ABDA API usage

Version History

Date	Version	Changes
10-20-2023	0.1.0	Initial document
11-06-2023	0.2.0	Updates authorization schema (Bearer format)
02-21-2024	0.3.0	Adds PZN search endpoint, fixed documentation error
02-22-2024	0.4.0	Adds /pzns/products endpoint
03-19-2024	0.5.0	Changes /pzns/products endpoint to include ATC output
05-15-2024	0.6.0	Adds potentially inadequate medicine (Priscus 2.0) endpoints

General Remarks

The ABDATA API has been provided by the Saarland University Clinical Pharmacy working group. The API is not intended for public use, but only for usage within the SafePolyMed project. This document is intended as a guide for using the API, it is, however, not a comprehensive manual or technical documentation of the API.

Access

The API is provided under the following URL: https://abdata.clinicalpharmacy.me/api.

Usage and Testing Info

The API is generally intended for usage with dedicated console utilities such as *curl* or the corresponding utilities in programming languages such as the *httr* or similar packages in the *R* programming language. There is no dedicated endpoint for testing access to the API yet. However, testing **GET** endpoints is possible in a browser, for instance https://abdata.clinicalpharmacy.me/api/limits should return


```
{
    "type": "https://tools.ietf.org/html/rfc7235#section-3.1",
    "title": "Unauthorized",
    "status": 401,
    "detail": "No login token provided.",
    "instance": "/limits"
}
```


Authorization

This API uses *Java Web Token (jwt)* for authentication. A *jwt* is provided to you upon login and must be provided when accessing all other routes. See GET /interactions/compounds as an example on how to provide the token.

Endpoints

The following is a list of endpoints for the API. All endpoints are only accessible **without a trailing slash!** All routes refer to https://abdata.clinicalpharmacy.me/api as the root URI. Currently, all access to all routes other than POST /login **require authentication.**

Group	Method	Route	Description
user	POST	/login	Log a user in. See POST /login for more information and example usage.
user	GET	/renew-token	Retrieve a new token. No parameters.
information	GET	/formulations	Retrieve a list of all formulations within the database. No parameters.
information	GET	/limits	Request limits of the server. No parameters.
information	GET	/interactions/description	Request description of the interaction table. No parameters.
interactions	GET	/interactions/compounds	Interaction endpoint for compound names input. See GET /interactions/compounds for more information and example usage.
interactions	POST	/interactions/compounds	Interaction endpoint for compound names input from json. See POST /interactions/compounds for more information and example usage.
interactions	GET	/interactions/pzns	Interaction endpoint for pzn input. See GET /interactions/pzns for more information and example usage.
interactions	POST	/interactions/pzns	Interaction endpoint for pzn input from json. See POST /interactions/pzns for more information and example usage.
priscus	GET	/priscus/compounds	Priscus 2.0 (potentially inadequate medicine for geriatric patients) endpoint for compound name input.
priscus	POST	/priscus/compounds	Priscus 2.0 (potentially inadequate medicine for geriatric patients) endpoint for compound name input from json.
priscus	GET	/priscus/pzns	Priscus 2.0 (potentially inadequate medicine for geriatric patients) endpoint for pzn input.
priscus	POST	/priscus/pzns	Priscus 2.0 (potentially inadequate medicine for geriatric patients) endpoint for pzn input from json.
atc	GET	/atcs/drugs	Drug endpoint for ATC input.

Group	Method	Route	Description
pzns	GET	/pzns/products	Drug products endpoint for PZN input.

Example Usage

POST /login

Input

Provide your credentials as a *json*. The *json* must be structured as follows:

```
{
    "credentials": {
        "username": "your_username",
        "password": "your_password"
    }
}
```

Example Usage

```
  curl -X POST "https://abdata.clinicalpharmacy.me/api/login"  \
    -H "accept: application/json" \
    -H "Content-Type: application/json" \
    -d '{"credentials":{"username":"username","password":"password"}}'
```

Output

The return value for a successful **POST** request has the following structure:

```
{
    "yourjwttoken"
}
```

GET /interactions/compounds

Input

Check for interactions based on compound names provided as query parameters.

Example Usage

```
curl -X GET "https://abdata.clinicalpharmacy.me/api/interactions/compounds?
compounds=verapamil,simvastatin" \
```

```
    -H "accept: application/json" \
    -H "Content-Type: application/json" \
    -H "Authorization: Bearer yourjwttoken"
```

Output

The return value for a successful **GET** request has the following structure:


```
{
  "interactions": [
    {
      "plausibility": "plausible mechanism",
      "relevance": "severe",
      "frequency": "not known",
      "credibility": "high",
      "direction": "unidirectional interaction",
      "left_compound": "Simvastatin",
      "right_compound": "Verapamil",
      "left_atc": "C10AA01",
      "right_atc": "C08DA01",
      "left_formulation": "FTA",
      "right_formulation": "FTA",
      "left_medication": "ZOCOR 10mg",
      "right_medication": "Isoptin 80mg",
      "left_dose": "10 mg",
      "right_dose": "74.06 mg"
    }
  ],
  "unknown_compounds": [],
  "timestamp": "2023-10-20 13:20:57",
  "api_version": "0.3.0",
  "compounds": [
    "verapamil",
    "simvastatin"
  ]
}
```


POST /interactions/compounds

Input

Check for interactions based on compound names provided as *json*. Drug lists must be provided matched to an *ID*:

```
{
    [
        {
            "id": "1",
```

```
            "compounds": ["verapamil","simvastatin"]
        },
        {
            "id": "2",
            "compounds": ["diltiazem","amiodarone","amlodipine","lovastatin"]
        }
    ]
}
```

Example Usage

```
curl -X POST "https://abdata.clinicalpharmacy.me/api/interactions/compounds" \
     -H "Authorization: Bearer yourjwttoken" \
     -H "Content-Type: application/json" \
     -d '[{"id":"1","compounds":["verapamil","simvastatin"]},
{"id":"2","compounds":["diltiazem","amiodarone","amlodipine","lovastatin"]}]'
```

Output

The return value for a successful **POST** request has the following structure:

```
{
  "results": [
    {
      "interactions": [
        {
          "plausibility": "plausible mechanism",
          "relevance": "severe",
          "frequency": "not known",
          "credibility": "high",
          "direction": "unidirectional interaction",
          "left_compound": "Simvastatin",
          "right_compound": "Verapamil",
          "left_atc": "C10AA01",
          "right_atc": "C08DA01",
          "left_formulation": "FTA",
          "right_formulation": "FTA",
          "left_medication": "ZOCOR 10mg",
          "right_medication": "Isoptin 80mg",
          "left_dose": "10 mg",
          "right_dose": "74.06 mg"
        }
      ],
      "unknown_compounds": [],
      "id": "1",
      "compounds": [
        "verapamil",
        "simvastatin"
```

```
      ]
    },
    {
      "interactions": [
        {
          "plausibility": "plausible mechanism",
          "relevance": "moderate",
          "frequency": "not known",
          "credibility": "weak",
          "direction": "undirected interaction",
          "left_compound": "Amiodarone",
          "right_compound": "Diltiazem",
          "left_atc": "C01BD01",
          "right_atc": "C08DB01",
          "left_formulation": "DFL",
          "right_formulation": "RET",
          "left_medication": "Cordarex 150mg/3ml Injektionslösung",
          "right_medication": "Dilzem 120mg retard",
          "left_dose": "141.98 mg",
          "right_dose": "110.3 mg"
        },
        {
          "plausibility": "plausible mechanism",
          "relevance": "severe",
          "frequency": "not known",
          "credibility": "high",
          "direction": "unidirectional interaction",
          "left_compound": "Lovastatin",
          "right_compound": "Amiodarone",
          "left_atc": "C10AA02",
          "right_atc": "C01BD01",
          "left_formulation": "TAB",
          "right_formulation": "DFL",
          "left_medication": "Lovastatin AL 20mg",
          "right_medication": "Cordarex 150mg/3ml Injektionslösung",
          "left_dose": "20 mg",
          "right_dose": "141.98 mg"
        },
        {
          "plausibility": "plausible mechanism",
          "relevance": "severe",
          "frequency": "not known",
          "credibility": "high",
          "direction": "unidirectional interaction",
          "left_compound": "Lovastatin",
          "right_compound": "Diltiazem",
          "left_atc": "C10AA02",
          "right_atc": "C08DB01",
          "left_formulation": "TAB",
          "right_formulation": "RET",
          "left_medication": "Lovastatin AL 20mg",
          "right_medication": "Dilzem 120mg retard",
          "left_dose": "20 mg",
          "right_dose": "110.3 mg"
```

```
        },
        {
          "relevance": "no statement possible",
          "left_compound": "Lovastatin",
          "right_compound": "Amlodipine",
          "right_atc": "C08CA01",
          "right_formulation": "TAB",
          "right_medication": "Norvasc 5mg",
          "right_dose": "5 mg"
        }
      ],
      "unknown_compounds": [],
      "id": "2",
      "compounds": [
        "diltiazem",
        "amiodarone",
        "amlodipine",
        "lovastatin"
      ]
    }
  ],
  "timestamp": "2023-10-20 13:21:59",
  "api_version": "0.3.0"
}
```


POST /interactions/pzns

Input

Check for interactions based on PZNs (*Pharmazentralnummer*, a German product identifier for drugs) provided as *json*. Lists of PZNs must be provided matched to an *ID*:

```
  [
      {
          "id": "1",
          "pzns":["03041347","17145955","00592733","13981502"]
      },
      {
          "id": "2",
          "pzns":["03041347","17145955","00592733","13981502"]
      }
]
```

Example Usage

```
curl -X POST "https://abdata.clinicalpharmacy.me/api/interactions/pzns" \
     -H "Authorization: Bearer yourjwttoken" \
```

```
     -H "Content-Type: application/json" \
     -d '[{"id":"1","pzns":["03041347","17145955","00592733","13981502"]},
{"id":"2","pzns":["03041347","17145955","00592733","13981502"]}]'
```

Output

The return value for a successful **POST** request has the following structure:


```
{
  "results": [
    {
      "interactions": [
        {
          "plausibility": "plausible mechanism",
          "relevance": "minor",
          "frequency": "not known",
          "credibility": "weak",
          "direction": "unidirectional interaction",
          "left_PZN": "03041347",
          "right_PZN": "00592733"
        }
      ],
      "unknown_pzns": [],
      "id": "1",
      "pzns": [
        "03041347",
        "17145955",
        "00592733",
        "13981502"
      ]
    },
    {
      "interactions": [
        {
          "plausibility": "plausible mechanism",
          "relevance": "minor",
          "frequency": "not known",
          "credibility": "weak",
          "direction": "unidirectional interaction",
          "left_PZN": "03041347",
          "right_PZN": "00592733"
        }
      ],
      "unknown_pzns": [],
      "id": "2",
      "pzns": [
        "03041347",
        "17145955",
        "00592733",
        "13981502"
      ]
```

```
    }
  ],
  "timestamp": "2023-10-20 13:17:41",
  "api_version": "0.3.0"
}
```

GET /priscus/compounds

Input

Check for potentially inadequate medication for geriatric patients (Priscus 2.0) based on compound names provided as query parameters.

Example Usage

```
curl -X GET "https://abdata.clinicalpharmacy.me/api/priscus/compounds?
compounds=metoprolol,pindolol,diazepam" \
    -H "accept: application/json" \
    -H "Content-Type: application/json" \
    -H "Authorization: Bearer yourjwttoken"
```

Output

The return value for a successful **GET** request has the following structure:

```
{
  "priscus": [
    {
      "compound": "Metoprolol",
      "priscus": false
    },
    {
      "compound": "Pindolol",
      "priscus": true
    },
    {
      "compound": "Diazepam",
      "priscus": true
    }
  ],
  "unknown_compounds": [],
  "timestamp": "2024-05-16 12:31:53",
  "api_version": "0.6.0",
  "compounds": ["metoprolol", "pindolol", "diazepam"]
}
```

POST /priscus/compounds

Input

Check for potentially inadequate medication for geriatric patients (Priscus 2.0) based on compound names provided as *json*. Drug lists must be provided matched to an *ID*:

```
{
    [
        {
            "id": "1",
            "compounds": ["metoprolol","pindolol"]
        },
        {
            "id": "2",
            "compounds": ["diazepam","ranitidine","amlodipine","lovastatin"]
        }
    ]
}
```

Example Usage

```
curl -X POST "https://abdata.clinicalpharmacy.me/api/priscus/compounds" \
     -H "Authorization: Bearer yourjwttoken" \
     -H "Content-Type: application/json" \
     -d '[{"id":"1","compounds":["metoprolol","pindolol","diazepam"]},
{"id":"2","compounds":["diazepam","ranitidine","amlodipine","lovastatin"]}]'
```

Output

The return value for a successful **POST** request has the following structure:

```
{
  "results": [
    {
      "priscus": [
        {
          "compound": "Metoprolol",
          "priscus": false
        },
        {
          "compound": "Pindolol",
          "priscus": true
        },
        {
          "compound": "Diazepam",
          "priscus": true
        }
      ],
```

```
      "unknown_compounds": [],
      "id": "1",
      "compounds": ["metoprolol", "pindolol", "diazepam"]
    },
    {
      "priscus": [
        {
          "compound": "Diazepam",
          "priscus": true
        },
        {
          "compound": "Ranitidine",
          "priscus": true
        },
        {
          "compound": "Amlodipine",
          "priscus": false
        },
        {
          "compound": "Lovastatin",
          "priscus": false
        }
      ],
      "unknown_compounds": [],
      "id": "2",
      "compounds": ["diazepam", "ranitidine", "amlodipine", "lovastatin"]
    }
  ],
  "timestamp": "2024-05-16 12:58:51",
  "api_version": "0.6.0"
}
```


GET /priscus/pzns

Input

Check for potentially inadequate medicine for geriatric patients based on PZNs (*Pharmazentralnummer*, a German product identifier for drugs) provided as *query parameters*.

Example Usage

```
curl -X POST "https://abdata.clinicalpharmacy.me/api/interactions/pzns?
pzns=03967062,03041347,00592733" \
     -H "Authorization: Bearer yourjwttoken" \
     -H "Content-Type: application/json" \
```

Output

The return value for a successful **GET** request has the following structure:

```
{
  "priscus": [
    {
      "pzn": "00592733",
      "priscus": false
    },
    {
      "pzn": "03041347",
      "priscus": true
    },
    {
      "pzn": "03967062",
      "priscus": true
    }
  ],
  "unknown_pzns": [],
  "timestamp": "2024-05-16 13:13:05",
  "api_version": "0.6.0",
  "pzns": ["03967062", "03041347", "00592733"]
}
```


POST /priscus/pzns

Input

Check for potentially inadequate medicine for geriatric patients based on PZNs (*Pharmazentralnummer*, a German product identifier for drugs) provided as *json*. Lists of PZNs must be provided matched to an *ID*:

```
[
  {
    "id": "1",
    "pzns":["03041347","17145955","00592733","13981502"]
  },
  {
    "id": "2",
    "pzns":["03041347","17145955","00592733","13981502"]
  }
]
```

Example Usage

```
curl -X POST "https://abdata.clinicalpharmacy.me/api/interactions/pzns" \
     -H "Authorization: Bearer yourjwttoken" \
     -H "Content-Type: application/json" \
     -d '[{"id":"1","pzns":["03041347","17145955","00592733","13981502"]},
{"id":"2","pzns":["03041347","17145955","00592733","13981502"]}]'
```

Output

The return value for a successful **POST** request has the following structure:


```
{
  "results": [
    {
      "priscus": [
        {
          "pzn": "00592733",
          "priscus": false
        },
        {
          "pzn": "03041347",
          "priscus": true
        },
        {
          "pzn": "13981502",
          "priscus": false
        },
        {
          "pzn": "17145955",
          "priscus": false
        }
      ],
      "unknown_pzns": [],
      "id": "1",
      "pzns": ["03041347", "17145955", "00592733", "13981502"]
    },
    {
      "priscus": [
        {
          "pzn": "00592733",
          "priscus": false
        },
        {
          "pzn": "03041347",
          "priscus": true
        },
        {
          "pzn": "13981502",
          "priscus": false
        },
        {
          "pzn": "17145955",
          "priscus": false
        }
      ],
      "unknown_pzns": [],
      "id": "2",
      "pzns": ["03041347", "17145955", "00592733", "13981502"]
    }
```

```
  ],
  "timestamp": "2024-05-16 12:58:51",
  "api_version": "0.6.0"
}
```

GET /atcs/drugs

Input

Get drug names based on ATCs. Please note, that some ATCs may not resolve to a unique drug product, especially in case of fixed drug combinations

Example Usage

```
curl -X GET "https://abdata.clinicalpharmacy.me/api/atcs/drugs?
atcs=C01BD01,C08DB01,C08DA01,J01CR02" \
    -H "accept: */*" \
    -H "Authorization: Bearer yourjwttoken"
```

Output

The return value for a successful **GET** request has the following structure:

```
{
    "names": [
        {
            "atc": "C01BD01",
            "name_german":"Amiodaron",
            "name_english":"Amiodarone"
        },
        {
            "atc": "C08DA01"
            ,"name_german": "Verapamil",
            "name_english":"Verapamil"
        },
        {
            "atc": "C08DB01",
            "name_german": "Diltiazem",
            "name_english": "Diltiazem"
        },
        {
            "atc": "J01CR02",
            "name_german": "Amoxicillin und Beta-Lactamase-Inhibitor",
            "name_english":"Amoxicillin and beta-lactamase inhibitor"
        }
    ],
    "unknown_atcs":[],
    "timestamp": "2023-10-20 10:56:21",
```

```
    "api_version": "0.3.0",
    "atcs": ["C01BD01","C08DB01","C08DA01","J01CR02"]
}
```

GET /pzns/products

Input

Get product names and ATC codes based on PZNs. Please note, that some PZNs may not be up to date.

Example Usage

```
curl -X GET "https://abdata.clinicalpharmacy.me/api/pzns/products?
pzns=03967062,03041347,00592733" \
    -H "accept: */*" \
    -H "Authorization: Bearer yourjwttoken"
```

Output

The return value for a successful **GET** request has the following structure:

```
{
    "products": [
    {
        "pzn":"00592733",
        "product":"Famotidin STADA 40mg",
        "atc":"A02BA03"
    },
    {
        "pzn":"03041347",
        "product":"Domperidon AbZ 10mg",
        "atc":"A03FA03"
    },
    {
        "pzn":"03967062",
        "product":"MCP-ratiopharm 10mg",
        "atc":"A03FA01"
    }
    ],
    "unknown_pzns":[],
    "timestamp":"2024-02-22 09:05:55",
    "api_version":"0.3.0",
    "pzns":[
        "03967062",
        "03041347",
        "00592733"
    ]
}
```