer. Professional

Тема: **Telegram-bot** поиска ближайшего работающего банкомата



Сергей Кубатко
skubatko@gmail.com

План защиты

Цели проекта



Что планировалось



Используемые технологии



Что получилось



Итоги

Цели проекта

1

Отработать знания, полученные в рамках курса, сделать проект на современном стеке технологий

2

Поэкспериментировать с Telegram-ботом

3

Попрактиковаться в подключении сторонних API.

4

Освоить разворачивание приложения в облаке.

Что планировалось

1

Работа над проектом была рассчитана на месяц

2

Освоить работу с Telegram-ботом

3

Попрактиковаться в подключении сторонних API

4

Попрактиковаться в разворачивании приложения в облаке (Heroku)



Используемые технологии

1

Spring Boot 2.3.8

2

Telegram-bot

3

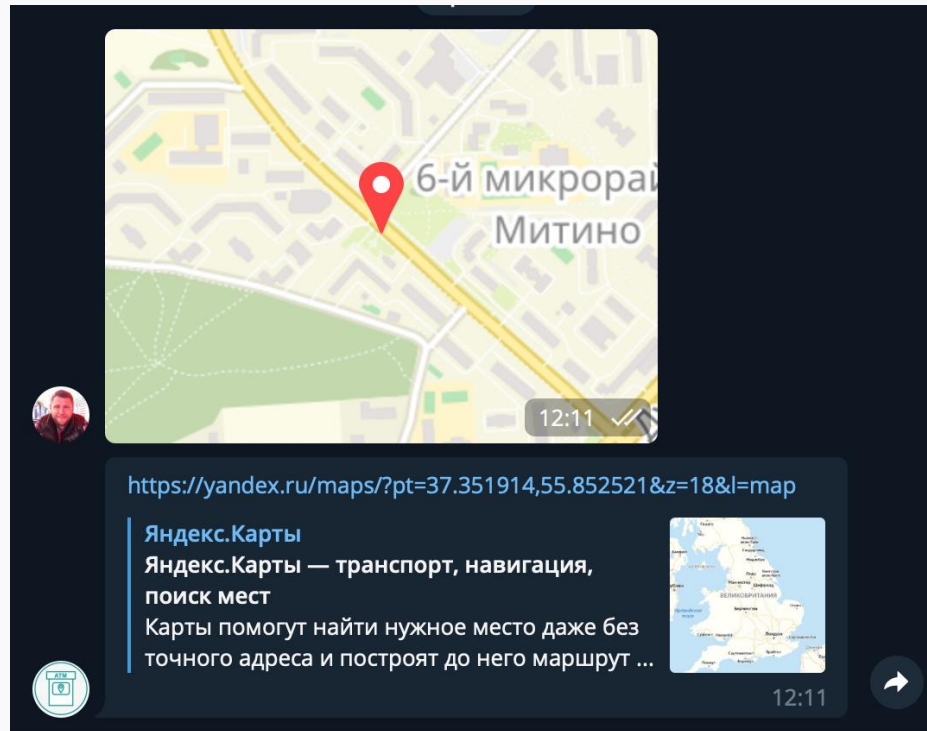
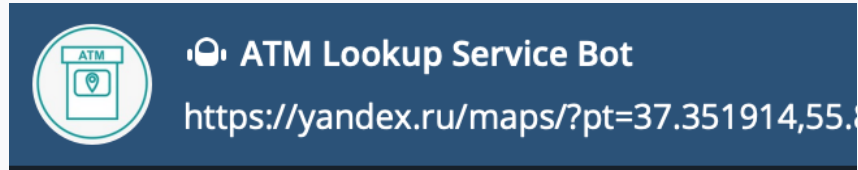
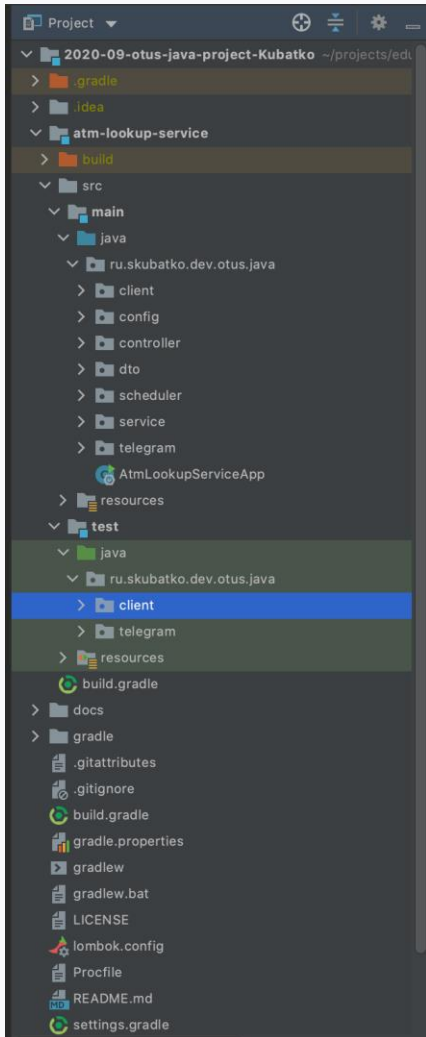
Alfa-bank API

4

Heroku



Что получилось



```
@S1f4j
@Component
@RequiredArgsConstructor
public class LocationMessageHandler implements MessageHandler {

    private final AtmService atmService;
    private final MapProps mapProps;

    private static final double EARTH_RADIUS = 6371;

    @Override
    public String handle(Message message) {
        if (message.hasLocation()) {
            val userLon :Double = message.getLocation().getLongitude();
            val userLat :Double = message.getLocation().getLatitude();
            log.trace("handle() - trace: location: longitude = {}, latitude = {}", userLon, userLat);

            val nearest :Optional<AtmDto.Coordinates> =
                atmService.getCashOutAtms().stream()
                    .map(AtmDto::getCoordinates)
                    .min((o1, o2) -> {
                        val o1Lon :Double = o1.getLongitude();
                        val o1Lat :Double = o1.getLatitude();
                        val o2Lon :Double = o2.getLongitude();
                        val o2Lat :Double = o2.getLatitude();
                        val o1distance :double = getDistance(o1Lon, o1Lat, userLon, userLat);
                        val o2distance :double = getDistance(o2Lon, o2Lat, userLon, userLat);
                        return (int) (o1distance - o2distance);
                    });

            if (nearest.isEmpty()) {
                return StringUtils.EMPTY;
            }

            val nearestCoordinates :AtmDto.Coordinates = nearest.get();
            val longitude :Double = nearestCoordinates.getLongitude();
            val latitude :Double = nearestCoordinates.getLatitude();
            return String.format(mapProps.getUrl(), longitude, latitude);
        }

        return StringUtils.EMPTY;
    }

    private double getDistance(double lon1, double lat1, double lon2, double lat2) {
        double sin1 = sin((lat1 - lat2) / 2);
        double sin2 = sin((lon1 - lon2) / 2);
        return 2 * EARTH_RADIUS * asin(sqrt(sin1 * sin1 + sin2 * sin2 * cos(lat1) * cos(lat2)));
    }
}
```


Итоги

1

Поставленные цели достигнуты

2

Найден стартёр для работы с Telegram-ботом

3

Исследовал подключение к стороннему API
с применением SSL/сертификатов

4

Пришло понимание как работать с Heroku



Спасибо за внимание!



Сергей Кубатко
skubatko@gmail.com

Репозиторий проекта: <https://github.com/skubatko/2020-09-otus-java-project-Kubatko>