

# МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

**«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИНЖЕНЕРНЫХ ТЕХНОЛОГИЙ»**

# МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

**«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИНЖЕНЕРНЫХ ТЕХНОЛОГИЙ»**

# МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

**«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИНЖЕНЕРНЫХ ТЕХНОЛОГИЙ»**

# МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

**«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИНЖЕНЕРНЫХ ТЕХНОЛОГИЙ»**

**Факультет управления и информатики в технологических системах**

**Кафедра**[**корпоративных информационных систем и программирования**](https://vsuet.ru/obuchenie/faculties/uits/k_kisip)

**Направление подготовки 09.03.02**

***(шифр)***

**Информационные системы и технологии**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

*(наименование направления подготовки)*

**Отчет**

**по практическому заданию № 5**

Выполнил студент гр. У-213

Боровкова Анастасия Дмитриевна

(ф.и.о)

**\_\_\_\_\_\_\_\_\_**\_\_\_\_\_\_\_\_\_\_\_\_\_\_

***(****подпись****)***

Проверил(а):

Демичев С.Е.

**\_\_\_\_\_\_\_\_\_**\_\_\_\_\_\_\_\_\_\_\_\_\_\_

***(****подпись****)***

Воронеж – 2023

public class FcmClient {

public FcmClient(FcmSettings settings) {

Path p = Paths.get(settings.getServiceAccountFile());

try (InputStream serviceAccount = Files.newInputStream(p)) {

FirebaseOptions options = new FirebaseOptions.Builder()

.setCredentials(GoogleCredentials.fromStream(serviceAccount))

.build();

FirebaseApp.initializeApp(options);

} catch (IOException e) {

Logger.getLogger(FcmClient.class.getName())

.log(Level.SEVERE, null, e);

}

}

public String sendByTopic(PushNotifyConf conf, String topic)

throws InterruptedException, ExecutionException {

Message message = Message.builder().setTopic(topic)

.setWebpushConfig(WebpushConfig.builder()

.putHeader("ttl", conf.getTtlInSeconds())

.setNotification(createBuilder(conf).build())

.build())

.build();

String response = FirebaseMessaging.getInstance()

.sendAsync(message)

.get();

return response;

}

public String sendPersonal(PushNotifyConf conf, String clientToken)

throws ExecutionException, InterruptedException {

Message message = Message.builder().setToken(clientToken)

.setWebpushConfig(WebpushConfig.builder()

.putHeader("ttl", conf.getTtlInSeconds())

.setNotification(createBuilder(conf).build())

.build())

.build();

String response = FirebaseMessaging.getInstance()

.sendAsync(message)

.get();

return response;

}

public void subscribeUsers(String topic, List<String> clientTokens)

throws FirebaseMessagingException {

for (String token : clientTokens) {

TopicManagementResponse response = FirebaseMessaging.getInstance()

.subscribeToTopic(Collections.singletonList(token), topic);

}

}

private WebpushNotification.Builder createBuilder(PushNotifyConf conf){

WebpushNotification.Builder builder = WebpushNotification.builder();

builder.addAction(new WebpushNotification

.Action(conf.getClick\_action(), "Открыть"))

.setImage(conf.getIcon())

.setTitle(conf.getTitle())

.setBody(conf.getBody());

return builder;

}

}