PDS API Documentation V1.0

As part of the Digital Transition Partnership of Urban Agenda for the EU the partners in the Action 8 have defined a standard (PDS) which specifies the data structure of participatory data in the urban planning process with the aim to support citizen centric planning. As a reference implementation an API was developed in Hamburg for its open source digital participation tool DIPAS which concretise and put the data structure as specified in the PDS into use. Contributions and other project related information stored in the database of DIPAS can be retrieved through this API by other applications or systems for further use.

The base URI for the API in DIPAS is beteiligung.hamburg/dipas/drupal/dipas-pds

Source code of DIPAS is available at https://bitbucket.org/geowerkstatt-hamburg/dipas/

Following end points are specified:

Method	Description	Response format
GET /projects	A list of all participatory projects including detailed information	GeoJSON
GET /projects/[proj_ID]	Detailed information of the project with the ID *proj_ID*	GeoJSON
GET /projects/[proj_ID] /contributions	All contributions from the project with the ID *proj_ID*	GeoJSON
GET /projects/[proj_ID]/contributions/ [contr_ID]	Accesses within a project the contribution with the ID *contr_ID*	GeoJSON
GET /projects/[proj_ID]/contributions/ [contr_ID] /comments	Accesses all comments on the contribution with the ID *contr_ID*	GeoJSON
GET /projects/[proj_ID]/commentedcontributions	All contributions including all their comments from the project with the ID *proj_ID*	GeoJSON
GET /projects/[proj_ID]/commentedcontributions/[contr_ID]	Accesses within a project the contribution with the ID *contr_ID* and all its comments	GeoJSON
GET projects/[proj_ID]/conception_comments	Accesses all comments on the design concepts in the project with the ID *proj_ID*	JSON

1. Projects

Projects are represented with *ParticipatoryProject* Features. Following properties can be contained in a Feature. As of now only the coloured fields are provided by the PDS-API.

ParticipatoryProject

Name	Type	Definition
id	String	An automatically generated identifier for the project
nameShort	String [0..1]	Short name or abbreviation of the project
nameFull	String	The full, formal name of the project
description	String	Description of the aims, content and target groups of the participatory project
dateStart	String (DateTime)	Starting date of the project, compliant to the ISO 8601 (YYYY-MM-DDThh:mm:ssTZD) ISO 8601, the International Standard for the representation of dates and times https://www.w3.org/TR/NOTE-datetime
dateEnd	String (DateTime) [0..1]	Ending date of the project, compliant to the ISO 8601 (YYYY-MM-DDThh:mm:ssTZD)
website	String (URL) [0..1]	URL to the web site of the project
processingStep	String [0..1]	The current processing step of the main project in which the participatory project takes place. The steps can be predefined and the values may vary depending on the country, so they should be defined uniformly in a district/city/region/or even country
areaSize	Number [0..1]	Size of the participatory project area in km^2
locationDescription	String [0..1]	Qualitative description of the location, e.g. demographics, land use
owner	String	Name of the person, unit or organization responsible for running the participatory project (project manager)
publisher	String	Name of person, unit or organization responsible for providing and publishing the data
standardCategories	Array	Standard categories in a list defined by the project owner, e.g. mobility, housing, greenery etc. For each contribution, a value from this list would be chosen to classify it.
standardSubcategories	Array of strings [0..1]	Standard subcategories for a participatory project defined by the project owner. For example the category mobility can have the subcategories pedestrian traffic, vehicle traffic, bicycle traffic etc.
projectContributionType	Array of strings [0..1]	A list of allowed values for the types of contribution in the project. This should be provided by the person or organisation responsible for the participatory project.

referenceSystem	String (EPSGCode)	The EPSG code of the original geographical data
hasParticipatoryText	Array of strings [0..*]	A list of contribution IDs which belong to the project
dipasCategoriesCluster	Array [0..*]	DIPAS specific property. Contains the clusters for each category from the NLP tool.
dipasPhase	String	Custom attribute of DIPAS. Possible values: Phase1, Phase2, PhaseMix

1.1. List of all projects

Request to /projects

Example GET Request "https://beteiligung.hamburg/dipas/drupal/dipas-pds /projects"

Response: The endpoint returns a list of all participatory projects as GeoJSON FeatureCollection with projects as Features. Following properties can be contained in a project Feature.

Example Response:

```
{
  "type": "FeatureCollection",
  "features": [{
    "type": "Feature",
    "id": "4551e539-5076-4c18-82d5-c390a4c5a363",
    "properties": {
      "nameFull": "Online-Beteiligung Grasbrook",
      "dateStart": "2019-02-20T18:17:00+01",
      "website": "https://geoportal-hamburg.de/beteiligung_grasbrook",
      "description": "Der neue Stadtteil besetzt für die kommenden Jahrzehnte...",
      "owner ": "FHH",
      "publisher ": "LGV",
      "standardCategories": ["Mobilität", "Wohnen und Nachbarschaft", "Städtebau", "Öffentlicher Raum"],
      "projectContributionType": ["Idee", "Frage", "Kritik"],
      "referencSystem": "4326",
      "hasParticipatoryText": ["contribution-id001", "contribution-id002"]
    },
    "geometry": {
      "type": "Polygon",
      "coordinates": [
        [
          [9.995077, 53.552037],
          [9.996708, 53.549487],
          [9.996965, 53.547397],
          [10.000828, 53.547499],
          [10.004262, 53.546326],
          [10.006322, 53.549793],
          [10.005636, 53.553057],
          [10.000657, 53.55581],
          [9.995077, 53.552037]
        ]
      ]
    }
  }, {
    "type": "Feature",
    "id": "a5c2d949-bc06-452b-9f78-5c2a82b5b4ce",
 "properties": {
      "nameFull": "Online-Beteiligung Grasbrook",
      "dateStart": "2019-02-20T18:17:00+01",
      "website": "https://geoportal-hamburg.de/beteiligung_grasbrook",
      "description": "Der neue Stadtteil besetzt für die kommenden Jahrzehnte...",
      "owner ": "FHH",
```

```
        "publisher ": "LGV",
        "standardCategories": ["Mobilität", "Wohnen und Nachbarschaft", "Städtebau", "Öffentlicher Raum"],
        "projectContributionType": ["Idee", "Frage", "Kritik"],
        "referencSystem": "4326",
        "hasParticipatoryText": ["contribution-id003", "contribution-id004"]
      },
      "geometry": {
        "type": "Polygon",
        "coordinates": [
          [
            [10.021431, 53.549997],
            [10.023405, 53.547397],
            [10.027955, 53.548722],
            [10.025981, 53.551374],
            [10.021431, 53.549997]
          ]
        ]
      }
    }]
}
```


1.2. Detailed information of a project

Request to /projects/[proj_ID]

Example GET Request "https://beteiligung.hamburg/dipas/drupal/dipas-pds /projects/id_1"

Response: The endpoint returns the detailed information of the participatory project with the ID *proj_ID* as GeoJSON Feature.

2. Contributions and comments

Contributions are represented with *Contribution* Features. In DIPAS different endpoints have been defined for contributions with or without comments. Following properties can be contained in a *Contribution* Feature. As of now only the coloured fields are provided by the PDS-API.

Contribution

Name	Type	Definition
id	String	Automatically generated identifier
dateCreated	String (DateTime)	Automatically generated date when contribution was created
link	String (URL) [0..1]	URL to a specific contribution, if existing
title	String [0..1]	Title of contribution
latitude	Number [0..1]	Latitude of the point where the stated issue in the contribution is located (WGS 84). When the location is specified with a line or polygon, this field should be left blank.
longitude	Number [0..1]	Longitude of the point where the stated issue in the contribution is located (WGS 84). When the location is specified with a line or polygon, this field should be left blank.
attachment	String (URL) [0..*]	URL to the attachments. For example photos and documents can be included as attachments.
contributionType	String [0..1]	A value from the list of the predefined types of contribution, e.g. suggestion, opinion, criticism
contributionContent	String	Content of the contribution
status	StatusValue [0..1]	The current status of the contribution, if it was not yet worked on, categorized, in progress or already finished etc.
commentsNumber	Number (integer)	The total number of comments on a contribution
category	String	Chosen by the contributor, this is a value from the list of standard categories defined by the project owner.
subCategory	String [0..1]	Chosen by the contributor from a list of standard subcategories defined by the product owner.
votingPro	Number (integer) [0..1]	Citizens have the possibility to vote for or against a contribution. This feature is optional, the final results of the pro-voting will be stored here.
votingContra	Number (integer) [0..1]	Citizens have the possibility to vote for or against a contribution. This feature is optional, the final results of the contra-voting will be stored here.
keywordSuggested	Array of strings [0..1]	A typical function of AI text analyze is the identification of keywords. Upon input, a AI-based system can suggest keywords for the author to choose from, the chosen keywords will then be assigned to the contribution and used for clustering

keywordPicked	Array of strings [0..1]	From the keywords that the AI system suggested, some are accepted and some rejected by the user. The accepted keywords are stored here for further processing.
sentiment	SentimentType [0..1]	Here an example of a custom attribute: sentiment. Another typical function of AI text analysis is the identification of sentiment. The result can be represented in different ways. Whether classified into categories like positive, negative and neutral or represented on a certain scale, this can be further specified by the data analyst and project owner.
customAttribute	CustomAttributeType [0..*]	The participatory data may need more attributes than specified in this model. For this purpose the data model can be extended according to need.
belongToProject	String (project.id)	The project to which the contribution belongs.
commentedBy	Array of contribution objects [0..1]	A list of comments to the contribution. For structure of the comments refer to the following table
author	Author [0..1]	Author who wrote the contribution or comment
dipasLocated	Boolean	Specifies if a contribution is located

Comments are represented with *Comment* Features. Comments to a certain contribution can be returned through a dedicated endpoint as Featurecollection or be nested in *Contribution* Features. Following properties can be contained in a comment Feature. As of now only the coloured fields are provided by the PDS-API.

Comment

Name	**Type**	**Definition**
id	String	Automatically generated identifier
dateCreated	String (DateTime)	Automatically generated date when contribution was created
link	String (URL) [0..1]	URL to a specific contribution, if existing
title	String [0..1]	Title of contribution
latitude	Number [0..1]	Latitude of the point where the stated issue in the contribution is located (WGS 84). When the location is specified with a line or polygon, this field should be left blank.
longitude	Number [0..1]	Longitude of the point where the stated issue in the contribution is located (WGS 84). When the location is specified with a line or polygon, this field should be left blank.
attachment	String (URL) [0..*]	URL to the attachments. For example photos and documents can be included as attachments.
contributionType	ContributionTypes [0..1]	List of the predefined types of contribution, e.g. suggestion, opinion, criticism
commentContent	String	Content of the comment

votingPro	Number (integer) [0..1]	Citizens have the possibility to vote for or against a contribution. This feature is optional, the final results of the pro-voting will be stored here.
votingContra	Number (integer) [0..1]	Citizens have the possibility to vote for or against a contribution. This feature is optional, the final results of the contra-voting will be stored here.
commentOnContribution	String (contribution.id) [0..1]	ID of the contribution which is commented
commentOnComment	String (contribution.id) [0..1]	ID of the comment which is commented
commentedBy	Array of comment objects [0..1]	List of the comments which comment on this object
customAttribute	CustomAttributeType [0..*]	The participatory data may need more attributes than specified in this model. For this purpose the data model can be extended according to need.

2.1. All Contributions from a project without comments

Request to /projects/[proj_ID] /contributions

Example GET Request "https://beteiligung.hamburg/dipas/drupal/dipas-pds/projects/id_1/contributions"

Response: The endpoint returns all contributions from a certain project as FeatureCollection. The comments from the project are not included in the response.

Example GET Request "https://beteiligung.hamburg/dipas/drupal/dipas-pds/projects/c8aefcd7-f6c3-4371-9914-1bb5ecbc4184/commentedcontributions"

2.2. A specific contribution without comments

Request to /projects/[proj_ID]/contributions/[contr_ID]

Response: The endpoint returns the contribution with the ID *contr_ID* as Feature. The comments on this contribution are not included in the response.

Example GET Request "https://beteiligung.hamburg/dipas/drupal/dipas-pds/projects/id_1/contributions/contr_1"

2.3. Comments on a specific contribution

Request to /projects/[proj_ID]/contributions/[contr_ID] /comments

Response: The endpoint returns all the comments on a contribution with the ID *contr_ID* as FeatureCollection.

Example GET Request "https://beteiligung.hamburg/dipas/drupal/dipas-pds/projects/id_1/contributions/contr_1/comments"

2.4. Commented Contributions

Request to /projects/[proj_ID]/commentedcontributions

Response: The endpoint returns all the contributions in a project with the ID *proj_ID* as FeatureCollection. Comments are nested in the contributions.

Example GET Request **"https://beteiligung.hamburg/dipas/drupal/dipas-pds/projects/id_1/commentedcontributions"**

Example Response:


```
{
  "type": "FeatureCollection",
  "features": [{
    "type": "Feature",
    "id": "contr_1",
    "properties": {
      "dateCreated": "2019-02-20T18:17:00+01",
      "link": "https://geoportal-hamburg.de/beteiligung_grasbrook/node/277",
      "title": "Wasserwege im Mobilitätskonzept",
      "contributionType": "opinion",
      "contributionContent": "Wasserwege in das Mobilitätskonzept einbeziehen. Zum einen für die Logistik des Gewerbes und zum anderen für den ÖPNV. Ggf unter der Auflage das an den Anliegern nur elektrisch betriebene Boote für eine zeitlich begrenzte Dauer anlegen dürfen.. ",
      "commentsNumber": 1,
      "category": "Mobilität",
      "votingPro": 10,
      "votingContra": 5,
      "belongToProject": "id_1",
      "commentedBy": [{
        "id": "comment_id001",
        "dateCreated": "2019-02-20T18:17:00+01",
        "title": "Wasserwege im Mobilitätskonzept",
        "commentContent": "Content of the comment",
        "votingPro": 10,
        "votingContra": 5,
        "commentOnContribution": "contribution_id001"
      }]
    },
    "geometry": {
      "type": "Point",
      "coordinates": [9.994647, 53.55219]
    }
  }, {
    "type": "Feature",
```

```
    "id": "contr_2",
    "properties": {
      "dateCreated": "2019-02-20T18:17:30+01",
      "link": "https://geoportal-hamburg.de/beteiligung_grasbrook/node/257",
      "title": "Contribution Title",
      "contributionType": "suggestion",
      "contributionContent": "Contribution Content02",
      "commentsNumber": 0,
      "category": "Städtebau",
      "votingPro": 5,
      "votingContra": 3,
      "belongToProject": "id_1"
    },
    "geometry": {
      "type": "LineString",
      "coordinates": [
        [9.996897, 53.553159],
        [9.998141, 53.553936]
      ]
    }
  }]
}
```


2.5. Details of a contribution with its comments

Request to /projects/[proj_ID]/commentedcontributions/[contr_ID]

Response: The endpoint returns the contribution with the ID *contr_ID*. Comments are nested in the contribution.

Example GET Request "https://beteiligung.hamburg/dipas/drupal/dipas-pds/projects/id_1/commentedcontributions/contr_1"

Example Response:


```
{
  "type": "FeatureCollection",
  "features": [{
    "type": "Feature",
    "id": "contr_1",
    "properties": {
      "dateCreated": "2019-02-20T18:17:00+01",
      "link": "https://geoportal-hamburg.de/beteiligung_grasbrook/node/277",
      "title": "Wasserwege im Mobilitätskonzept",
      "contributionType": "opinion",
```

```
"contributionContent": "Wasserwege in das Mobilitätskonzept einbeziehen. Zum einen für die Logistik des Gewerbes und zum anderen für den ÖPNV. Ggf unter der Auflage das an den Anliegern nur elektrisch betriebene Boote für eine zeitlich begrenzte Dauer anlegen dürfen.. ",
"commentsNumber": 1,
"category": "Mobilität",
"votingPro": 10,
"votingContra": 5,
"belongToProject": "id_1",
"commentedBy": [{
  "id": "comment_id001",
  "dateCreated": "2019-02-20T18:17:00+01",
  "title": "Wasserwege im Mobilitätskonzept",
  "commentContent": "Content of the comment",
  "commentOnContribution": "contribution_id001"
}, {
  "id": "comment_id002",
  "dateCreated": "2020-02-20T18:17:00+01",
  "title": "Titel of the comment",
  "commentContent": "Content of the comment",
  "votingPro": 3,
  "votingContra": 2,
  "commentOnContribution": "contribution_id001"
}]
},
"geometry": {
  "type": "Point",
  "coordinates": [9.991777, 53.549156]
}
}]
}
```


2.6. Comments on the design concepts

Request to projects/[proj_ID]/conception_comments

Example GET Request "https://beteiligung.hamburg/dipas/drupal/dipas-pds/projects/id_1/contributions/conception_comments

Response: The endpoint returns a list of comments to a certain design concept in JSON format. Comments on comment are delivered in a nested structure.

Comments on design concepts are also represented with *Comment* Features. Comments to a certain design concept will be returned through a dedicated endpoint as Featurecollection.