

# Device App Register

## Endpoints Summary

Method	Path	Swagger
POST	<a href="/device_app_register/">/device_app_register/</a>	<a href="#">Swagger ↗</a>

“ The Device App Register API provides functionality for registering mobile and desktop applications with the backend system. This endpoint captures comprehensive device and user information including hardware identifiers, network details, operating system information, and user credentials for device enrollment and management purposes.

**Base URL:** `https://gate.zequenze.com/api/v1`

**Authentication:** All endpoints require a Bearer token:

```
Authorization: Bearer <your-api-token>
```

## Overview

The Device App Register API is designed for initial device and application enrollment in the Zequenze platform. This API category handles the critical first step of device onboarding, where mobile apps, desktop applications, or other client software register themselves and their associated devices with the backend system.

### Key Capabilities:

- **Device Identification:** Captures multiple hardware identifiers (IMEI, IMSI, ICCID, MAC address) for comprehensive device tracking
- **Operating System Detection:** Records detailed OS information including family, version, and API level
- **Network Context:** Identifies home mobile network information and carrier details
- **User Association:** Links devices to user accounts with flexible authentication options

- **Device Categorization:** Supports classification across device types (desktop, laptop, tablet, mobile phone)

### Common Integration Scenarios:

- Mobile app first-launch registration
- Desktop application installation and activation
- IoT device provisioning and enrollment
- User device inventory management
- Compliance and security auditing

The registration process typically occurs during application installation or first run, establishing the device identity that will be used for subsequent API interactions, push notifications, and device management operations.

---

# Endpoints

## POST /device\_app\_register/

**Description:** Registers a new device and application instance with the backend system. This endpoint should be called during app installation or first launch to establish the device identity and associate it with user credentials. The registration captures comprehensive device information for inventory management, security tracking, and push notification delivery.

### Use Cases:

- Mobile app first-time setup and device enrollment
- Desktop application activation and licensing
- IoT device provisioning in enterprise environments
- User device inventory for compliance and security auditing
- Establishing push notification delivery channels

### Full URL Example:

```
https://gate.zequenze.com/api/v1/device_app_register/
```

### Request Body Parameters:

Parameter	Type	Required	Description
number	string	No	Device phone number (MSISDN) for mobile devices

Parameter	Type	Required	Description
generic_id	string	No	Unique identifier from OS/App store (e.g., Android ID, iOS IDFA)
imei	string	No	International Mobile Equipment Identity for cellular devices
imsi	string	No	International Mobile Subscriber Identity from SIM card
iccid	string	No	Integrated Circuit Card Identifier from SIM card
mac_addr	string	No	MAC address of primary network interface
home_mobile_network_name	string	No	Carrier name (e.g., "Verizon", "AT&T", "Vodafone")
home_mobile_network_mcc_mnc	string	No	Mobile Country Code and Mobile Network Code (e.g., "310410")
app_version	string	No	Version of the registering application
os_family	string	No	Operating system family (e.g., "Android", "iOS", "Windows", "macOS")
os_version	string	No	Operating system version (e.g., "14.0", "Windows 11")
os_api_version	string	No	OS API level or SDK version
device_family	string	No	Device family or series (e.g., "iPhone", "Galaxy", "ThinkPad")
device_brand	string	No	Device manufacturer (e.g., "Apple", "Samsung", "Lenovo")
device_model	string	No	Specific device model (e.g., "iPhone 15 Pro", "Galaxy S24")
device_type	string	No	Device category: 'de' (Desktop), 'la' (Laptop), 'tb' (Tablet), 'ph' (Mobile phone)
push_key	string	No	Push notification token for message delivery

Parameter	Type	Required	Description
username	string	No	User account username for device association
password	string	No	User account password for authentication
user_external_id	string	No	External system user identifier for integration
user_class	string	No	User classification or service tier

### cURL Example:

```
curl -X POST "https://gate.zequence.com/api/v1/device_app_register/" \  
-H "Authorization: Bearer YOUR_API_TOKEN" \  
-H "Content-Type: application/json" \  
-d '{  
  "number": "+1234567890",  
  "generic_id": "android-id-12345",  
  "imei": "123456789012345",  
  "mac_addr": "00:1B:44:11:3A:B7",  
  "home_mobile_network_name": "Verizon Wireless",  
  "home_mobile_network_mccmnc": "311480",  
  "app_version": "2.1.0",  
  "os_family": "Android",  
  "os_version": "14.0",  
  "os_api_version": "34",  
  "device_family": "Galaxy",  
  "device_brand": "Samsung",  
  "device_model": "Galaxy S24",  
  "device_type": "ph",  
  "push_key": "fcm-token-abc123xyz789",  
  "username": "john.doe",  
  "password": "securePassword123",  
  "user_class": "premium"  
}'
```

### Example Response:

```
{  
  "number": "+1234567890",  
  "generic_id": "android-id-12345",
```

```
"imei": "123456789012345",
"imsi": "311480123456789",
"iccid": "89014103271234567890",
"mac_addr": "00:1B:44:11:3A:B7",
"home_mobile_network_name": "Verizon Wireless",
"home_mobile_network_mccmnc": "311480",
"app_version": "2.1.0",
"os_family": "Android",
"os_version": "14.0",
"os_api_version": "34",
"device_family": "Galaxy",
"device_brand": "Samsung",
"device_model": "Galaxy S24",
"device_type": "ph",
"push_key": "fcm-token-abc123xyz789",
"username": "john.doe",
"user_external_id": "ext-user-456",
"user_class": "premium"
}
```

### Response Codes:

Status	Description
201	Created - Device successfully registered with the system
400	Bad Request - Invalid data format or missing required fields
401	Unauthorized - Invalid or missing authentication token
409	Conflict - Device already registered (duplicate IMEI/MAC address)
422	Unprocessable Entity - Valid format but business logic validation failed

# Common Use Cases

## Use Case 1: Mobile App First Launch Registration

When a user installs and first opens a mobile app, the application collects device information and registers with the backend. This establishes the device identity for future API calls and enables push notifications.

**Required endpoints:** POST /device\_app\_register/

## Use Case 2: Enterprise Device Enrollment

IT administrators deploying company devices use the registration API to enroll devices in the corporate management system, associating them with employee accounts and applying appropriate device policies.

**Required endpoints:** POST /device\_app\_register/

## Use Case 3: IoT Device Provisioning

Smart devices and IoT endpoints register themselves during initial setup, providing hardware identifiers and network information for device management and monitoring.

**Required endpoints:** POST /device\_app\_register/

## Use Case 4: Cross-Platform Device Inventory

Applications running on multiple platforms (mobile, desktop, web) register their host devices to maintain a unified inventory across all user touchpoints for security and compliance reporting.

**Required endpoints:** POST /device\_app\_register/

## Use Case 5: Push Notification Setup

Mobile and desktop applications register their push notification tokens during the device registration process, enabling the backend to deliver messages and alerts to specific devices.

**Required endpoints:** POST /device\_app\_register/

---

## Best Practices

- **Privacy Compliance:** Only collect device identifiers that are necessary for your use case. Be aware of privacy regulations (GDPR, CCPA) when collecting IMEI, MAC addresses, or other persistent identifiers.

- **Unique Device Identification:** Use multiple identifiers when available (IMEI + MAC address + generic\_id) to create robust device fingerprinting while having fallback options if some identifiers are unavailable.
  - **Secure Credential Handling:** If passing username/password in registration, ensure the API connection uses HTTPS and consider using temporary tokens or OAuth flows instead of permanent passwords.
  - **Device Type Classification:** Always specify the device\_type parameter to enable proper device management policies and user experience optimization based on device capabilities.
  - **Error Handling:** Implement retry logic for network failures, but avoid re-registering devices that return 409 (Conflict) responses, as this indicates the device is already known to the system.
  - **Push Token Management:** Keep push\_key tokens updated as they can expire or change when apps are updated or reinstalled. Consider implementing a separate endpoint update mechanism for token refresh.
  - **Data Validation:** Validate device identifiers format before sending (e.g., IMEI should be 15 digits, MAC addresses should follow standard format) to avoid unnecessary API calls and improve registration success rates.
- 
- 

Revision #4

Created 2026-02-04 05:13:53 UTC by ipena@zequenze.com

Updated 2026-02-11 03:15:38 UTC by ipena@zequenze.com