IoT Private Platform User Manual

Version: 1.0.0

Date: 2024-10-21

Status: Preliminary

About the Document

Revision History

Version	Date	Description
1.0.0	2024-10-21	Preliminary

Contents

About the Document **2**
Contents **3**
Table Index **4**

1 Log in & Forgot Password **5**
1.1. Log in 5
1.2. Forgot Password 5

2 Develop Product **6**
2.1. Manage Product 6
2.2. Define Feature 9
2.2.1. TSL Model Overview 9
2.2.2. Manage Feature 12
2.2.2.1. Add Feature 12
2.2.2.1.1. Add a Custom Property 13
2.2.2.1.2. Add a Custom Event 14
2.2.2.1.3. Add a Custom Service 16
2.2.2.2. Add TSL Features in Batches 17

3 Manage Device **18**
3.1. Device Maintenance 18
3.1.1. Device List 18
3.1.2. Pre-import Management 18
3.1.3. Device Information 19
3.1.4. Data Logs 22
3.1.5. Exception Logs 23
3.1.6. Property Logs 23
3.1.7. Event Logs 24
3.1.8. Service Logs 25
3.1.9. Device Debugging 25
3.1.10. Device Positioning 27

4 Personal Center **28**
4.1. Personal Account 28
4.1.1. My Profile 28
4.1.2. Password Change 28
4.1.3. Security Settings 28
4.2. Sub-account Management 28
4.3. Logs 29
4.4. Custom Configuration 29
4.4.1. Style Configuration 29
4.4.2. Email Configuration 30

Table Index

Table 1: Product Creation 7
Table 2: Feature Type 9
Table 3: Supported Data Types 10
Table 4: TSL File Definition 11
Table 5: Property Parameters 14
Table 6: Event Parameters 15
Table 7: Service Parameters 16
Table 8: Device List Actions 18
Table 9: Pre-import Management for Other Device Actions 19
Table 10: Basic Information 21
Table 11: More Device Information 21
Table 12: Data Logs Parameters 22
Table 13: Exception Logs Parameters 23
Table 14: Property Logs Parameters 24
Table 15: Event Logs Parameters 24
Table 16: Service Logs Parameters 25
Table 17: Transparent Transmission Data Debugging Parameters 26
Table 18: Device Positioning Parameters 27
Table 19: Sub-account Management Parameters 28
Table 20: Logs Parameters 29

1 Log in & Forgot Password

1.1. Log in

Access the domain of the platform, and log in with the default assigned username and password.

1.2. Forgot Password

If you forget the login password, click "**Forgot Password**" to reset it via email.

2 Develop Product

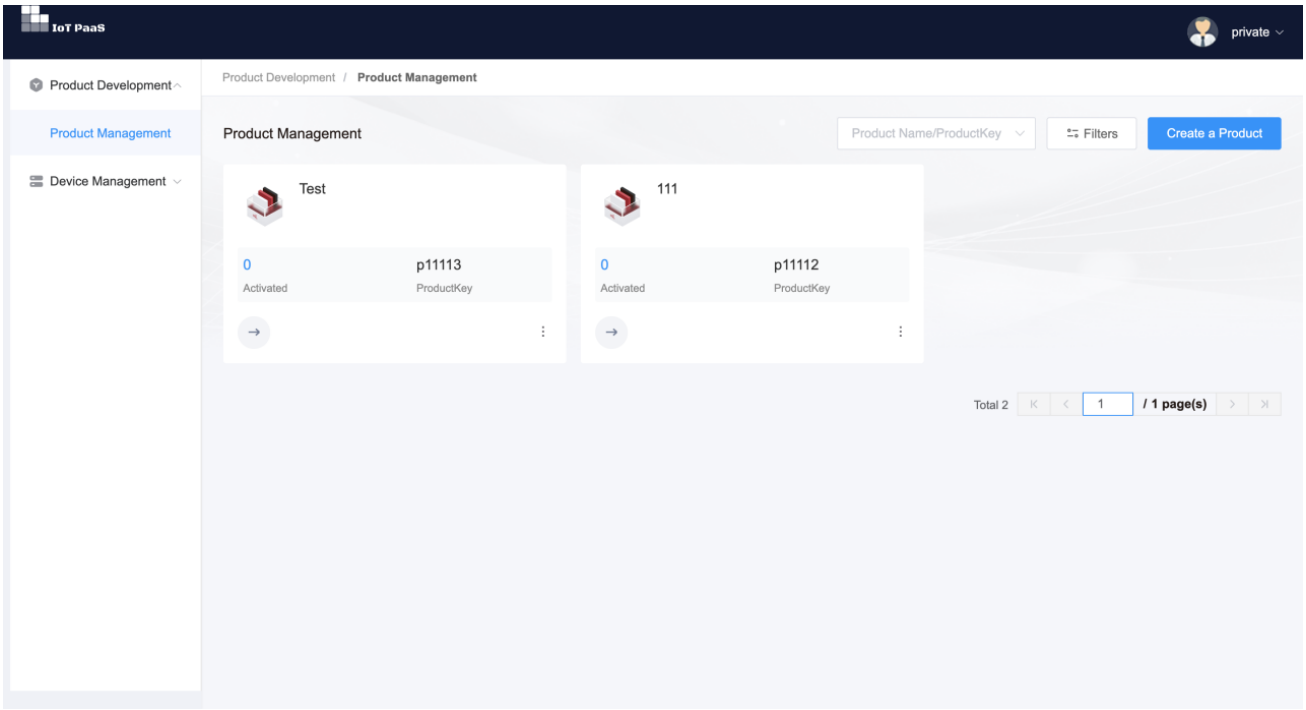
This section outlines the steps involved in preparing for product development, as well as for products creation and maintenance.

2.1. Manage Product

A product is a collection of devices that have the same feature definition (TSL model). A unique ProductKey is issued by the platform for each product. All devices under the product will be flashed with the same ProductKey for authentication and communication with the platform.

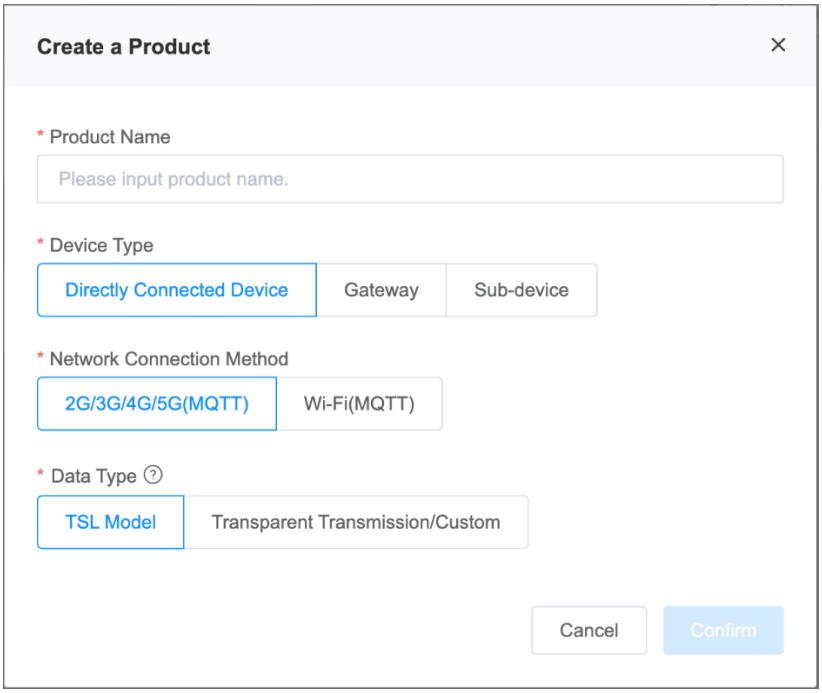
Steps

1. Login to the platform, click and enter "**Product Management**" page.



2. Click "**Create a Product**" to bring up the "Create a Product" window.

3. Select the device type, network connection method and data type of the product.



4. After configuring your product, you need enter a valid product name.

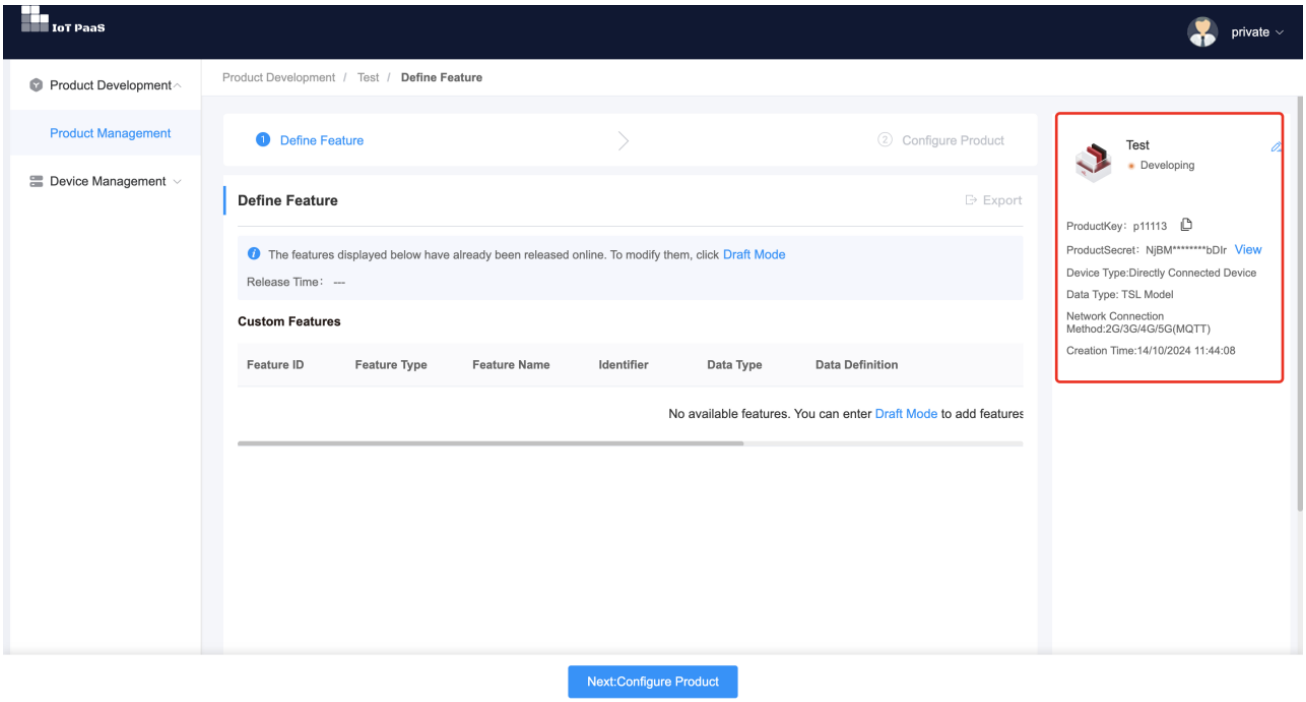
5. Click "**Confirm**" to create the product successfully.

Table 1: Product Creation

Parameter	Description
Device Type	Device types under the product. ● Directly connected device: A hardware device that integrates a communication module and can be directly connected to the platform. ● Gateway: A device that integrates a communication module, mounts sub-devices through different protocols such as 485, Zigbee and RS232, and provides Internet access through a proxy server for sub-devices, enabling them to connect to the platform. A gateway can manage sub-devices and maintain a topological relationship with them. ● Gateway sub-device: A sub-device that does not directly connect to the platform, but instead connects through gateway proxy.
Network Connection Method	Network connection method for directly connected device and gateway. ● Directly connected device: 2G/3G/4G/5G, Wi-Fi and NB-IoT. When creating an NB-IoT product, if using a China Telecom SIM card, you need to set whether to use DTLS encryption for communication. ● Gateway: 2G/3G/4G/5G, Wi-Fi.
Data Format	Uplink and downlink data format. ● TSL (Recommended): A data exchange protocol between the device and the platform provided by the platform. It is in JSON format.

	● Transparent Transmission/Custom: If you want to use a custom data format, you can select "**Transparent Transmission/Custom**".
Product Name	Name of the product. A product name under an account is unique.

On the product details page, the column on the right contains the basic information for the product, where you can modify the product name and logo image.



2.2. Define Feature

2.2.1. TSL Model Overview

- What is a TSL Model

A Thing Specification Language (TSL) model digitally represents physical entities (such as sensors, power storage devices, buildings or factories) in the IoT platform. It can also be regarded as a set of device feature definition. It is specifically tailored for complex device modelling in real-world scenarios, allowing you to define devices with unique features across different products without interfering with one another. You can customize and extend TSL model definitions by importing and exporting the TSL files.

- Feature Types

You can define the TSL features of a product through properties, services and events, which respectively describe the nature of the product, its capabilities, and the information it can provide.

Table 2: Feature Type

Terms	Description
Property	Properties are device capabilities that can be read and set. They are used to describe the

	running status of a device, such as the temperature, humidity, power, voltage and coordinate. Properties support the GET and SET request methods. Applications can send requests to retrieve and set properties.
Event	Events that are actively reported to the platform by running devices. Events are classified into messages, alerts and faults. They can be customized to fit specific application scenarios. You can subscribe to events.
Service	Device capabilities or methods that can be called by applications. You can set their input and output parameters. Services can be called by using a command to implement complex business logic.

- Supported Data Types

You can use these supported data types to define product TSL features on the platform.

Table 3: Supported Data Types

Data Type	Description	Example
BOOL	Boolean. Boolean values include 0 (false) or 1 (true).	True: On; False: Off.
INT	32-bit integer.	10
FLOAT.	Single-precision floating-point type. It has a precision of seven significant decimal digits.	1.1
DOUBLE.	Double-precision floating point. It has a precision of fifteen significant decimal digits.	1.23
ENUM	Enumeration type. ENUM is an enumeration data type that defines a set of named values and descriptions, each represented by an integer.	0 indicates cold wind, 1 indicates hot air, 2 indicates natural wind.
TEXT	Text. String. The data cannot exceed 10,240 bytes in length.	"Hello,world"
RAW	Transparent transmission data type. The content to be sent is encoded in hexadecimal. The data cannot exceed 4,096 bytes in length.	0x0016
DATE	Timestamp. A UTC timestamp in string format. Unit: milliseconds.	"1635839462000"
STRUCT	Structure. Complex encapsulation. When you define a JSON structure, parameters in the structure support BOOL, INT, FLOAT, DOUBLE, ENUM, TEXT, RAW and DATE data types. Structure nesting is not supported.	{ "name":"Tom","age": 20 }
ARRAY	Array. You must specify the element type and number of elements in an array. Array elements support INT ,FLOAT, DOUBLE, TEXT, RAW and STRUCT data types. Make sure that the array elements are the same type. The number of elements must be 1 to 100.	[1, 2, 3, 4, 5, 6]

Table 4: TSL File Definition

Parameter	Description
id	Feature ID
code	The identifier of the feature, which corresponds to the feature ID. Applications use it to exchange the key fields with the device. The identifiers of properties, events, and services in each TSL model are unique.
name	The name of the feature, which corresponds to the feature ID. It is used to display feature.
type	Fixed value feature type. ■ Property：PROPERTY ■ Event：EVENT ■ Service：SERVICE
subType	Subtype of feature type. When feature type=PROPERTY, subType is Read/Write. ■ R: read only ■ W: write only ■ RW: read and write When feature type=EVENT, subType is event. ■ INFO: message ■ WARN: alert ■ ERROR: fault When feature type=SERVICE, subType is call. ■ ASYNC: asynchronous call
sort	The number of a feature. It is used to sort the features.
desc	Feature description, which helps users to clearly understand the meaning and function of a feature.
dataType	Data type INT, FLOAT, DOUBLE, ENUM, TEXT, DATE, STRUCT, ARRAY, BOOL. See Supported Data Types.
specscode	It is used to limit the structure of a data type when feature type is property. When dataType=INT/FLOAT/DOUBLE, specs contains the following fields: ■ step: step length. ■ min: minimum value. ■ max: maximum value. ■ unit: unit. Its value can be empty. When dataType=ENUM, specs contains the following fields: ■ name: Name ■ value: Value

	When dataType=TEXT, specs contains the following fields: ■ length: Length
inputData	Input parameter. This field is available only when feature type=SERVICE. inputData can contain references to properties or newly created parameters.
outputData	Output parameter. When feature type=SERVICE or EVENT, outputData can contain references to properties or newly created parameters.

2.2.2. Manage Feature

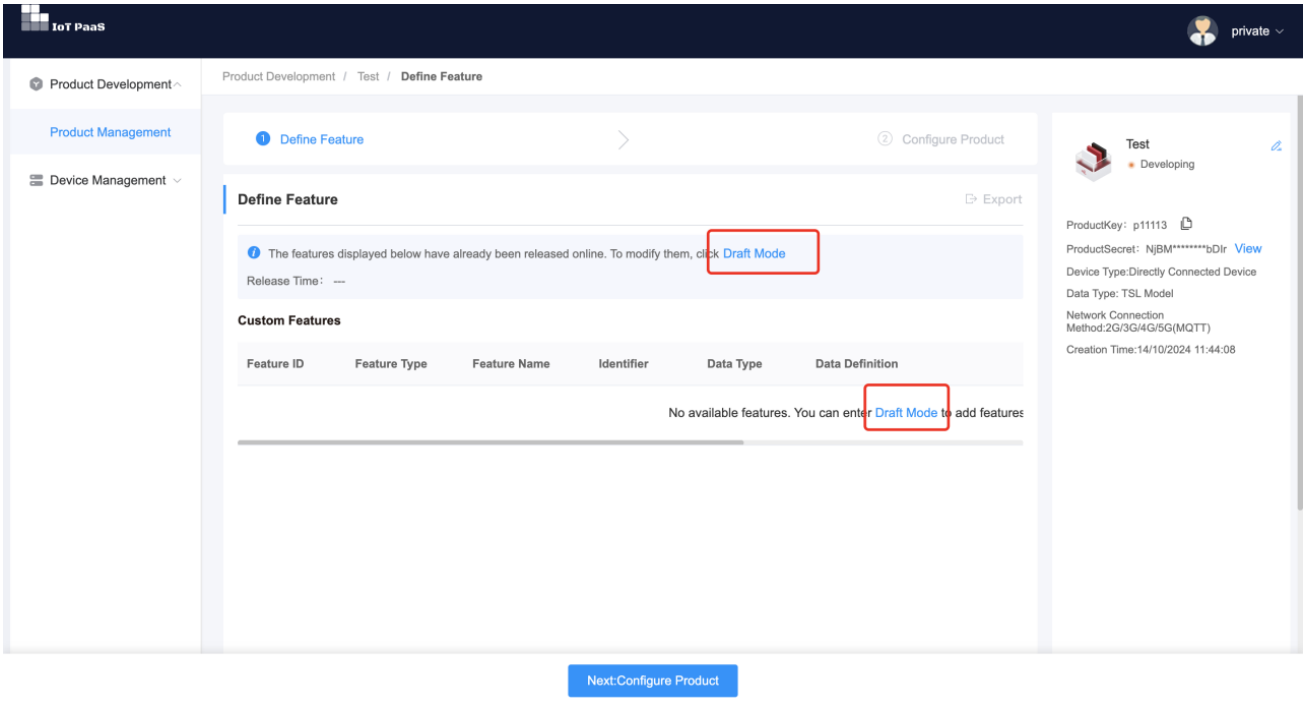
2.2.2.1. Add Feature

Prerequisites

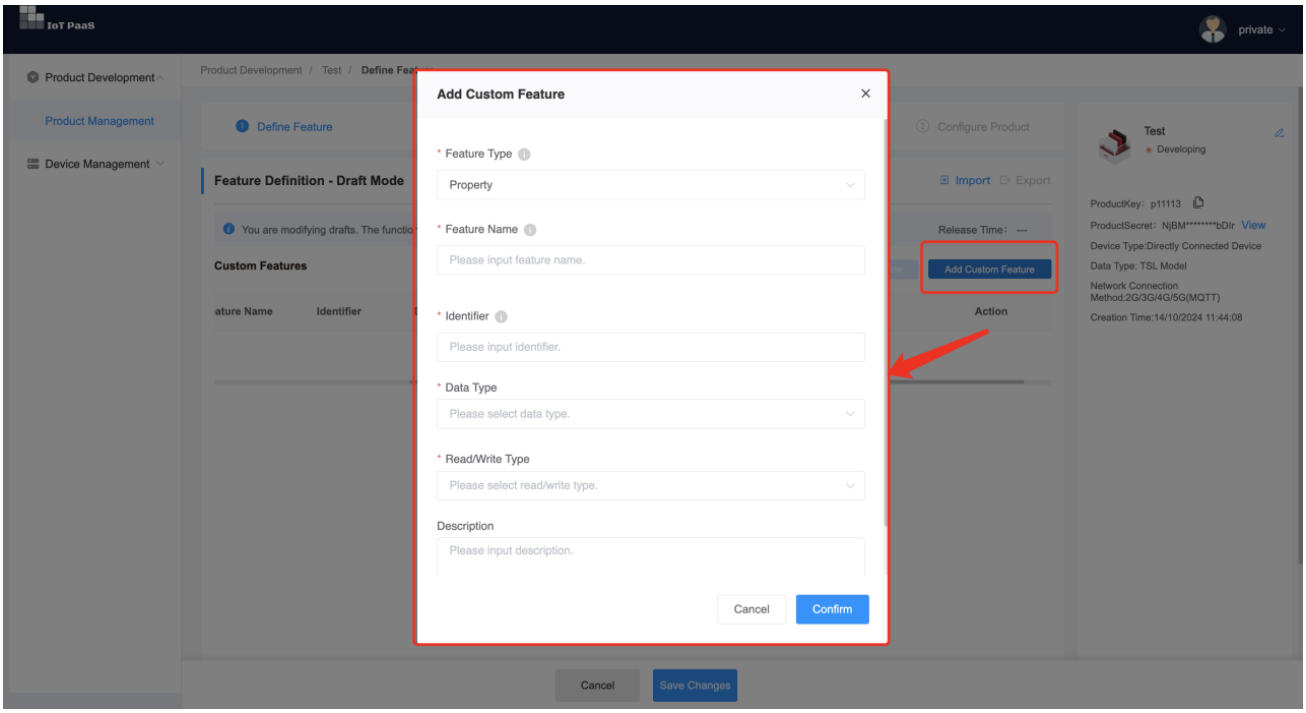
A product has been created in TSL data format.

Steps

1. Log in to the platform, click the product card to enter the "**Product Management**" page.
2. On "Define Feature" page, click "**Draft Mode**" to enter the draft editing page.



3. Click "**Add Custom Feature**" to bring up the following window.
   You can define properties, events and services for the product under development.



2.2.2.1.1. Add a Custom Property

Select "**Property**" in the feature type drop-down box and enter the required parameters, then click "**Confirm**" to add a property.

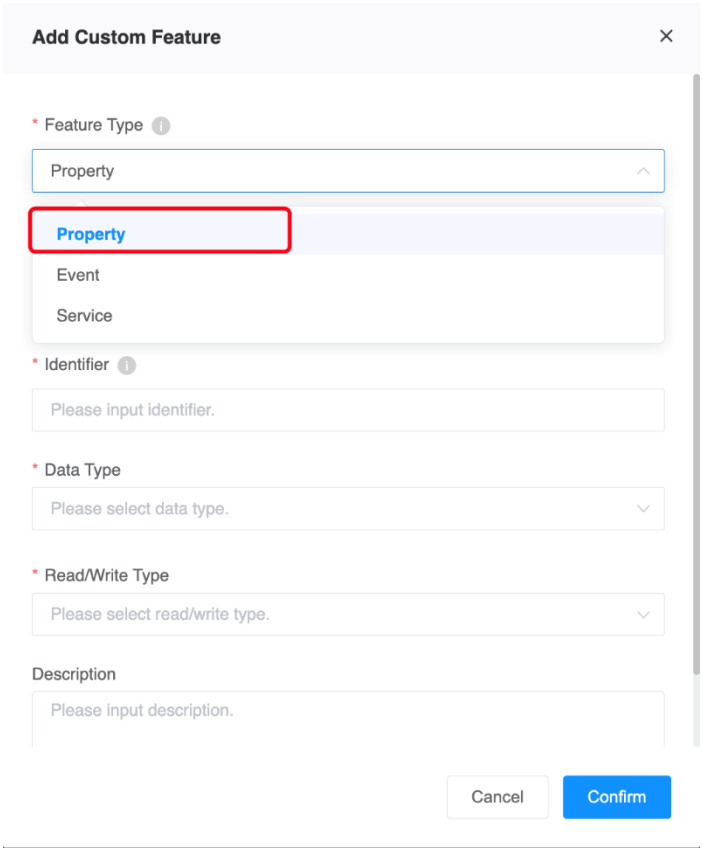


Table 5: Property Parameters

Property	Description
Feature Name	Name of the property. such as power consumption. The feature name must be unique under a product. Chinese characters, uppercase and lowercase letters, numbers, hyphens, underscores, slashes, and periods are supported. It must start with a Chinese character, English letter, or number, and cannot exceed 30 characters.
Identifier	The identifier of the feature, which corresponds to the feature ID. Application uses it to exchange the key fields with the device. The identifiers of properties, events, and services in each TSL model are unique.
Data Type	Data type INT, FLOAT, DOUBLE, ENUM, TEXT, RAW, DATE, STRUCT, ARRAY, BOOL.
Value Range	If the data type is numeric (INT/FLOAT/DOUBLE), the value range can be limited.
Step Length	If the data type is numeric (INT/FLOAT/DOUBLE), you can set the minimum granularity of property variations.
Unit	If the data type is numeric (INT/FLOAT/DOUBLE), you can select the corresponding unit according to your business scenarios.
Data Length	If the data type is TEXT, you must limit the length of the text. Maximum text length cannot exceed 10,240 bytes.
Maximum Number of Elements	When the data type is ARRAY, you must set the maximum length of the array.
Element Type	When the data type is ARRAY, you must specify the element type in the array to make sure that every array element is of the same type.
Read/Write Feature Subtype	R: read only W: write only RW: read and write
Description	Description or remarks on the feature. Maximum length cannot exceed 200 characters.

2.2.2.1.2. Add a Custom Event

Select "**Event**" in the feature type drop-down box and enter the required parameters, then click "**Confirm**" to add a custom event.

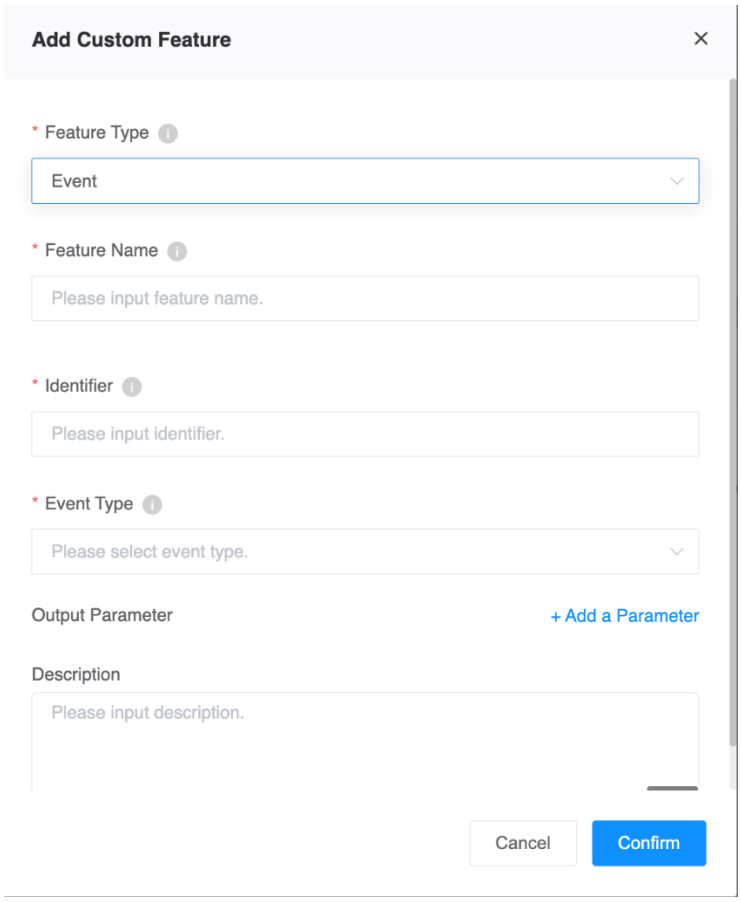


Table 6: Event Parameters

Property	Description
Feature Name	Name of the event, such as over-voltage alert. The feature name must be unique under a product. Chinese characters, uppercase and lowercase letters, numbers, hyphens, underscores, slashes, and periods are supported. It must start with a Chinese character, English letter, or number, and cannot exceed 30 characters.
Identifier	The identifier of the feature, which corresponds to the feature ID. Application uses it to exchange the key fields with the device. The identifiers of properties, events, and services in each TSL model are unique.
Event Type	● Message: a general notification reported by a device, such as the completion of a task. ● Alert: an emergency or exception that a running device actively reports to the platform. The alert information has a high priority and requires immediate attention. ● Fault: an emergency or exception that a running device actively reports to the platform. The fault information has a high priority and requires immediate attention.
Output Parameter	The output parameter of the event. Click **"Add a Parameter"** to add an event output parameter in the pop-up window. You can use a property as a parameter or customize a parameter. For example, if a fault event is defined and the voltage property is selected as the output parameter, the voltage value of the device will be carried in the fault event reported by the device. This information can be used to determine the cause of the fault

	and take appropriate actions.
Description	Description or remarks on the feature. Maximum length cannot exceed 200 characters.

2.2.2.1.3. Add a Custom Service

Select "**Service**" in the feature type drop-down box and enter the required parameters, then click "**Confirm**" to add a custom service.

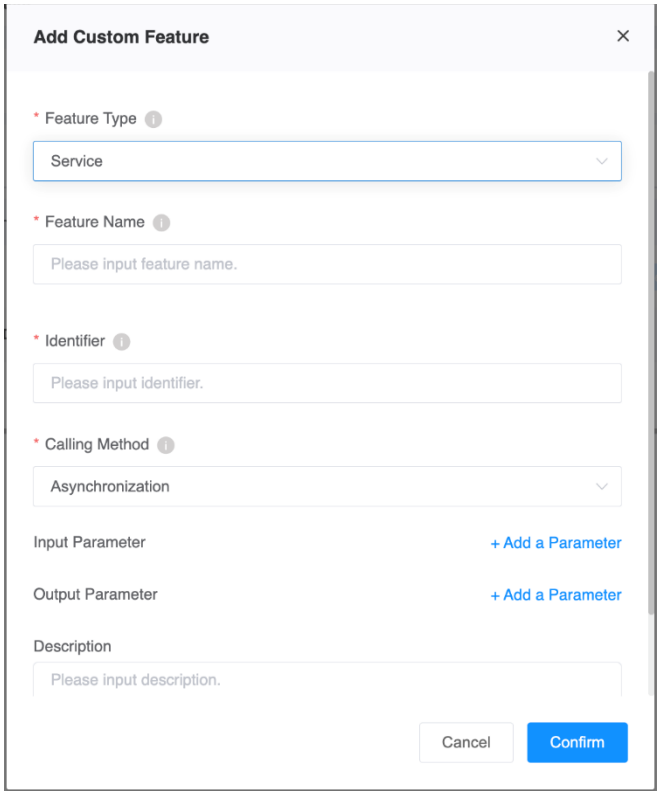


Table 7: Service Parameters

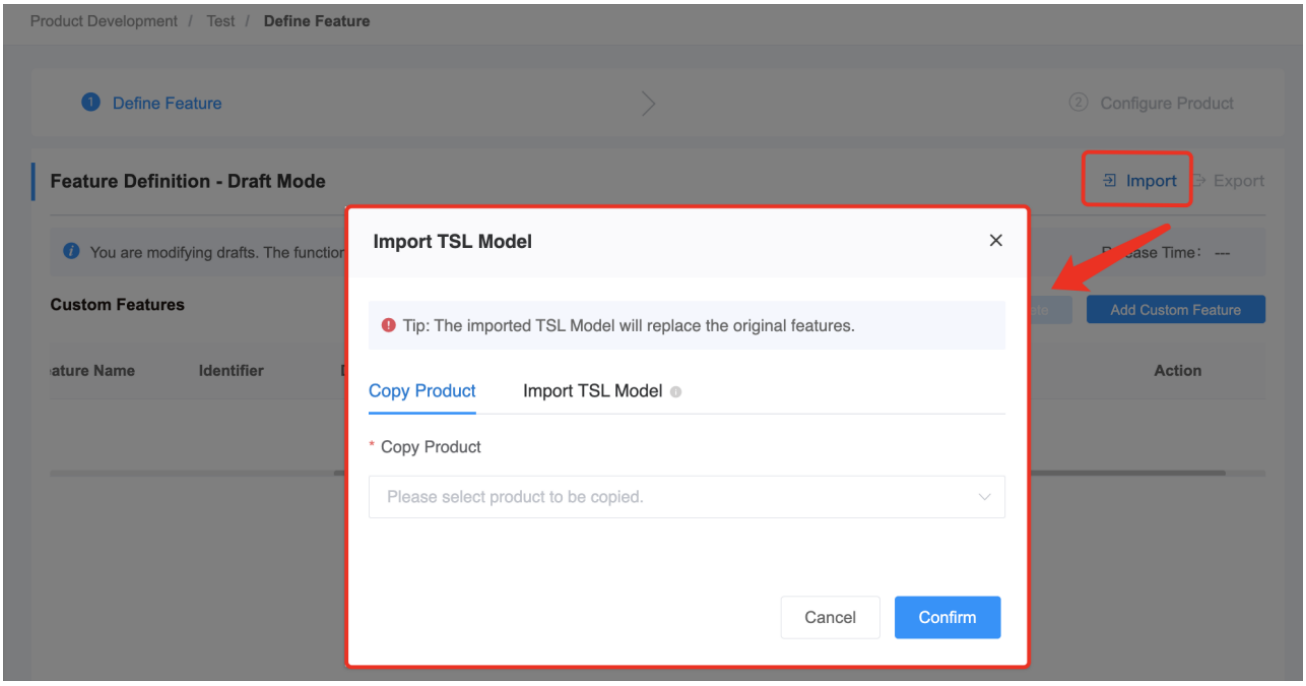
Property	Description
Feature Name	Name of the service, such as open/close gate. The feature name must be unique under a product. Chinese characters, uppercase and lowercase letters, numbers, hyphens, underscores, slashes, and periods are supported. It must start with a Chinese character, English letter, or number, and cannot exceed 30 characters.
Identifier	The identifier of the feature, which corresponds to the feature ID. Application uses it to exchange the key fields with the device. The identifiers of properties, events, and services in each TSL model are unique.
Call Method	● ASYNC: The service is called asynchronously, and the platform returns the results directly without waiting for a response from the device.

	● SYNC: The service is called synchronously, and the platform waits for a response from the device. If the device does not respond within a specified time, the service call times out. (currently not supported)
Input Parameter	The input parameter of the service, which is optional. Click "**Add a Parameter**" to add the service input parameter in the pop-up window. You can use a property as an input parameter or customize a parameter.
Output Parameter	The output parameter of the service, which is optional. Click "**Add a Parameter**" to add the service output parameter in the pop-up window. You can use a property as a parameter or customize a parameter.
Description	Description or remarks on the feature. Maximum length cannot exceed 200 characters.

2.2.2.2. Add TSL Features in Batches

Steps

In "Draft Mode" page, click "**Import**" to bring up the following window.



There are two methods to add TSL features in batches:

1） On the "Copy Product" tab, select a source product and version of TSL. Click "**Confirm**". The custom TSL model corresponding to the source product is imported. You can go to the"Define Feature" tab, and click "**Draft Mode**". On the subsequent page, find the required TSL feature and click "**Edit**" to modify it.

2） On the "Import TSL" tab, upload a JSON file or an Excel file.

3 Manage Device

3.1. Device Maintenance

3.1.1. Device List

Log in to the platform, click "**Device Management**" → "**Device Maintenance**" in the left-hand navigation bar to display the "Device List".

In the "Device List", you can view information about devices that have been connected to the IoT platform, and update the name and SN of local devices. The update does not affect data in the pre-import list.

Table 8: Device List Actions

Parameter	Description
View	You can view device information. See Device Information for details.
Delete	You can delete it and connect it to the platform again. After the device is deleted, the platform retains the historical data generated by the device.
Sub-device Management	This feature only applies to gateway products. You can view, control, and manage the sub-devices under the current gateway device on the "Sub-device Management" tab.
Export Device	You can export local device data according to different products.
Reset DS/ Certificate Fingerprint	If you want to reset DeviceSecret or X.509 certificate fingerprints after the device is connected to the platform, you can click "**Reset**" on the "Device Information" tab to enter the authentication process.

3.1.2. Pre-import Management

Log in to the platform, click "**Device Management**" → "**Device Maintenance**" in the left-hand navigation bar to enter the "Pre-import Management" page.

On the "Pre-import Management" page, you can add devices with different authentication types. Authentication will be performed as selected in pre-import when a device is authenticated. Upon successful verification, the device will be displayed in the "Device List", and the device name and SN configured

during the pre-import will be followed.

Modifying device information in the "Pre-import Management" does not affect the data of devices that are already online in the "Device List".

Add a Device

- **Dynamic authentication**: Devices are authenticated and connected to the platform through PK/PS. When product-level dynamic authentication is disabled, only imported devices can use ProductKey and ProductSecret to connect to the platform for authentication and obtain DeviceSecret.

- Static authentication: Only devices with DeviceSecret flashed in advance can be connected to the platform. the platform will reject the authentication requested by the device whose DeviceKey and DeviceSecret do not match.

  After successfully adding a device with static authentication, you can obtain the DeviceSecret on the "Device Details" or "Manage in Batches" page.

  Note: The device with static authentication can only be added to products using MQTT and LwM2M network protocols.

Table 9: Pre-import Management for Other Device Actions

Parameter	Description
Manage in Batches	Used to obtain device information (including DS) in batches. Users can obtain information about devices connected to the platform through static authentication by checking "Batch No", or obtain device information by using the download code through calling OpenAPI.
Export Devices	You can export the pre-imported device data according to different products.

3.1.3. Device Information

Log in to the platform, click "**Device Management**"→ "**Device Maintenance**" in the left-hand navigation bar. Click "**View"** in the "Action" column to navigate to the "Device Information" page.

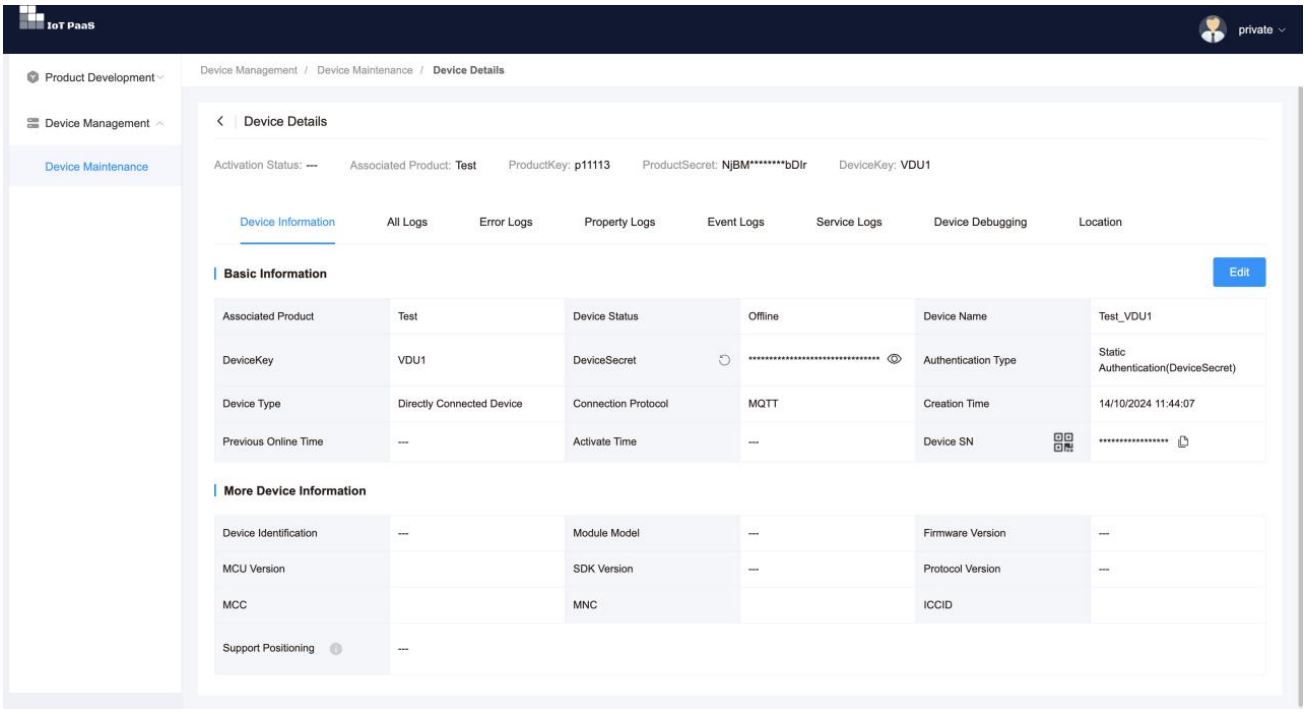


Table 10: Basic Information

Parameter	Description
Associated Product	Name of the product to which the device is associated.
Device Status	Online/Offline.
Device Name	The default value is the product name plus the last four digits of the DeviceKey(DK). It can be edited.
DeviceKey	DeviceKey is the customized unique identifier for devices under a product used for connecting to the platform.
DeviceSecret	Device key. Device key is issued by the platform after the device is certificated.
Authentication Type	Authentication type when the device accesses the platform.
Certificate Fingerprint	Device certificate identifier. After a successful bidirectional authentication between the device and the platform, the certificate fingerprint will be bound to the device. Subsequent connections to the platform can only be made by using this device certificate. To calculate the device certificate fingerprint, use SHA-1 hashing, such as openssl x509 -fingerprint -sha1 -in certfile.crt. The fingerprint generated by OpenSSL includes colons ":" and must be converted to a 40-bit hexadecimal string without the colons before uploading. You can reset the fingerprint and set a new value. Once the new fingerprint is set successfully, the device can use it to authenticate with the platform.
Device Type	Directly Connected Device, Gateway, Gateway sub-device.
Connection Protocol	Protocol used by the device to connect to the platform. MQTT and LwM2M are currently supported.
Creation Time	The first time when the device is added to the platform.
Previous Online Time	The last time the device was online.
Device SN	ID of the device bound to App. It can be edited.

More Device Information includes module information, and the details are as follows:

Table 11: More Device Information

Parameter	Description
Device ID	Module’s IMEI or MAC address.
Module Type	Module model reported by the device.

Firmware Version	Module's firmware version number reported by the device.
MCU Version	MCU'S firmware version number reported by the device.
ICCID	ICCID number of the SIM card used by the cellular module.
MCC	Mobile country code.
MNC	Mobile network code.
Positioning Support	Module's supported positioning feature reported by the device.
Protocol Version	Data protocol version between the module and the platform.
SDK Version	QuecThing version.

3.1.4. Data Logs

Log in to the platform, click "**Device Management**"→"**Device Maintenance**" in the left-hand navigation bar to display the list of devices. Click "**View**" in the "Action" column to enter the "Device Information" tab. Then click the "**Data Logs**" tab.

Table 12: Data Logs Parameters

Parameter	Description
Ticket	Unique data identifier.
Creation Time	The time when the data was stored.
Data Type	Uplink Data: The commands or messages that the device reports to the platform. Downlink Data: The commands or messages that the platform sends to the device through API or the platform's webpage.
Sending Status	Pending: When the platform sends downlink data to the device through device debugging or API, and the device is offline with a cache duration set, the status of the data will be shown as "Pending". This indicates that the data will be delivered to the device if it goes online, or sends uplink data within the cache duration. Note that the pending status only applies to downlink data. ● Sent: The "Sent" status for downlink data indicates that the downlink data sent from the platform through device debugging or API has been successfully delivered to the device and a response has been received by the platform within 5 seconds. Similarly, the "Sent" status for uplink data indicates that the data sent from the device has been successfully delivered to the platform. ● Failed: The "Failed" status for uplink data means that the device has connected to the gateway successfully, but no subsequent action has been performed, resulting in abnormal server or network. Similarly, the "Failed" status for downlink

	data indicates that the downlink data sent from the platform through device debugging or API was not delivered to the device, or a response was not received by the platform within 5 seconds. This is caused by the disconnection between the module and the platform, with the device disconnected, resulting in a pseudo-connection.
Data Content	Click "**View**" to view the data in various data formats. HexString: Data or byte array is converted to hexadecimal characters; (for example, binary 0x1234AB (3 bytes) is converted to string "1234AB" (6 bytes); String "1234AB" (6 bytes) is converted to binary 0x1234AB (3 bytes)). Base64: Binary data based on 64 printable characters (A-Z in upper case, a-z in lower case, +, /). Text: String in UTF-8 encoding format. Json: The format in which a TSL model is transmitted and presented.
Sending Time	The "Transmission Time" of downlink data indicates the date when the data was sent from the platform to the device. The "Transmission Time" of uplink data indicates the date when the platform received the data sent from the device.

3.1.5. Exception Logs

Log in to the platform, click "**Device Management**" → "**Device Maintenance**" in the left-hand navigation bar to display the list of devices. Click "**View**" in the "Action" column to navigate to "Device Information" page, and then click "**Exception Logs**".

Table 13: Exception Logs Parameters

Parameter	Description
Ticket	Unique data identifier.
Creation Time	The time when the data was stored.
Data Type	Uplink Data: The commands or messages that the device reports to the platform.
Data Content	Display the specific ID of the uplink data with failed analysis. Exception data will not be sent to WebSocket or the AMQP client.

3.1.6. Property Logs

Log in to the platform, click "**Device Management**" → "**Device Maintenance**" in the left-hand navigation bar to display the device list. Click "**View**" in the "Action" column to enter "Device Information" page, and then click "**Property Logs**".

Table 14: Property Logs Parameters

Parameter	Description
Feature ID	Feature ID defined in the TSL model.
Property Name	Name of the property defined in the TSL model.
Data Type	Data type of the feature defined in the TSL model.
Identifier	Identifier of the feature defined in TSL model.
Update Time	The most recent time when the property was reported.
Current Value	Value of the property reported by the device at the last time.
Actions	View Data: Historical reported values can be queried by specified properties. View Curve: For the properties in INT/FLOAT/DOUBLE type, historical reported values can be viewed in a curve format.

3.1.7. Event Logs

Log in to the platform, click "**Device Management** "→ "**Device Maintenance**" in the left-hand navigation bar to display the list of devices. Click "**View**" in the "Action" column to navigate to "Device Information" page, and then click "**Event Logs**".

Table 15: Event Logs Parameters

Parameter	Description
DeviceKey	Device identifier.
Creation Time	The time when the event log is stored.
Event Type	● Device Online: An event is recorded each time the device connects to the platform. ● Device Offline: An event is recorded if the device actively disconnects from the platform, or if it does not communicate with the platform (including going online, sending uplink data and downlink data) for an extended period of time, in which case the platform forces the device to log off. ● Back Online: An event is recorded when the device registers and accesses the platform. It disconnects from the platform and then connects again to gets back online. This occurs when a sub-device is connected to different gateways that are dynamically switched. ● Device Reset: An event is recorded when the device actively restores the factory settings. A new key for device connection is issued, and the topology relationship between the gateway and the sub-device and the binding relationship between the end user and the device are released.

	● Message: Only the messages defined in TSL model are recorded in this event. ● Alert: Only the alerts defined in TSL model are recorded in this event. ● Fault: Only the faults defined in TSL model are recorded in this event.
Output Parameter	Specific output parameters are only carried in TSL events.
Remarks	Event details are displayed, such as reasons for disconnection. Invalid events will not be pushed to AMQP or WebSocket.

3.1.8. Service Logs

Log in to the platform, click "**Device Management**"→ "**Device Maintenance**" in the left-hand navigation bar to display the list of devices. Click "**View**" in the "Action" column to navigate to "Device Information" page and then click "**Service Logs**".

Table 16: Service Logs Parameters

Parameter	Description
Feature ID	Unique identifier of a product feature.
Service Name	Name of the service defined in the TSL model.
Identifier	Identifier of the service defined in TSL model, which corresponds to the service name.
Sending Time	The time when a service was called.
Input Parameter	The input parameter when a service is called.
Output Parameter	The output parameter in the device response.

3.1.9. Device Debugging

Log in to the platform, click "**Device Management**" → "**Device Maintenance**" in the left-hand navigation bar to display the list of devices. Click "**View**" in the "Action" column to navigate to " Device Information" page. Then click "**Device Debugging**" tab to access the debugging feature.

Device debugging is sending commands or data from the platform's webpage to the device to verify the interaction logic between the device and platform. Depending on the product data formats, device debugging can be divided into two types: transparent transmission data debugging and TSL data debugging.

- **Transparent Transmission Data Debugging**

Transparent transmission data debugging involves sending and receiving custom packet data between the device and application, which are interconnected through a custom protocol. In this type of debugging, the platform only serves as a transmitter of the data, without parsing the transparent transmission data or expanding data applications. The device and application parse the packet data by themselves.

The left is the input area for data debugging, and the right is the display area for real-time data logs.

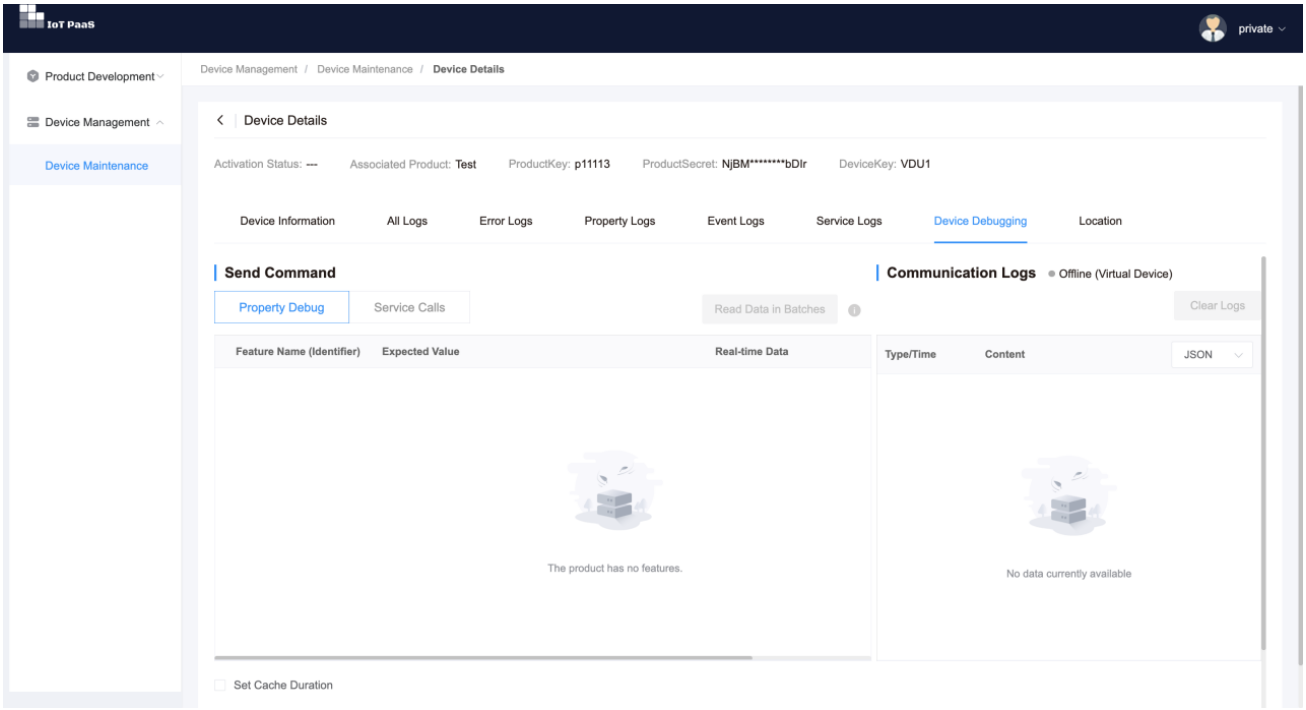


Table 17: Transparent Transmission Data Debugging Parameters

Parameter	Description
Data Format	Data format when sending transparent transmission data. Hex: Data or byte array is converted to hexadecimal characters. (For example, binary 0x1234AB (3 bytes) is converted to string "1234AB" (6 bytes); string "1234AB" (6 bytes) is converted to binary 0x1234AB (3 bytes)); Text: The data is sent in UTF-8 encoding format.
Data Content	Maximum length of data is 4,096 characters, and the size of the data content is determined by the length of the data that the module can receive.
Cache Duration	Cache the data to be sent. If the device that is currently being debugged is offline, the content in the cache will be delivered the next time the device goes online or sends uplink data. Unit: second
isCover	• Yes: indicates that when the current device is offline, if repeated commands are sent, only the latest command will be sent the next time the device goes online or sends uplink data, and the previous command will be overwritten by the latest command. • No: indicates that when the device is offline, if repeated commands are sent,

	all cached data will be delivered when the device goes online or sends uplink data.
Send Command	Click "**Send Command**" to send data, and the data will be displayed in the real-time log.
Reset	Clear all current settings and data content, and return to the initial status of "Device Debugging" page.

3.1.10. Device Positioning

Log in to the platform, click "**Device Management**" → "**Device Maintenance**" in the left-hand navigation bar to display the list of devices. Click "**View**" in the "Action" column to navigate to "Device Information" page, and then click "**Location**" tab to view location-related data for the device.

Table 18: Device Positioning Parameters

Parameter	Description
Positioning Time	The most recent time when the positioning data was reported.
Number of Visible Satellites	When the positioning data is GPS, the number of satellites currently in view for positioning may be reported.
Positioning State	● Differential Positioning: A satellite signal receiver (Satellite Ground Station) is at a fixed position, enabling it to calculate satellite positioning error. The handheld receiver uses the satellite signals to calculate coordinates, which are then refined by synthesizing the positioning error data provided by the ground station, resulting in improved accuracy. ● Non-differential Positioning: A single receiver is used for positioning to get the absolute coordinates of the receiver antenna. ● Valid Positioning: Accurate information such as latitude and longitude have been obtained. ● Invalid Positioning: Invalid latitude and longitude, speed and direction information has been obtained. ● Estimating: indicates an approximate estimate of the current location information and positioning state.
Positioning Method	● GNSS: Global Navigation Satellite System ● GPS: Global Positioning System (United States) ● GL: GLONASS (Russia) ● GA: Galileo (European Union) ● BD: BeiDou Navigation Satellite System (China)
Longitude and Latitude	The latitude and longitude reported based on WGS-84 coordinate system.

4 Personal Center

4.1. Personal Account

4.1.1. My Profile

You can edit your personal data such as username, enterprise name, telephone number and time zone on "My Profile" page.

4.1.2. Password Change

You can change your password at any time if you think that your current password is not secure.

4.1.3. Security Settings

You can enable email verification for secondary login confirmation.

4.2. Sub-account Management

You can create a sub-account and assign data permissions for products and SaaS applications.

Table 19: Sub-account Management Parameters

Parameter	Description
Username	Username for the sub-account.
Password	Initial password for the sub-account.
Confirm password	Re-enter the password to avoid incorrect input.
Email	Email address for the sub-account. Subsequent operations of the sub-account will be notified by email. Make sure to enter a valid email address.
Authorized Product	Authorized products include All Products and Specified Products. ● **Permissions for All Products**: the sub-account has data permissions for all products owned by the enterprise user, including future products.

- **Permissions for Specified Products**: If you have selected a specified product, the sub-account has the data permissions for the selected product. The data permissions for products include the data permissions for the products created by the sub-account itself, regardless of whether All Products or Specified Products is selected. If no product is selected, the sub-account only has data permissions for the products it has created.

If you want to delete a sub-account, click "**Sub-account Management**" tab and find the sub-account you want to delete in the sub-account list, and then click "**Delete**" in the Action column to delete it. All resources created by the sub-account are retained after the sub-account is deleted, and belong to the enterprise user.

4.3. Logs

You can view the operation logs of key data on the "Logs" page, such as product creation/deletion, SaaS application creation/deletion.

Table 20: Logs Parameters

Parameter	Description
ID	A unique identifier for each event log.
Type	Event type: user, product, device, application, system, OTA upgrade, and message subscription.
Actions	Operations performed: addition, modification, deletion, import, export, registration, upload, login, download, and others.
Time	Time at which the event was triggered.
Username	User or system that produced the event.
Description	Description of the log. You can click "**More**" to view a specific log.
Remarks	You can add your own remarks to differentiate logs.

4.4. Custom Configuration

4.4.1. Style Configuration

You can customize the platform name, platform logo, website icon, login page logo, login page background, default product image, theme color, copyright information, and other details.

4.4.2. Email Configuration

You can customize the email account used by the platform to send emails by filling in the account name, SMTP server address and port, SMTP account and password, and choosing whether to enable SSL and TLS encryption.

Except for email accounts, you can edit the email templates for ProductKey and ProductSecret notifications, user password reset requests, message queue deactivation, and message queue expiration in the system as needed.