

# ApiForm instruction - VTTI specific

BzCtrl. provides the ability to call predefined Forms through API communication using the **ApiForms** feature. This allows external systems to create and link Form instances directly from within their own operational processes.

The main value of ApiForms is that a user can launch a specific Form directly from the source system, while the created Form instance remains permanently accessible through a stored URL within that source system.

## How ApiForms Work

Each ApiForm provides a unique API endpoint that allows an external system to create a new Form instance in BzCtrl.

The request includes:

- Predefined Form reference (via API Endpoint)
- Tags and identifiers (for example an Order Number, Shipment Number, or Case ID).

Once the Form instance is created, BzCtrl. returns a response containing the direct URL to the Form instance. This URL can then be stored in the source system, allowing users to always reopen the corresponding Form directly from their operational process.

## User actions

**Step 1** - From the Forms Page the User can create an ApiForm via the Action Button for a specific Form;

Forms

Community database By Category

Procedure Control e-Learning Brainpick

MULTI-SELECT

| Name                                       | Description  | Contact Person | Create Date | Form Type ↑ | Domain | Action |
|--|--|----------------|-------------|-------------|--------|--------|
| Competence and professional qualifications | Purpose of this control is to validate the procedures in place to ensure that the professional standard of the company and its competence is of a sufficient standard.   |                | 01-12-2022  | Control     | iPear  |        |
| Customer screening                         | Purpose of this control is to validate if procedures are in place to ensure that the appropriate records are kept.   |                | 01-12-2022  | Control     | iPear  |        |
| Internal audit                             | Purpose of this control is to validate that a proper internal audit program is in place to monitor the customs processes and ensure follow up when actions for improvement are required.                                     |                | 28-05-2026  | Control     | iPear  |        |
| Incident review                            | The objective of this Control is to periodically check the incidents recorded to check on a macro level if there are recurring incidents that call for specific measures or to review any incident in detail                 |                | 21-07-2022  | Control     | iPear  |        |
| Information security                       | Procedures as regards back-up, recovery and fall-back and archival options (Subsection 3.6 from SAQ), protection of computer systems (Subsection 3.7 from SAQrev6) and documentation security (Subsection 3.8 from SAQrev6). |                | 01-12-2022  | Control     | iPear  |        |
| Proven financial solvency                  | The purpose of this control is to validate that the company continues to meet the criteria around financial solvency. (Article 39 (c) UCC, Article 26 UCC IA, AEO Guidelines Part 2 Section III)                             |                | 15-07-2022  | Control     | iPear  |        |

Create ApiForm

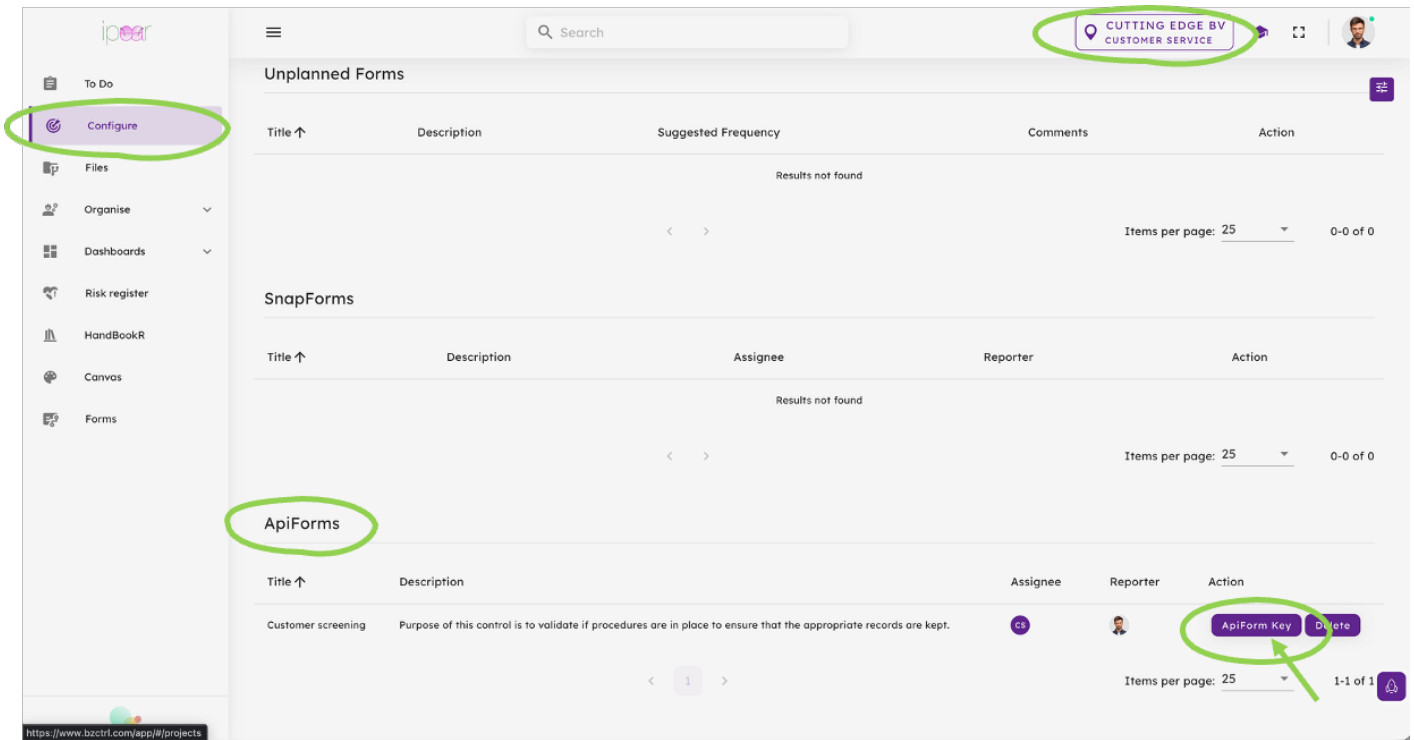
**Step 2** - Select the Workspace where the Form Instances should be created and stored;

Create ApiForm

Suppliers

- Gotlieb group
- DMR Trading
- Fashion Profile
- Freight Forward
- Cutting Edge BV
  - Customer Service
  - Finance
  - Logistics
  - Compliance

**Step 3** - From the Configure page the Form is now available as ApiForm. Please make sure the correct Workspace is selected (right top). From the ApiForms Table select the ApiForm Key Button to get the details to share with your developers.



# Input to Developers External system

## Step 1 - Authentication

Before an ApiForm can be called, the external application must authenticate with BzCtrl. Authentication is performed by sending a **POST** request to the appropriate authentication endpoint using credentials provided for API communication.

## Endpoints

### Acceptance Environment (ACC)

<https://acc.bzctrl.com/bzctrl-core-api/oauth/token>

### Production Environment (PROD)

<https://bzctrl.com/bzctrl-core-api/oauth/token>

## Request Method

POST

## Request Body (x-www-form-urlencoded)

```
grant_type=password
username=cms@api.com
```

```
password=Nologicincustoms#24
```

## Sample Response

```
{
  "access_token": "xxxxxxxxxx",
  "refresh_token": "xxxxxxxxxx",
  "token_type": "Bearer",
  "expires_in": 3600,
  "scope": "*",
  "jti": "xxxxxxxxxx"
}
```

## Result

Upon successful authentication, BzCtrl. returns an **access token** that must be included in subsequent API requests. The token serves as proof of authentication and authorizes communication with the BzCtrl. API. The token remains valid for the period specified in the expires\_in field.

## Step 2 - Api Form Key

Here you see the API Endpoint url to share with your developers, which use will result in the creation of a Form Instance of the respective Form on your selected Workspace. In the Body of the message Tags can be included, which is usually used for adding Order ID's to be recorded in the Form Instance.

### ApiForm Key - Acceptance Environment (ACC)

<https://acc.bzctrl.com/bzctrl-core-api/api/v1/task/add-exposed-task/eyJwcm9qZWN0SWQiOjUwMywiY29tcGFueUlkljozMzEslmNvbnRyb2xjZCI6MTQxNSwiZW5jb2RlVXJsljoiZDMzOGExNmYtZGZlMS00OTQzLTliMzUtMzQ4YmYzMzc1ZTE1Iiwia3JlYXRvcklkljo2NjMslndvcmtTcGFjZU5hbWUiOiJkdDdXN0b21zliwicmVwb3J0ZXJvc2VySWQiOjY2MywicmVwb3J0ZXJvc2VyTmFtZSI6IlJlbXkgR2VlcnRzIn0=>

### ApiForm Key - Production Environment (PROD)


<https://www.bzctrl.com/bzctrl-core-api/api/v1/task/add-exposed-task/eyJwcm9qZWN0SWQiOjI4MywiY29tcGFueUlkljoxOTksImNvbnRyb2xjZCI6MTQ1NSwiZW5jb2RlVXJsljoiNWJhM2QxMGYtNTgzMy00N2VjLWE5NzktMDk5NTM5MTNjNTM3Iiwia3JlYXRvcklkljoxMywi>

[d29ya1NwYWNITmFtZSI6IkN1c3RvbXMiLCJyZXBvcnRlcVzZXJjZCI6MTMsInJlcG9ydGVyVXNlck5hb](#)


[WUiOjSZW15IEdlZXJ0cyJ9](#)

 API Endpoint

```
https://www.bzctrl.com/bzctrl-core-api/api/v1/task/add-exposed-task/eyJwcm9qZWNO0SWQI0jQ2NCwiY29tcGFueUlkiJoxMDEsImNvbnRyb2xJZCI6MTQ1NCwiZW5jb2RlVXJsIjoIbmRkODVhODgtZGY5Ny00NDkzLWE2OWMtZjJmMmJjYjAzZDY1IiwiaWY3JiYXRvcklkIjo0MjEsImFzc2lnbmVlVXNlckdyb3VwSWQI0jEzMWid29ya1NwYWNIImFtZSI6Ikd1c3RvbWVvIFNlcnZpY2UiLCJyZXBvcnRlclVzZXJJZCI6NDIxLClJyZXBvcnRlclVzZXJOYW1IiwoiSmFtZXMGmQmlzaG9wIn0=
```

 Copy URL

POST request to update tags for a specific task

 Request Body Example

 You can update the tags value with your own tags as needed.

## JSON Request

 Copy

```
{
  "tags": [
    "urgent",
    "bug-fix",
    "frontend"
  ]
}
```

 Sample Response

## JSON Response

 Copy

```
{
  "title": "Fix navigation bug",
  "taskType": "CONTROL",
  "description": "Resolve issue with mobile navigation menu",
  "taskKey": "FED-98",
  "status": "TODO",
  "priority": null,
  "dueDate": null,
  "reminderDate": null,
  "completionDate": null,
  "taskLink": "https://example.com/#/todo/fed-98",
  "priorityLevel": 0,
  "assigneeUserGroupId": null,
  "creatorUserId": 1,
  "deleted": false,
  "isPrivate": false,
  "showResult": false,
  "tags": [
    "urgent",
    "bug-fix",
    "frontend"
  ],
  "assignee": "John Doe",
  "reporter": "Jane Smith",
  "taskAttributes": [],
  "watchers": [],
  "todoList": [],
  "categoryList": [],
  "subCategoryList": [],
  "company": "TechCorp Inc.",
  "project": "Website Redesign"
}
```

 Copy

After creation the Response message includes a Tasklink that can be stored to allow the user access to the Form Instance from the External system at any stage.

---

Revision #1

Created 8 June 2026 12:00:40 by Remy Sway

Updated 8 June 2026 12:00:40 by Remy Sway