



PUSH APP SERVER
HTTP GATEWAY
Versão 1.2

Índice

Introdução	2
Plataforma técnica	2
Pedido de criação de campanha PUSH APP	3
URL	3
Método	3
Parâmetros	3
Obrigatório	3
name (<i>string</i>)	3
type (<i>string</i>)	3
groups (<i>array</i>)	3
apps (<i>array de arrays</i>)	3
os (<i>array de arrays</i>)	4
Android	4
iOS	5
Opcional	5
schedule (<i>object</i>)	5
date (<i>string</i>)	5
timezone (<i>string</i>)	5
batch (<i>object</i>)	6
parts (<i>integer</i>)	6
minutes (<i>integer</i>)	6
tags (<i>array de strings</i>)	6
geolocation (<i>objeto</i>)	6
latitude (<i>float</i>)	6
longitude (<i>float</i>)	6
radius (<i>integer</i>)	6
segmentation (<i>array de objetos</i>)	7
field (<i>string</i>)	7
condition (<i>string</i>)	7
value (<i>string</i>)	8
Exemplo de pedido PHP	9
Exemplo de pedido PYTHON	10
Exemplo de pedido JAVA	11
Exemplo de pedido C#	13
Respostas	16
201 Pedido correto. Campanha criada.	16

422 Erro nos parâmetros.	16
400 Erro ao criar a campanha.	16

Introdução

A plataforma http server permite ao utilizador criar uma campanha de envio de mensagens PUSH app através desta plataforma. Para poder aceder às suas estatísticas e dados de faturação pode aceder ao site <https://www.360nrs.com> com os seus dados de utilizador.

A comunicação com o cliente não se realizará por meio de qualquer API fornecida pela empresa, mas será realizada simplesmente uma comunicação HTTP com alguns parâmetros para o URL indicado.

Este processo será detalhado a seguir.

Plataforma técnica

Cada pedido efetuado deverá incluir no respetivo cabeçalho do pedido http a autenticação do cliente. Para isso, usa-se a autenticação de acesso básica de HTTP.

O cabeçalho de autorização constrói-se combinando a sequência "utilizador: palavra-passe" e codificando-a em base64. A esta sequência antepõe-se a sequência "Authorization: Basic"

Por exemplo, para o utilizador "myuser" e palavra-passe "mypass" o cabeçalho resultante seria:

Autorização: Basic bWl1c2VyOm1pcGFzcw==

Em seguida, especificam-se as opções de envio disponíveis, o URL a que se deve chamar e os parâmetros que admite.

Pedido de criação de campanha PUSH APP

URL

<https://dashboard.360nrs.com/api/rest/app-push>

Método

POST

Parâmetros

Obrigatório

name (*string*)

Nome da campanha

type (*string*)

Tipo de campanha. Opções:

- basic

groups (*array*)

Identificadores dos grupos de contactos da plataforma 360nrs a que se quer enviar a campanha PUSH app.

apps (*array de arrays*)

Identificadores das aplicações que temos disponíveis na plataforma 360nrs a que se quer enviar a campanha PUSH app.

os (*array de arrays*)

Dados de composição da mensagem PUSH para os diferentes sistemas operativos (*Android e iOS*).

Dependendo do tipo escolhido de notificação, deverão especificar-se alguns campos ou outros.

Android

notification_type

Este é o campo que nos indicará o tipo de notificação a enviar, pelo que é sempre obrigatório. É do tipo string e as opções são:

- basic
- advanced
- alertbox
- intersticial

Dependendo do tipo de notificação, os campos obrigatórios são os seguintes:

basic

- title (*string*) (*máximo 30 caracteres*)
- subtitle (*string*) (*máximo 100 caracteres*)

advanced

- title (*string*) (*máximo 30 caracteres*)
- subtitle (*string*) (*máximo 100 caracteres*)

alertbox

- title (*string*) (*máximo 30 caracteres*)
- subtitle (*string*) (*máximo 100 caracteres*)
- buttonText (*string*) (*máximo 14 caracteres*)
- url (*string*)

intersticial

- image (*url ou base64*)

iOS

notification_type

Este é o campo que nos indicará o tipo de notificação a enviar, pelo que é sempre obrigatório. É do tipo string e as opções são:

- basic
- advanced
- alertbox

Dependendo do tipo de notificação, os campos obrigatórios são os seguintes:

basic

- title (*string*) (*máximo 30 caracteres*)
- subtitle (*string*) (*máximo 100 caracteres*)
- url (*string*)

advanced

- title (*string*) (*máximo 30 caracteres*)
- subtitle (*string*) (*máximo 100 caracteres*)

alertbox

- title (*string*) (*máximo 30 caracteres*)
- subtitle (*string*) (*máximo 100 caracteres*)
- buttonText (*string*) (*máximo 14 caracteres*)
- url (*string*)

Opcional

schedule (*object*)

Objeto com a programação de envio numa data futura:

date (*string*)

Data de envio com o formato 'YYYY-mm-dd HH:ii:ss' (*MySQL*).

timezone (*string*)

Zona horária que se deseja ter em conta para a hora de envio.

Por exemplo, "Europa/Madrid".

batch (object)

Objeto com a configuração de em quantas partes se quer dividir o envio e a cada quantos minutos se quer enviar.

parts (integer)

Número de partes em que se quer dividir o envio.

minutes (integer)

A cada quantos minutos, a partir da data de início, se deseja enviar as partes do envio.

tags (array de strings)

Array de tags que se queiram aplicar à campanha.

Por exemplo,

['blackfriday', 'saldos']

geolocation (object)

Objeto com a configuração da geolocalização da campanha.

latitude (float)

Coordenada de latitude a que se deseja restringir o envio. Sendo uma latitude, deve estar entre -90 e 90.

longitude (float)

Coordenada de longitude a que se deseja restringir o envio. Sendo uma longitude, deve estar entre -90 e 90.

radius (integer)

Dado o ponto com a latitude e longitude, em que raio **em metros** se deseja que as mensagens cheguem. Deve ser superior a 10 metros e inferior a 99999 metros.

segmentation (*array de objetos*)

Array de condições que se devem cumprir para que a mensagem chegue ao utilizador. Cada elemento do array é um objecto que tem as seguintes propriedades:

field (*string*)

Nome do dado pelo qual se deseja alcançar. Estão disponíveis os seguintes dados:

- name
- surname
- gender
- age

condition (*string*)

Condição que o campo deve cumprir para ser entregue. As condições disponíveis dependerão do campo escolhido.

- name
 - diff-to
 - equal-to
 - contains
 - start-with
 - end-with
- surname
 - diff-to
 - equal-to
 - contains
 - start-with
 - end-with
- gender
 - male
 - female
- age
 - diff-to
 - equal-to
 - greater-than
 - greater-than-or-equal-to
 - lower-than
 - lower-than-or-equal-to

value (*string*)

O valor que o dado do utilizador deve ter comparando-o usando a condição enviada em condition. As comparações de texto não têm em conta as maiúsculas (*case insensitive*).

Por exemplo (*javascript*)

```
{  
  field: "name",  
  condition: "equal-to",  
  value: "Jose"  
}
```

Exemplo de Pedido Php

```
<?php

$post["name"]    = "Nome da Campanha";
$post["type"]    = "basic";
$post["groups"]  = [2579];
$post["apps"]    = [
    "android" => [310]
];
$post["os"] = [
    "android" => [
        "notification_type" => "advanced",
        "title" => "titulo",
        "subtitle" => "legenda",
        "image" => "https://www.w3schools.com/w3css/img_lights.jpg",
        "buttonText" => "texto do botão an",
        "url" => "https://www.w3schools.com/w3css/img_lights.jpg",
        "options" => [
            "vibrate" => true,
            "sound" => true
        ]
    ],
    "ios" => [
        "notification_type" => "advanced",
        "title" => "titulo",
        "subtitle" => "legenda",
        "image" => "https://www.w3schools.com/w3css/img_lights.jpg",
        "buttonText" => "texto do botão ios",
        "url" => "https://www.apple.es",
        "options" => [
            "vibrate" => true,
            "sound" => true
        ]
    ]
];

$user      = "miuser";
$password  = "mipass";

try {
    $ch      = curl_init();
    curl_setopt($ch, CURLOPT_URL,
        "https://dashboard.360nrs.com/api/rest/app-push");
    curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
    curl_setopt($ch, CURLOPT_POST, 1);
    curl_setopt($ch, CURLOPT_POSTFIELDS, json_encode($post));
    curl_setopt($ch, CURLOPT_SSL_VERIFYHOST, 0);
```

```

curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, 0);
curl_setopt($ch, CURLOPT_HTTPHEADER,
    array(
        "Content-type: application/json",
        "Authorization: Basic ".base64_encode($user.":".$password));
$result = curl_exec($ch);
var_dump($result);
} catch (Exception $exc) {
    echo $exc->getTraceAsString();
}

```

Exemplo de Pedido Python

```

import pycurl
import base64
import json

if __name__ == "__main__":

    url = "https://dashboard.360nrs.com/api/rest/app-push"
    usrPass = "miuser:mipass"

    data=json.dumps({
        "name":"Nome da Campanha",
        "type":"basic",
        "groups":[
            101
        ],
        "apps":{
            "android":[
                684,735
            ]
        },
        "os":{
            "android":{
                "notification_type": "advanced",
                "title": "titulo",
                "subtitle": "legenda",
                "image": "https://www.w3schools.com/w3css/img_lights.jpg",
                "buttonText": "texto do botão an",
                "url": "https://www.android.com",
                "options": {
                    "vibrate": True,
                    "sound": True
                }
            },
            "ios": {
                "notification_type": "advanced",
                "title": "titulo",

```

```

        "subtitle": "legenda",
        "image": "https://www.w3schools.com/w3css/img_lights.jpg",
        "buttonText": "texto do botão ios",
        "url": "https://apple.es",
        "options": {
            "vibrate": True,
            "sound": True
        }
    })
b64Val = base64.b64encode(usrPass)
headers=[
    "Content-type: application/json",
    "Authorization:Basic %s"%b64Val]

    c = pycurl.Curl()
    c.setopt(pycurl.URL, url)
    c.setopt(pycurl.HTTPHEADER,headers)
    c.setopt(pycurl.POST, 1)
    c.setopt(pycurl.POSTFIELDS, data)
    c.setopt(pycurl.SSL_VERIFYHOST, 0)
    c.setopt(pycurl.SSL_VERIFYPEER, 0)
    c.perform()

    http_code = c.getinfo(pycurl.HTTP_CODE)
    print(http_code)

```

Exemplo de Pedido Java

```

import java.io.BufferedReader;
import java.io.DataOutputStream;
import java.io.IOException;
import java.io.InputStreamReader;
import java.net.URL;
import javax.net.ssl.HttpsURLConnection;
import org.json.simple.JSONArray;
import org.json.simple.JSONObject;

public class ApiPushApp {

    public static void main(String args[]) throws IOException {
        String url = "https://dashboard.360nrs.com/api/rest/app-push";
        URL obj = new URL(url);
        HttpsURLConnection con = (HttpsURLConnection) obj.openConnection();

        con.setRequestMethod("POST");

```

```

String userpass = "miuser" + ":" + "mipass";
String basicAuth = "Basic " +
javax.xml.bind.DatatypeConverter.printBase64Binary(userpass.getBytes("UTF-8"));
con.setRequestProperty("Authorization", basicAuth);
con.setRequestProperty("Content-type", "application/json");

JSONObject data = new JSONObject();
JSONArray groups = new JSONArray();
JSONObject android = new JSONObject();
JSONArray listAppsAndroid = new JSONArray();
JSONObject notificationAndroid = new JSONObject();
JSONObject notificationIos = new JSONObject();
JSONObject osList = new JSONObject();
JSONObject optionsAndroid = new JSONObject();
JSONObject optionsIos = new JSONObject();

data.put("name", "Nome da Campanha");
data.put("type", "basic");

groups.add(2579);
data.put("groups", groups);

listAppsAndroid.add(310);
android.put("android", listAppsAndroid);
data.put("apps", android);

notificationAndroid.put("notification_type", "advanced");
notificationAndroid.put("title", "titulo");
notificationAndroid.put("subtitle", "legenda");
notificationAndroid.put("image",
"https://www.w3schools.com/w3css/img_lights.jpg");
notificationAndroid.put("buttonText", "texto do botão an");
notificationAndroid.put("url", "https://www.android.com");
optionsAndroid.put("vibrate", true);
optionsAndroid.put("sound", true);
notificationAndroid.put("options", optionsAndroid);

notificationIos.put("notification_type", "advanced");
notificationIos.put("title", "titulo");
notificationIos.put("subtitle", "legenda");
notificationIos.put("image",
"https://www.w3schools.com/w3css/img_lights.jpg");
notificationIos.put("buttonText", "texto do botão ios");
notificationIos.put("url", "https://apple.es");
optionsIos.put("vibrate", true);
optionsIos.put("sound", true);
notificationIos.put("options", optionsIos);

osList.put("android", notificationAndroid);
osList.put("ios", notificationIos);

```

```

        data.put("os", osList);

        String jsonText = data.toJSONString();

        con.setDoOutput(true);
        DataOutputStream wr = new DataOutputStream(con.getOutputStream());
        wr.writeBytes(jsonText);
        wr.flush();
        wr.close();

        BufferedReader in = new BufferedReader(
            new InputStreamReader(con.getInputStream()));
        String inputLine;
        StringBuffer response = new StringBuffer();

        while ((inputLine = in.readLine()) != null) {
            response.append(inputLine);
        }
        in.close();

        System.out.println(response.toString());
    }
}

```

EXemplo de pedido C#

```

using Newtonsoft.Json;
using System;
using System.IO;
using System.Net;

namespace nrs_api
{
    class Program
    {
        static void Main(string[] args)
        {
            var httpRequest =
            (HttpWebRequest)WebRequest.Create("https://dashboard.360nrs.com/api/rest/app-pus
h");

            httpRequest.Method = "POST";
            httpRequest.ContentType = "application/json";
            String username = "miuser";
            String password = "mipass";
            String encoded =
            System.Convert.ToBase64String(System.Text.Encoding.GetEncoding("ISO-8859-1").Get
Bytes(username + ":" + password));

```

```

        httpRequest.Headers.Add("Authorization", "Basic " + encoded);

        using (var streamWriter = new
StreamWriter(httpWebRequest.GetRequestStream()))
        {

            var name = "Nome da Campanha";
            var type = "basic";
            string[] groups = new string[1];
            groups[0] = "101";
            string[] appsAndroid = new string[1];
            appsAndroid[0] = "684";

            var apps = new
            {
                android = appsAndroid
            };

            var optionsAndroid = new
            {
                vibrate = true,
                sound = true
            };

            var notificationAndroid = new
            {
                notification_type = "advanced",
                title = "titulo",
                subtitle = "legenda",
                image = "https://www.w3schools.com/w3css/img_lights.jpg",
                buttonText = "texto do botão an",
                url = "https://www.android.com",
                options = optionsAndroid
            };

            var optionsIos = new
            {
                vibrate = true,
                sound = true
            };

            var notificaronIos = new
            {
                notification_type = "advanced",
                title = "titulo",
                subtitle = "legenda",
                image = "https://www.w3schools.com/w3css/img_lights.jpg",
                buttonText = "texto do botão ios",
                url = "https://www.apple.es",
                options = optionsIos
            };

```

```

    };

    var os = new
    {
        android = notificationAndroid,
        ios = notificarionIos
    };

    var data = new
    {
        name = name,
        type = type,
        groups = groups,
        apps = apps,
        os = os
    };

    string json = JsonConvert.SerializeObject(data);

    streamWriter.Write(json);
    streamWriter.Flush();
    streamWriter.Close();
}

var httpResponse = (HttpWebResponse)httpWebRequest.GetResponse();
using (var streamReader = new
StreamReader(httpResponse.GetResponseStream()))
{
    var result = streamReader.ReadToEnd();
    Console.WriteLine(result);
    Console.ReadKey();
}
}
}
}

```


Respostas

201 Pedido correto. Campanha criada.

422 Erro nos parâmetros.

Especificar-se-á que parâmetro ou parâmetros falharam na resposta.

400 Erro ao criar a campanha.