

Создание библиотеки для http запросов на языке программирования Java

Жуков Дмитрий Сергеевич

Приамурский государственный университет им. Шолом-Алейхема

Студент

Аннотация

Целью данного исследования является создание библиотеки для http запросов на языке программирования Java.

Ключевые слова: http, библиотека, запрос, Java.

Creating a library for http requests in the Java programming language

Zhukov Dmitry Sergeevich

Sholom-Aleichem Priamursky State University

Student

Abstract

The purpose of this research is to create a library for http requests in the java programming language.

Keywords: http, library, request, java.

На рынке информационных услуг и различных программ ежедневно появляются различные новинки. Для создания программ используется большое количество языков программирования. Создавая программу, программист стоит перед выбором наилучшего языка. Язык Java - один из самых популярных языков на сегодняшний день по многим характеристикам. Созданные программы на этом языке могут использоваться в различных сферах деятельности.

С.С.Петрищев в своей статье рассмотрел применение скриптового языка java для сокращения объёма кода с целью повышения наглядности и читаемости, а также с целью сокращения времени на разработку приложения [1]. Е.Д.Лакомкин в своей статье проводит анализ проблем создания приложений реального времени на Java. Также приводятся базовые техники и методы достижения детерминизма и описываются некоторые полезные библиотеки, которые позволяют создавать быстрые, предсказуемые приложения [2]. А.Н.Гребнев. в своей работе рассматривает спроектированное автором и разработанное под руководством автора Web-приложение AtLeap-Java-каркас с открытым исходным кодом для быстрого построения на его основе Web-приложений. Объясняются причины выбора основных библиотек, используемых при построении. Особое внимание уделено рассмотрению основной функциональности - AtLeap-системе

управления содержимым на основе концепции, разработанной автором [3]. Не менее значимы иностранные исследования в данной сфере [4-5].

Целью данного исследования является создание библиотеки для http запросов на языке программирования java.

Разработку библиотеки следует начать с создания интерфейса для обработки ответа на Web запрос.

```
public interface RespListener {  
    void onResponse(String response);  
}
```

Рис.1. Интерфейс для обработки ответа на Web запрос

Создаем класс MyWebReq, в этом классе объявим конструкторы для двух разных видов использования библиотеки (с отслеживанием ошибки и без). Также для удобства использования библиотеки создадим методы добавления данных в http запрос (Рис.2).

```
public class MyWebReq extends StringRequest {  
    private final ArrayList<String> keys;  
    private final ArrayList<String> values;  
  
    public MyWebReq(String url, final RespListener respListener) {  
        super(Method.GET, url, respListener::onResponse, errorListener: null);  
        keys = new ArrayList<>();  
        values = new ArrayList<>();  
    }  
  
    public MyWebReq(int method, String url, final RespListener listener, final Response.ErrorListener errorListener){  
        super(method, url, listener::onResponse, errorListener);  
        keys = new ArrayList<>();  
        values = new ArrayList<>();  
    }  
  
    public void addField(String key, String value){  
        keys.add(key);  
        values.add(value);  
    }  
  
    public void addList(String key, ArrayList<String> value){  
        for(String aValue : value) addField(key, aValue);  
    }  
  
    @Override  
    public byte[] getBody() {  
        if (keys.size() > 0) {  
            StringBuilder builder = new StringBuilder("");  
            for (int i=0;i<keys.size();i++) {  
                builder.append("&").append(keys.get(i));  
                builder.append("=").append(values.get(i));  
            }  
            if (builder.length() > 0) { // ! isEmpty()  
                builder.deleteCharAt(0);  
            }  
            return builder.toString().getBytes();  
        }  
        return null;  
    }  
}
```

Рис.2. Класс MyWebReq

Опишем метод добавления hash суммы в данные запросы. Это необходимо для, например, отправки запроса в платежную систему (Рис.3).

```
public void addHashField(){
    if (keys.size() == 0){
        addRandomField();
    }
    StringBuilder origin = new StringBuilder();
    for (int i=0; i<keys.size();i++){
        if (i!=0)
            origin.append('&');
        origin.append(keys.get(i)).append('=').append(values.get(i));
    }
    origin.append(BuildConfig.API_SECRET_KEY);
    addField("hash", DigestUtils.sha256Hex(origin.toString()));
}
```

Рис.3. Метод добавления hash суммы

Пример использования библиотеки показан на рисунке 4.

```
RespListener respListener2 = response -> {
    Log.d( tag: "response", response);
    //response - тело ответа
    //Обработка ответа на запрос
};
Response.ErrorListener errorListener2 = error -> {
    NetworkResponse networkResponse = error.networkResponse;
    //Обработка ошибки
};
String url = "http://e-postulat.ru";
MyWebReq post = new MyWebReq(ReqMethod.POST, url, respListener, errorListener);
post.addField("key1", "data1");
post.addField("key2", "data2");
post.addHashField();
RequestQueue queue = Volley.newRequestQueue( context: this);
queue.add(post);
```

Рис.4. Пример использования библиотеки

Таким образом, была разработана библиотека для простой и удобной работы с http запросами.

Библиографический список

1. Петрищев С.С. Проблемы производительности разработки java приложений // Международный научно-исследовательский журнал. 2012.

- № 7-1 (7). С. 58-60.
2. Лакомкин Е.Д. Методы и практики проектирования web-приложений реального времени с использованием технологии java // RSDN Magazine. 2012. № 1. С. 34-38.
 3. Гребнев А.Н. Atleap-java-каркас с открытым исходным кодом для web-приложений // Вестник Ижевского государственного технического университета. 2006. № 1. С. 64-68.
 4. Smith A. W. et al. Application program interface that enables communication for a network software platform : пат. 7117504 США. 2006.
 5. Parikh V., Moore R., Cheng H. Application program interface for a graphics system : пат. 6456290 США. 2002.