P4ALL CROWDFUNDING REST API

This document describes the REST API and resources provided by Crowdfunding System. The REST APIs are for developers who want to integrate Crowdfunding into their application and for administrators who want to script interactions with the Crowdfunding server.

Crowdfunding's REST APIs provide access to resources (data entities) via URI paths. To use a REST API, your application will make an HTTP request and parse the response. The response format is JSON. Your methods will be the standard HTTP methods like GET, PUT, POST and DELETE.

Because the REST API is based on open standards, you can use any web development language to access the API.

# Resources

The fundamental concept in any RESTful API is the *resource*. A resource is an object with a type, associated data, relationships to other resources, and a set of methods that operate on it. It is similar to an object instance in an object-oriented programming language, with the important difference that only a few standard methods are defined for the resource (corresponding to the standard HTTP GET, POST, PUT and DELETE methods), while an object instance typically has many methods.

The P4All crowdfunding REST API exposes 4 types of resources: *projects*, *proposals*, *pledges* and *offers*. Projects are products or services in need of funding. Pledges are an amount of money that backers give to support the development of a project. A proposal is a suggestion for a new service or product. Offer is an amount of money that a developer wants for implementing a proposal.

## Project related API Calls

Operations pertaining to projects.

### POST /projects

Create a project

#### Parameters

**Body parameters**

|  |  |
| --- | --- |
| **Name** | **Description** |
| **project \*** | a project as a json object |

#### Consumes

This API call consumes the following media types via the Content-Type request header:

* application/json

#### Return type

ResponseEntity

#### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

* application/json

#### Responses

200 OK ResponseEntity

201 Created

401 Unauthorized

403 Forbidden

404 Not Found

### DELETE /projects/{id}

Delete a project

#### Parameters

**Path parameters**

|  |  |
| --- | --- |
| **Name** | **Description** |
| **id\*** | Integer (int32)  id of the project  Required |

#### Return type

ResponseEntity

#### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

* application/json

#### Responses

200 OK ResponseEntity

204 No Content

401 Unauthorized

403 Forbidden

### GET /projects

Retrieve a list of available projects

#### Parameters

**Query parameters**

|  |  |
| --- | --- |
| **Name** | **Description** |
| **page** | Object  Results page you want to retrieve (0..N) |
| **size** | Object  Number of records per page. |
| **sort** | array[String]  Sorting criteria in the format: property(,asc|desc). Default sort order is ascending. Multiple sort criteria are supported. |

#### Return type

Page

#### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

* application/json

#### Responses

200 Successfully retrieved list Page

401 You are not authorized to view the resource

403 Accessing the resource you were trying to reach is forbidden

404 The resource you were trying to reach is not found

### GET /projects/{id}

Retrieve a project by ID

#### Parameters

**Path parameters**

|  |  |
| --- | --- |
| **Name** | **Description** |
| **id\*** | Integer (int32)  id of the project  Required |

#### Return type

Project

#### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

* application/json

#### Responses

200 OK Project

401 Unauthorized

403 Forbidden

404 Not Found

### PUT /projects/{id}

Update a project

#### Parameters

**Path parameters**

|  |  |
| --- | --- |
| **Name** | **Description** |
| **id\*** | Integer (int32)  id of the project  Required |

**Body parameters**

|  |  |
| --- | --- |
| **Name** | **Description** |
| **project \*** | a project as a json object |

#### Consumes

This API call consumes the following media types via the Content-Type request header:

* application/json

#### Return type

ResponseEntity

#### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

* application/json

#### Responses

200 OK ResponseEntity

201 Created

401 Unauthorized

403 Forbidden

404 Not Found

## Proposal related API Calls

Operations pertaining to proposals.

### POST /proposals

Create a proposal (createUsingPOST3)

#### Parameters

Body parameters

|  |  |
| --- | --- |
| **Name** | **Description** |
| **proposal \*** | a proposal as a json object |

#### Consumes

This API call consumes the following media types via the Content-Type request header:

* application/json

#### Return type

ResponseEntity

#### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

* application/json

#### Responses

200 OK ResponseEntity

201 Created

401 Unauthorized

403 Forbidden

404 Not Found

### DELETE /proposals/{id}

Delete a proposal (deleteUsingDELETE3)

#### Parameters

**Path parameters**

|  |  |
| --- | --- |
| **Name** | **Description** |
| **id\*** | Integer (int32)  id of the proposal  Required |

#### Return type

ResponseEntity

#### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

* application/json

#### Responses

200 OK ResponseEntity

204 No Content

401 Unauthorized

403 Forbidden

### GET /proposals

Retrieve a list of available proposals (listUsingGET3)

#### Parameters

**Query parameters**

|  |  |
| --- | --- |
| **Name** | **Description** |
| **page** | Object  Results page you want to retrieve (0..N) |
| **size** | Object  Number of records per page. |
| **sort** | array[String]  Sorting criteria in the format: property(,asc|desc). Default sort order is ascending. Multiple sort criteria are supported. |

#### Return type

Page

#### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

* application/json

#### Responses

200 Successfully retrieved list Page

401 You are not authorized to view the resource

403 Accessing the resource you were trying to reach is forbidden

404 The resource you were trying to reach is not found

### GET /proposals/{id}

Retrieve a proposal by ID (showUsingGET3)

#### Parameters

Path parameters

|  |  |
| --- | --- |
| **Name** | **Description** |
| **id\*** | Integer (int32)  id of the proposal  Required |

#### Return type

Proposal

#### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

* application/json

#### Responses

200 OK Proposal

401 Unauthorized

403 Forbidden

404 Not Found

### PUT /proposals/{id}

Update a proposal (updateUsingPUT3)

#### Parameters

Path parameters

|  |  |
| --- | --- |
| **Name** | **Description** |
| **id\*** | Integer (int32)  id of the proposal  Required |

Body parameters

|  |  |
| --- | --- |
| **Name** | **Description** |
| **proposal \*** | a proposal as a json object |

#### Consumes

This API call consumes the following media types via the Content-Type request header:

* application/json

#### Return type

ResponseEntity

#### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

* application/json

#### Responses

200 OK ResponseEntity

201 Created

401 Unauthorized

403 Forbidden

404 Not Found

## Pledge related API Calls

Operations pertaining to project pledges.

### POST /pledges

Create a pledge

#### Parameters

Body parameters

|  |  |
| --- | --- |
| **Name** | **Description** |
| **pledge \*** | a pledge as a json object |

#### Consumes

This API call consumes the following media types via the Content-Type request header:

* application/json

#### Return type

ResponseEntity

#### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

* application/json

#### Responses

200 OK ResponseEntity

201 Created

401 Unauthorized

403 Forbidden

404 Not Found

### DELETE /pledges/{id}

Delete a pledge

#### Parameters

Path parameters

|  |  |
| --- | --- |
| **Name** | **Description** |
| **id\*** | Integer (int32)  id of the pledge  Required |

#### Return type

ResponseEntity

#### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

* application/json

#### Responses

200 OK ResponseEntity

204 No Content

401 Unauthorized

403 Forbidden

### GET /pledges

Retrieve a list of available pledges

#### Parameters

Query parameters

|  |  |
| --- | --- |
| **Name** | **Description** |
| **page** | Object  Results page you want to retrieve (0..N) |
| **size** | Object  Number of records per page. |
| **sort** | array[String]  Sorting criteria in the format: property(,asc|desc). Default sort order is ascending. Multiple sort criteria are supported. |

#### Return type

Page

#### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

* application/json

#### Responses

200 Successfully retrieved list Page

401 You are not authorized to view the resource

403 Accessing the resource you were trying to reach is forbidden

404 The resource you were trying to reach is not found

### GET /pledges/{id}

Retrieve a pledge by ID

#### Parameters

Path parameters

|  |  |
| --- | --- |
| **Name** | **Description** |
| **id\*** | Integer (int32)  id of the pledge  Required |

#### Return type

Pledge

#### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

* application/json

#### Responses

200 OK Pledge

401 Unauthorized

403 Forbidden

404 Not Found

### PUT /pledges/{id}

Update a pledge

#### Parameters

Path parameters

|  |  |
| --- | --- |
| **Name** | **Description** |
| **id\*** | Integer (int32)  id of the pledge  Required |

Body parameters

|  |  |
| --- | --- |
| **Name** | **Description** |
| **pledge \*** | a pledge as a json object |

#### Consumes

This API call consumes the following media types via the Content-Type request header:

* application/json

#### Return type

ResponseEntity

#### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

* application/json

#### Responses

200 OK ResponseEntity

201 Created

401 Unauthorized

403 Forbidden

404 Not Found

## Offer related API Calls

Operations pertaining to proposal offers.

### POST /offers

Create an offer

#### Parameters

Body parameters

|  |  |
| --- | --- |
| **Name** | **Description** |
| **offer \*** | an offer as a json object |

#### Consumes

This API call consumes the following media types via the Content-Type request header:

* application/json

#### Return type

ResponseEntity

#### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

* application/json

#### Responses

200 OK ResponseEntity

201 Created

401 Unauthorized

403 Forbidden

404 Not Found

### DELETE /offers/{id}

Delete an offer

#### Parameters

Path parameters

|  |  |
| --- | --- |
| **Name** | **Description** |
| **id\*** | Integer (int32)  id of the offer  Required |

#### Return type

ResponseEntity

#### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

* application/json

#### Responses

200 OK ResponseEntity

204 No Content

401 Unauthorized

403 Forbidden

### GET /offers

Retrieve a list of available offers

#### Parameters

Query parameters

|  |  |
| --- | --- |
| **Name** | **Description** |
| **page** | Object  Results page you want to retrieve (0..N) |
| **size** | Object  Number of records per page. |
| **sort** | array[String]  Sorting criteria in the format: property(,asc|desc). Default sort order is ascending. Multiple sort criteria are supported. |

#### Return type

Page

#### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

* application/json

#### Responses

200 Successfully retrieved list Page

401 You are not authorized to view the resource

403 Accessing the resource you were trying to reach is forbidden

404 The resource you were trying to reach is not found

### GET /offers/{id}

Retrieve an offer by ID

#### Parameters

Path parameters

|  |  |
| --- | --- |
| **Name** | **Description** |
| **id\*** | Integer (int32)  id of the offer  Required |

#### Return type

Offer

#### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

* application/json

#### Responses

200 OK Offer

401 Unauthorized

403 Forbidden

404 Not Found

### PUT /offers/{id}

Update an offer

#### Parameters

Path parameters

|  |  |
| --- | --- |
| **Name** | **Description** |
| **id\*** | Integer (int32)  id of the offer  Required |

Body parameters

|  |  |
| --- | --- |
| **Name** | **Description** |
| **offer \*** | a offer as a json object |

#### Consumes

This API call consumes the following media types via the Content-Type request header:

* application/json

#### Return type

ResponseEntity

#### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

* application/json

#### Responses

200 OK ResponseEntity

201 Created

401 Unauthorized

403 Forbidden

404 Not Found

# Models

## Project

A service or product that needs fundraising.

|  |  |
| --- | --- |
| **amount\*** | string  Fundraising goal. The amount of money this project must raise. |
| **content** | string  The project's long description. |
| **ctime** | string($date-time)  Creation date. |
| **currency** | string  The used currency. |
| **description** | string  The project's short description |
| **etime\*** | string($date-time)  Project expiration date. If the project doesn’t meet all the requirements until this date, it must be canceled. |
| **expired** | boolean  True if project is expired. |
| **id** | integer($int32)  The database generated project Id. |
| **name\*** | string  The project's name. |
| **user\*** | string  Id of the project's owner. |

Example Value

{

"id": 0,   
"name": "string",   
"description": "string",   
"content": "string",   
"amount": "string",   
"currency": "string",   
"ctime": "2018-03-02T15:33:36.984Z",   
"etime": "2018-03-02T15:33:36.984Z",   
"expired": false,   
"user": "string"

}

## Proposal

A proposal for a service or product.

|  |  |
| --- | --- |
| **content** | string  The proposal's long description. |
| **ctime** | string($date-time)  Creation date. |
| **description** | string  The proposal's short description |
| **etime\*** | string($date-time)  Proposal expiration date. If the proposal doesn’t meet all the requirements until this date, it must be canceled. |
| **expired** | boolean  True if proposal is expired. |
| **id** | integer($int32)  The database generated proposal Id. |
| **name\*** | string  The proposal's name. |
| **user\*** | string  Id of the proposal's owner. |

Example Value

{

"id": 0,   
"name": "string",   
"description": "string",   
"content": "string",   
"ctime": "2018-03-02T15:33:36.984Z",   
"etime": "2018-03-02T15:33:36.984Z",   
"expired": false,   
"user": "string"

}

## Offer

An offer for implementing a proposal.

|  |  |
| --- | --- |
| **amount\*** | string  The offer amount. |
| **currency** | string  The used currency |
| **id** | integer($int32)  The database generated Offer Id |
| **message** | string  An optional message body for the proposal's owner. |
| **proposal\*** | integer($int32)  Proposal |
| **title** | string  An optional message title for the proposal's owner. |
| **user\*** | string  Id of the user that made this offer. |

Example Value

{

"id": 0,  
"amount": "string",  
"currency": "string",  
"title": "string",  
"message": "string",  
"user": "string",  
"proposal": 0,

}

## Pledge

A pledge for a project.

|  |  |
| --- | --- |
| **amount\*** | string  The amount of money the user pledged. |
| **currency** | string  The used currency. |
| **id** | integer($int32)  The database generated Pledge Id. |
| **message** | string  An optional message body for the developer. |
| **project\*** | integer($int32)  The project this user backed. |
| **title** | string  An optional message title for the developer. |
| **user\*** | string  Id of the user that made this pledge. |
| **paymentID** | string  Paypal payment id |
| **payerID** | string  Paypal payer id |

Example Value

{

"id": 0,  
"amount": "string",  
"currency": "string",  
"title": "string",  
"message": "string",  
"payerID": "string",  
"paymentID": "string",  
"project": 0,  
"user": "string"

}

## Page

|  |  |
| --- | --- |
| **number** | integer  number of page |
| **last** | boolean  True if this is the last page |
| **numberOfElements** | integer size of content array  Creation date. |
| **size** | integer  Number of records per page. |
| **totalPages** | integer  Number of available pages |
| **sort** | array  Sorting criteria in the format: property(,asc|desc). Default sort order is ascending. Multiple sort criteria are supported. |
| **content** | array  An array of objects. |
| **first** | boolean  True if this is the first page |
| **totalElements** | integer  Number of available records. |

Example Value

{

"number" : 0,  
"last" : true,  
"numberOfElements" : 6,  
"size" : 1,  
"totalPages" : 5,  
"sort" : { },  
"content" : [ "{}" ],  
"first" : true,  
"totalElements" : 5

}

## ResponseEntity

|  |  |
| --- | --- |
| **body** | object |
| **statusCode** | string |
| **statusCodeValue** | integer |

Example Value

{

"statusCodeValue" : 0,  
"body" : "{}",  
"statusCode" : "100"

}