

API Documentation for ElectricityTrendService

This API manages mainly the retrieval of community-related electricity trends, especially from the **Community** and **CommunityTrends** table and considering other relationships.

Service Description

The ElectricityTrendService is a software service that plays a crucial role in giving anonymized data to the researchers in the form of trends which is the functional requirement of the business service BS4. It analyzes and forms trends in electricity usage, consumption, and trading over time. In addition, it also provides researchers with data about community-specific efforts for the shared energy concept in the energy commons domain on what they are doing to save how much energy. This software service works together with **ElectricityDataDeliveryService** to deliver its functionality to the end user which is **researcher** in our case. Likewise, most of the operations are Read only as researchers are not allowed to edit or delete the data that this service is providing.

General Error Codes

The service uses the following error response codes for all endpoints:

- **401 Unauthorized**: for endpoints that require authentication, this status code is used if no authentication is included in the request.
- **403 Forbidden**: for endpoints that require authentication, this status code is used if authentication is provided but the role / privileges are insufficient. For instance, to access these apis, the role should be a **researcher**.
- **404 Not Found**: this status code is used if a non-existent endpoint is invoked.
- **500 Internal Server Error**: this status code is used when an unhandled exception occurs, which should ideally never happen.

Reused JSON Objects

Community --> an object representing a community with the following attributes:

- **communityId** (int): a numeric identifier for the community, generated by the service.
- **communitySize** (int): the int of households in the community.
- **geographicRegion** (string): the region where the community is located.
- **powerSources** (array of strings): the alternative energy sources used by the community.
- **devices** (array of objects): a list of energy-generating devices in the community, where each object includes:
 - **type** (string): the type of device.
 - **count** (int): the int of devices of that type.
- **communityEngagementLevel** (object): metrics reflecting the community's participation and engagement, including:

- **participationRate** (double): the percentage of active members in the community.
- **activeMembers** (int): the count of members actively engaged in community activities.

CommunitySummary --> an object representing a community summary with the following attributes:

- **communityDetails** (**Community**): a JSON object of the Community
- **environmentalImpact** (**EnvironmentalImpact**): a JSON object of the EnvironmentalImpact
- **trendsStatistics** (**TrendsStatistics**): a JSON object of the TrendsStatistics

TrendsSummary --> an object representing aggregated information about electricity trends, combining consumption, production, and trading summaries.

- **consumptionSummary** (**ConsumptionSummary**): a JSON object of the ConsumptionSummary
- **productionSummary** (**ProductionSummary**): a JSON object of the ProductionSummary
- **tradingSummary** (**TradingSummary**): a JSON object of the TradingSummary

ConsumptionSummary --> an object representing an aggregated information about electricity consumption with following attributes:

- **totalConsumption** (double): Total electricity consumed over the date range.
- **peakDemand** (double): The peak electricity demand observed.
- **peakHours** (array of Strings): An array of peak hours during which consumption is highest.
- **seasonalVariations** (object): Insights on seasonal consumption variations:
 - **summer** (object): Contains:
 - **averageConsumption** (double): Average consumption during summer.
 - **averageProduction** (double): Average production during summer.
 - **winter** (object): Contains:
 - **averageConsumption** (double): Average consumption during winter.
 - **averageProduction** (double): Average production during winter.

ProductionSummary --> an object representing an aggregated information about electricity production:

- **totalProduction** (double): Total electricity produced over the date range.
- **renewableSourcesUsed** (double): The int of renewable sources used.
- **environmentalImpact** (**EnvironmentalImpact**): Insights on the environmental impact of the production:

TradingSummary --> an object representing an aggregated information about electricity trading:

- **totalTradedElectricity** (double): Total electricity traded over the date range.
- **currency** (string): The currency used.
- **averageTradePrice** (double): The average price at which electricity was traded.
- **mostTradedPeriod** (object): The time period during which trading was highest:
 - **startTime** (string): Start time of the most traded period.
 - **endTime** (string): End time of the most traded period.

EnvironmentalImpact --> an object representing the sustainability efforts of a community with the following attributes:

- **estimatedCO2Savings** (double): the estimated reduction in CO2 emissions due to the community's sustainable practices.

- **renewableEnergyUsage** (double): the percentage of energy consumption that comes from renewable sources.

TrendsStatistics --> an object that provides statistical data on the community's energy consumption, production and trading with the following attributes:

- **averageConsumption** (double): the average amount of energy consumed by the community over a specified period.
- **totalProduction** (double): the total amount of energy generated by the community's power sources.
- **tradedEnergy** (double): the quantity of energy that has been traded within the community.
- **intOfDevices** (int): the total count of energy-related devices installed in the community.

Endpoints

GET /communities

Description: retrieve all existing communities in the repository, potentially sorted by communityId

Auth: Requires authentication and the **researcher** role

Path parameters: none

Query parameters:

- **sortBy** (optional, default: **communityId**): indicates the attribute by which the results will be sorted, either **communityId** or **communitySize**
- **sortOrder** (optional, default: **asc**): either **asc** or **desc** to indicate ascending or descending sort order
- **page** (optional, default: 1): Indicates the current page of results to retrieve.
- **pageSize** (optional, default: 10): Indicates the int of results to return per page.

Request body: none

Response body: a JSON object with the following attributes:

- **intOfResults** (int): the int of found results
- **totalResults** (int): the total int of communities available in the database.
- **page** (int): the current page int of the results being returned. - **pageSize** (int): the int of results returned per page.
- **sortBy** (string): the used attribute for sorting
- **sortOrder** (string): the used sort order
- **results** (array of **Community**'s): an array of the found communities

Example:

```
// request
GET /communities?sortBy=communityId&sortOrder=asc&page=1&pageSize=2

// response
{
```

```
"message": "Community list retrieved successfully.",
"intOfResults": 2,
"totalResults": 5,
"page": 1,
"pageSize": 2,
"results": [
  {
    "communityId": 1,
    "communitySize": 150,
    "geographicRegion": "North Region",
    "powerSources": ["solar", "wind"],
    "devices": [
      {
        "type": "solar panel",
        "count": 50
      },
      {
        "type": "wind turbine",
        "count": 20
      }
    ],
    "communityEngagementLevel": {
      "participationRate": 75.0,
      "activeMembers": 100
    }
  },
  {
    "communityId": 2,
    "communitySize": 100,
    "geographicRegion": "South Region",
    "powerSources": ["solar", "wind", "geothermal"],
    "devices": [
      {
        "type": "solar panel",
        "count": 80
      },
      {
        "type": "wind turbine",
        "count": 30
      },
      {
        "type": "geothermal heat pump",
        "count": 15
      }
    ],
    "communityEngagementLevel": {
      "participationRate": 50.0,
      "activeMembers": 50
    }
  }
]
```

Success response code: 200 Success

Specific error codes:

- **403 Forbidden:** this status code is used if the authenticated user does not have the necessary permissions to access the requested resource or perform the requested action. For instance, the role should be **researcher** to access this API.

GET /communities/:id

Description: retrieve the details of an existing community-specific data along with summary of the insights on environmental impacta and different energy-related patterns of the community.

Auth: Requires authentication and the **researcher** role

Path parameters:

- **id:** the id of the community to be retrieved

Query parameters: none

Request body: none

Response body:

- **message** (int): a message about success or error
- **result** (**CommunitySummary**): an object containing detailed summary about the community including community related info to community contributions to environmental impact.

Example:

```
// request
GET /communities/2

// response
{
  "message": "Community-specific insights retrieved successfully.",
  "result": {
    "communityDetails": {
      "communityId": 1,
      "communitySize": 150,
      "powerSources": ["solar", "wind"],
      "devices": [
        {
          "type": "solar panel",
          "count": 50
        },
        {
          "type": "battery storage",
          "count": 20
        }
      ]
    },
    "communityEngagementLevel": {
```

```
        "participationRate": 75.0,
        "activeMembers": 100
    },
    "environmentalImpact": {
        "estimatedCO2Savings": 500.0,
        "renewableEnergyUsage": 75.0
    },
    "trendsStatistics": {
        "averageConsumption": 200.5,
        "totalProduction": 5000,
        "tradedEnergy": 300,
        "intOfDevices": 70
    }
}
```

Success response code: 200 OK

Specific error codes:

- **404 Not Found:** this status code is used if the community **id** indicated via the path parameter doesn't exist.

GET /communities/:id/electricity-trends

Description: fetches the electricity trends of a specific community based on various query parameters like time range and trends data type (consumption, production, or trading). This endpoint also supports filtering trends daily, weekly, monthly.

Auth: requires authentication and the **admin** role

Path parameters:

- **id:** the id of the community whose trends are to be fetched

Query parameters:

- **startDate** (string, optional): The start date for the trend analysis in YYYY-MM-DD. By default, it provides the start date of the current month (e.g., "2024-10-01" for October).
- **endDate** (string, optional): The end date for the trend analysis in YYYY-MM-DD. y default, it provides the end date of the current month (e.g., "2024-10-31" for October).
- **trendInterval** (string, optional, default: **daily**): The type of trend data to retrieve. Supported values: **daily**, **weekly**, **monthly**, **yearly**.
- **trendCategory** (string, optional, default: **all**): The type of data to retrieve. Supported values: **consumption**, **production**, **trading**, **all**.

Request body: none

Response body: a JSON object with the following attributes:

- **message** (string): a message about the success or error of fetching the trends

- **communityId** (string): the id of the community for which the trends are fetched.
- **trendInterval** (string): the interval type for the trend data retrieved.
- **dateRange** (object): an object containing:
 - **startDate** (string): the start date for which trends are fetched.
 - **endDate** (string): the end date for which trends are fetched..
- **result** (object): this object contains combination of multiple objects containing:
 - **trendsData** (array of **TrendData**): an array of trends data in specified interval for the specified date range
 - **trendsSummary** (**TrendsSummary**): an object of the fetched summary for electricity trends

Example:

```
// request
GET /communities/1/electricity-trends?startDate=2024-10-01&endDate=2024-10-07&trendInterval=weekly&trendCategory=all

// response
{
  "message": "Electricity trends retrieved successfully.",
  "communityId": "1",
  "trendInterval": "weekly",
  "dateRange": {
    "startDate": "2024-10-01",
    "endDate": "2024-10-07"
  },
  "result": {
    "trendsData": [
      {
        "date": "2024-10-01",
        "electricity": {
          "consumption": {
            "amount": 2200,
            "unit": "KWHs"
          },
          "production": {
            "amount": 2500,
            "unit": "KWHs"
          },
          "traded": {
            "amount": 300,
            "unit": "KWHs",
            "price": {
              "amount": 45.0,
              "currency": "EUR"
            }
          }
        }
      }
    ],
    "trendsSummary": {
      "consumptionSummary": {
```

```
    "totalConsumption": 14000,
    "peakDemand": 18000,
    "peakHours": ["18:00", "19:00", "20:00"],
    "seasonalVariations": {
      "summer": {
        "averageConsumption": 2500.0,
        "averageProduction": 2000.0
      },
      "winter": {
        "averageConsumption": 3000.0,
        "averageProduction": 1500.0
      }
    },
    "productionSummary": {
      "totalProduction": 15000,
      "renewableSourcesUsed": 5,
      "environmentalImpact": {
        "estimatedCO2Savings": 500.0,
        "renewableEnergyUsage": 75.0
      }
    },
    "tradingSummary": {
      "totalTradedElectricity": 2000,
      "averageTradePrice": 50.0,
      "mostTradedPeriod": {
        "startTime": "09:00",
        "endTime": "11:00"
      }
    }
  }
}
```

Success response code: 200 OK

Specific error codes:

- **404 Not Found:** this status code is used if the community **id** indicated via the path parameter doesn't exist.