

# ContactFormRequest Service

## Description

This service is used to simplify entering basic web forms into Atollon. In Atollon, it's possible to add more complex business logic based on request type and it's parameters. Web developers are let go of understanding Atollon blackbox.

## Request

```
POST https://<HOST>/crab/contact_form/<request_type>?instance=<instance_name>
```

New URL format:

```
POST https://<HOST>/svc/contact_form/<request_type>?instance=<instance_name>
```

// Consider removing instance

## Request Types

### New Lead

```
https://<HOST>/crab/contact_form/lead?instance=<instance_name>
```

New URL format:

```
POST https://<HOST>/svc/contact_form/lead?instance=<instance_name>
```

// Consider removing instance

POST data in JSON format:

```
{
  "person": {
    "firstName": "<jmeno>",
    "lastName": "<prijmeni>"
  },
}
```

```

"company": {
    "companyName": "<nazev_firmy>"
},
"contacts": {
    "email": "<email>",
    "phone": "<telefon>"
},
"comment": "<text_pozadavku>",
"interestedIn": [ "<p1>", "<p2>", " ..." ],
"leadCategory": "<string>",
"trackingCode": "<string>",
"url": "<string>"
}

```

## Function progress

- When above request is sent, new TIF is created (TIF = Temporary Import File) and the contents is saved in TIF; TIF type = **contactform-lead**
- Contact form that is not processed has TIF.status = 1
- The plugin calls auto-processing that makes the following steps:
  - Company is filled? => lead is "**Company lead**" (and the person is representative of the company), otherwise it's "**Person lead**"

## Company lead

- Check whether company is existing (company name matches some record in DB) // see: <https://help.atollon.com/books/backend-development/page/modcontact#bkmrk-%C2%A0>
- If company does not exist, create new contact (company)
- With company contact id, check whether contact has the **desired contact folder**.  
Desired contact folder is folder with FPATypeCategory = com.atollon.leads.fpatypecategory.lead OR optional **lead\_category** (represents FPATypeCategory of the lead folder type). If the lead folder does not exist, it is created.
- Check whether person is existing (based on e-mail, mobile / if these are not filled-in, the person is created anyway). E-mail and mobile is stored with person contact
- Comment is created as message and stored with lead folder ID // Subject: "New lead " & {{ firstName }} {{ lastName }}, {{ companyName }}
- Interested in - represents contact group. Find the contact group with the same name. If the contact group does not exist, create it in "**Interested in**" category. Assign the contact to the group
- Create marketing trigger (baseType = goal-achieved, url = trackingCode, description = "Contact filled-in form")

## Person lead

- The main difference from company lead is that the lead ID is searched / created for the person contact (because company contact is missing)

After the request is imported successfully, change the TIF status to "2".

## New Issue

```
https://<HOST>/svc/contact_form/v1/issue?instance=<instance_name>
```

// consider removing instance\_name

POST data in JSON format:

```
{
  "person": {
    "firstName": "<jmeno>",
    "lastName": "<prijmeni>"
  },
  "company": {
    "companyName": "<nazev_firmy>",
    "companyRegNo": "<registration_number>"
  },
  "contacts": {
    "email": "<email>",
    "phone": "<telefon>"
  },
  "comment": "<text_pozadavku>",
  "interestedIn": [ "<p1>", "<p2>", "..."],
  "leadCategory": "<string>",
  "trackingCode": "<string>",
  "url": "<string>",
  "requestType": "<New_Issue_Template>",
  "formFields": {
    "formField1": "<value1>",
    "formField2": "<value2>",
    "...": "..."
  }
}
```

## Function progress

The same as above for New Lead, except instead of New Message with new comment, we create **New Issue**.

- field requestType is identification of issue template based on Issue Template Name
- formFields values are saved into newly created issue formValues
- issue solver / user is identify based on issue template setting:

The screenshot shows a 'New Issue' form window with the following fields and controls:

- Název šablony: [Text input field]
- Předmět žádosti: [Text input field]
- Popis: [Text input field]
- Opakovaný termín: ☐
- Typ práce: [Dropdown menu]
- Stav: [Dropdown menu, value: Initial]
- Připomínka: [Dropdown menu, value: Bez upozornění] [Text input field, value: 0]
- Priorita: [Dropdown menu, value: Normální]
- Zaslat upozornění: ☐
- Odhadované hod...: [Text input field, value: 0]
- Zdroj řešitele: [Dropdown menu, value: Vybraný uživatel] (highlighted with an orange box)
- Uživatel: [Text input field]

Buttons at the bottom right: Zrušit, OK

Revision #21

Created 29 July 2021 13:11:05 by Martina Pivonkova

Updated 26 January 2025 07:10:51 by Jan Safka