

REST API: Add User Image

Version 7

Published 1/29/2016 10:34 AM by [Ravi Rawal](#) Last updated 6/7/2019 10:33 AM by [Aparna Anand](#)

Add a user image

PUT /api/users/{userid}/profile/image

Add a profile/avatar image for a user. Valid for Communifire version 5.2 and above.

Method Details

HTTP Method

Response Format

JSON

Requires Authentication?

YES

Product Version

5.2 and above

Required Parameters	How to use	Description
API Key	rest-api-token: Used in the request URL or set in the POST data	Communifire REST API key for the Communifire portal you are making the call for.
Image Stream	Used in the request body.	Here you have to send the byte stream of the image you want to set as profile, avatar image.
UserID	Used in the request URL (see below).	Unique identifier for a particular user.
FileName	Used in the request URL (see below).	Name of the image.

Example URL to POST: <https://your-community.com/api/users/56/profile/image?filename=windows.jpg>

Response JSON sample where indicator (true/false) is returned as ResponseData

```
{
  "IsError":false,
  "ResponseMessage": "",
  "ResponseData":true
}
```

Please Note Your content type that you pass in the header of your request should be '**application/octet-stream**'.

C# Sample Code

```
public static void UpdateUserImage()
{
```

```

string filePath = @"C:\[FILENAME]";

string postData = filePath;
string serviceUrl = "[CommuniFireURL]/api/users/1/profile/image?filename=[FILENAME]";

// Create a new HttpRequest
var myRequest = (System.Net.HttpWebRequest)WebRequest.Create(serviceUrl);

myRequest.Method = "PUT";
// Set the content header type. Note: use "application/octet-stream" for file data
myRequest.ContentType = "application/octet-stream";

// Add the API key
myRequest.Headers.Add("Rest-Api-Key", "[APIKEY]");

// Add the data to be posted in the request stream
if (postData != null)
{
    byte[] fileDataInByte;

    fileDataInByte = File.ReadAllBytes(postData);
    //set request stream (Content)
    using (var requestStream = myRequest.GetRequestStream())
    {
        requestStream.Write(fileDataInByte, 0, fileDataInByte.Length);
    }
}

// Post the request and get the response details
using (var response = myRequest.GetResponse())
{
    if (response.ContentLength > 0)
    {
        {
            using (var reader = new StreamReader(response.GetResponseStream()))
            {
                //check the results assuming XML is returned: note for JSON: use JSON stringfy
                string result = reader.ReadToEnd();
                Console.WriteLine(result);
            }
        }
    }
}
}

```

tags : API, users, image, put, rest, site-administrator