

PMS-v2 API Documentation

/api/users

GET /get-all :

- Protection: none
- Request: no data required
- Reponse: returns all registered users

GET /logout :

- Protection: none
- Request: no data required
- Response: removes cookies: **REFRESH_TOKEN**, **ACCESS_TOKEN**

POST /register-admin :

- Protection: none
- Request:
 - required: **name(max:50)**,
surname(max:50),
password(min:8)
- Response: create admin account if not exist

POST /login-user :

- Protection: none
- Request:
 - required: **idUser**, **password**
- Reponse: returns user payload and creates **REFRESH_TOKEN** cookie

POST /insert :

- Protection: auth with **JWT**, specific role auth: **ADMIN**
- Request:
 - required: **name(max:50)**,
surname(max:50),
password(min:8),
role([„ADMIN”, „SEKRETARIAT”, „KSIEGOWOSC”, „TEREN”])
- Reponse: creates new user

PUT /update :

- Protection: auth with **JWT**, specific role auth: **ADMIN**
- Request:
 - required: **idUser**,
name(max:50),
surname(max:50),
role([„ADMIN”, „SEKRETARIAT”, „KSIEGOWOSC”, „TEREN”])
- Reponse: updates user

PUT /update-password :

- Protection: auth with **JWT**, specific role auth: **ADMIN**
- Request:
 - Required: **idUser**, **password(min:8)**
- Response: updates user password

DELETE /delete:

- Protection: auth with **JWT**, specific role auth: **ADMIN**
- Request:
 - Required: **idUser**
- Response: deletes user

/api/mpzp

GET /get-all :

- Protection: auth with **JWT**,
- Request: no data required
- Reponse: returns all available **MPZP**

POST /insert :

- Protection: auth with **JWT**, specific role auth: **ADMIN**
- Request:
 - Required: **code(max:5)**, **description(max:70)**
- Response: creates new **MPZP**

POST /insert-file:

- Protection: auth with **JWT**, specific role auth: **ADMIN**
- Request: no data required
- Response: creates new **MPZP** from mpzp.json file

PUT /update:

- Protection: auth with **JWT**, specific role auth: **ADMIN**
- Request:
 - Required: **idMpzp**, **code(max:5)**, **description(max:70)**
- Response: updates **MPZP**

DELETE /delete :

- Protection: auth with **JWT**, specific role auth: **ADMIN**
- Request:
 - Required: **idMpzp**
- Response: deletes **MPZP**

/api/general-plans

GET /get-all :

- Protection: auth with **JWT**,
- Request: no data required
- Reponse: returns all available **General Plans**

POST /insert :

- Protection: auth with **JWT**, specific role auth: **ADMIN**
- Request:
 - Required: **code(max:5)**, **description(max:70)**
- Response: creates new **General Plan**

POST /insert-file:

- Protection: auth with **JWT**, specific role auth: **ADMIN**
- Request: no data required
- Response: creates new **General Plans** from generalPlans.json file

PUT /update:

- Protection: auth with **JWT**, specific role auth: **ADMIN**
- Request:
 - Required: **idGeneralPlan**, **code(max:5)**, **description(max:70)**
- Response: updates **General Plan**

DELETE /delete :

- Protection: auth with **JWT**, specific role auth: **ADMIN**
- Request:
 - Required: **idGeneralPlan**
- Response: deletes **General Plan**

/api/land-purposes

GET /get-all :

- Protection: auth with **JWT**,
- Request: no data required
- Reponse: returns all available **Land Purposes**

POST /insert :

- Protection: auth with **JWT**, specific role auth: **ADMIN**
- Request:
 - Required: **type(max:50)**
- Response: creates new **Land Purpose**

POST /insert-file:

- Protection: auth with **JWT**, specific role auth: **ADMIN**
- Request: no data required
- Response: creates new **Land Purposes** from landPurposes.json file

PUT /update:

- Protection: auth with **JWT**, specific role auth: **ADMIN**
- Request:
 - Required: **idLandPurpose, type(max:50)**
- Response: updates **Land Purpose**

DELETE /delete :

- Protection: auth with **JWT**, specific role auth: **ADMIN**
- Request:
 - Required: **idLandPurpose**
- Response: deletes **Land Purpose**

/api/land-types

GET /get-all :

- Protection: auth with **JWT**,
- Request: no data required
- Reponse: returns all available **Land Types**

POST /insert :

- Protection: auth with **JWT**, specific role auth: **ADMIN**
- Request:
 - Required: **type(max:50)**
- Response: creates new **Land Type**

POST /insert-file:

- Protection: auth with **JWT**, specific role auth: **ADMIN**
- Request: no data required
- Response: creates new **Land Types** from landTypes.json file

PUT /update:

- Protection: auth with **JWT**, specific role auth: **ADMIN**
- Request:
 - Required: **idLandType, type(max:50)**
- Response: updates **Land Type**

DELETE /delete :

- Protection: auth with **JWT**, specific role auth: **ADMIN**
- Request:
 - Required: **idLandType**
- Response: deletes **Land Type**

/api/owners

GET /get :

- Protection: auth with **JWT**,
- Request:
 - Optional: **nameFilter**, **limit(int min:0)**
- Response: returns filtered owners and their lands

POST /insert :

- Protection: auth with **JWT**
- Request:
 - Required: **name(max:100)**
 - Optional: **phone(polish phone format)**
- Response: creates new **Owner**

PUT /update:

- Protection: auth with **JWT**
- Request:
 - Required: **idOwner**, **name(max:100)**
 - Optional: **phone(polish phone format)**
- Response: updates **Owner**

DELETE /delete :

- Protection: auth with **JWT**, specific role auth: **ADMIN**
- Request:
 - Required: **idOwner**
- Response: deletes **Owner**

/api/renters

GET /get-all :

- Protection: auth with **JWT**,
- Request: no data required
- Response: returns all **Renters**

GET /get :

- Protection: auth with **JWT**,
- Request:
 - Optional: **monthFilter(int)**, **nameFilter**, **endYearFilter(int)**, **ownerNameFilter**, **showExpired(bool)**, **limit(int, min:0)**
- Response: returns filtered renters and their rents

POST /insert :

- Protection: auth with **JWT**
- Request:
 - Required: **name(max:100)**, **phone(polish phone format)**
- Response: creates new **Renter**

PUT /update:

- Protection: auth with **JWT**
- Request:
 - Required: **idRenter**, **name(max:100)**, **phone(polish phone format)**
- Response: updates **Renter**

DELETE /delete :

- Protection: auth with **JWT**, specific role auth: **ADMIN**
- Request:
 - Required: **idRenter**

- Response: deletes **Renter**