

# GeoJSONファイル入出力仕様書

## 概要

本仕様書は、GeoReferencerで使用するGeoJSONファイルの構造と入出力形式を定義する。

## 基本構造

```
{
  "type": "FeatureCollection",
  "features": [...]
}
```

## Feature タイプ別仕様

### 1. ポイントGPS

主要なポイントのGPS座標情報を格納する。

#### プロパティ構造

```
{
  "type": "Feature",
  "properties": {
    "id": "文字列（必須）",
    "name": "文字列（必須）",
    "type": "ポイントGPS（必須）",
    "source": "GPS_Excel（必須）",
    "description": "文字列（必須）",
    "notes": "文字列（オプション）"
  },
  "geometry": {
    "type": "Point",
    "coordinates": [経度, 緯度, 標高]
  }
}
```

#### フィールド詳細

フィールド	型	必須	説明	例
id	string	○	識別子	"J-05"
name	string	○	名前、地点名	"東海道自然歩道"
type	string	○	固定値	"ポイントGPS"

フィールド	型	必須	説明	例
source	string	○	固定値	"GPS_Excel"
description	string	○	固定値	"緊急ポイント（Excel管理GPS値）"
notes	string	-	備考	""
coordinates	array	○	[経度, 緯度, 標高]	[135.49331, 34.87202, 564.7]

2. ルート中間点

ルート上の中間地点情報を格納する。

プロパティ構造

```
{
  "type": "Feature",
  "properties": {
    "id": "文字列（導出値）",
    "name": "文字列（必須）",
    "type": "route_waypoint（必須）",
    "source": "image_transformed（読み込み時）"/"route_editor（出力時）",
    "route_id": "文字列（導出値）",
    "description": "文字列（必須）"
  },
  "geometry": {
    "type": "Point",
    "coordinates": [経度, 緯度, 標高]
  }
}
```

フィールド詳細

フィールド	型	必須	説明	例
id	string	○	中間点ID	"route_C-03_to_J-01_waypoint_06"
name	string	○	中間点名	"waypoint_06"
type	string	○	固定値	"route_waypoint"
source	string	-	文字列	"image_transformed"/"route_editor"
route_id	string	○	ルートID	"route_C-03_to_J-01"
description	string	○	固定値	"ルート中間点"
coordinates	array	○	[経度, 緯度, 標高]	[135.49353, 34.86449, 564.7]

注意

- 標高はオプション
- route\_idは、"route\_"+開始ポイント+"to"+終了ポイント
- idは、route\_id+"\_"+name(=中間点名)
- nameは、"waypoint\_"+中間点連番(2桁)
- sourceは、GeoJSON出力の際は"routeEditor"。読み込み時は問わない。

3. スポット

休憩所や施設などの地点情報を格納する。

プロパティ構造

```
{
  "type": "Feature",
  "properties": {
    "id": "文字列（導出値）",
    "name": "文字列（必須）",
    "type": "spot（必須）",
    "source": "image_transformed（読み込み時）"/"route_editor（出力時）",
    "description": "文字列（必須）"
  },
  "geometry": {
    "type": "Point",
    "coordinates": [経度, 緯度, 標高]
  }
}
```

フィールド詳細

フィールド	型	必須	説明	例
id	string	○	自動生成ID	"spot08_薬師堂"
name	string	○	スポット名	"薬師堂"
type	string	○	固定値	"spot"
source	string	-	文字列	"image_transformed"/"route_editor"
description	string	○	固定値	"スポット"
coordinates	array	○	[経度, 緯度, 標高]	[135.49052, 34.86557, 564.7]

注意

- 標高はオプション
- idは、"spot"+中間点連番(2桁)+"\_"+name(=スポット名)
- sourceは、GeoJSON出力の際は"routeEditor"。読み込み時は問わない。

データソース分類

## GPS

- 実測されたGPS座標データ
- 精度が高く、標高情報も含む

## image\_transformed

- 画像からジオリファレンス変換されたデータ
- 地図画像や航空写真から抽出された座標情報

## ファイル形式

- **フォーマット**: GeoJSON FeatureCollection
- **座標系**: WGS84 (EPSG:4326)
- **文字エンコーディング**: UTF-8

## 座標系

- **座標系**: WGS84 (EPSG:4326)
- **形式**: [経度, 緯度, 標高 (オプション)]
- **単位**: 度 (decimal degrees)、標高はメートル
- **精度**: 小数点以下5桁 (約1m精度)
- **標高精度**: 小数点以下1桁 (約0.1m精度)

## マーカー属性

- **ポイントGPS**: 緑色(#ff0000)、円形、サイズ:半径6px
- **ルート中間点**: 橙色(#f58220)、菱形(=ダイヤモンド型)(transform: rotate(45deg))、サイズ:8x8px
- **スポット**: 青色(#0000ff)、正方形、サイズ:12x12px

## バリデーションルール

### 必須フィールドチェック

- すべての必須フィールドが存在すること
- 座標値が有効な範囲内にあること (経度: -180~180、緯度: -90~90)

### データ型チェック

- coordinates配列は数値のみ
- 文字列フィールドは空文字列でないこと (notesを除く)

### 一意性チェック

- 各Featureのidは一意であること

## 使用例

```
{  
  "type": "FeatureCollection",
```

```
"features": [  
  {  
    "type": "Feature",  
    "properties": {  
      "id": "J-05",  
      "name": "東海道自然歩道",  
      "type": "ポイントGPS",  
      "source": "GPS",  
      "description": "緊急ポイント（Excel管理GPS値）",  
      "notes": ""  
    },  
    "geometry": {  
      "type": "Point",  
      "coordinates": [135.49331, 34.87202, 564.7]  
    }  
  }  
]  
}
```

## 更新履歴

- 初版: 2025年09月29日