Sarl 1.0 Acola
E-mail : brc@bisatel.com
61 rue de Lyon, 75012 Paris

Standard : 09 67 40 91 57
WhatsApp : +330667207166
Siret : 749 812 558 / TVA : FR23749812558

BISATEL TELECOM

DOCUMENTATION TECHNIQUE PLATEFORME ADMINISTATION OPERATEUR MOBILE ESIM/ SIM V 1.0 24 mars 2026

MENU

Roles on the platform

1. Organization

An Organization is the level at which contracts with operators are signed and SIM cards are issued. At this level, you can include one or multiple Companies, and it is responsible for managing all the Companies within its structure.

On this level, an Organization can:

- Add SIM cards
- Holds contracts with operators
- Adds its payment systems (Stripe, PayPal)
- Integrate SIM cards with operators or simply store information about them
- Create appropriate plans, although SIM cards do not necessarily have to be integrated with operators

An Organization can also add SIM cards without integrations. This means Organizations can use our platform as a place to store all their SIM cards.

Example:

If an Organization has 1,000 SIM cards, but the operator does not have integration with our platform or the integration has not been implemented yet, the Organization can still:

- View all their SIM cards
- Assign them to their clients
- Track which clients are using these SIM cards and which Company they belong to

2. Company

- Belongs to an Organization
- Created for business clients or virtual Companies
- Can be used to sell SIM cards to end consumers
- End Users are associated with the Company
- Can switch to the level: End User

A Company **always belongs to an Organization**. We cannot create a Company without an Organization. This means that a Company is a business client that belongs to an Organization.

Each Company can have its own settings, plans, and users. Even if the Organization does not have business clients (companies) and wants to sell SIM cards directly to end consumers, it still needs to create a virtual Company that will not be visible to customers but will represent, for example, a point of sale.

Example:

An Organization has a prepaid product and wants to sell it through Amazon. To do this, the Organization needs to create a Company and name it, for instance, “Amazon Band.” Thus:

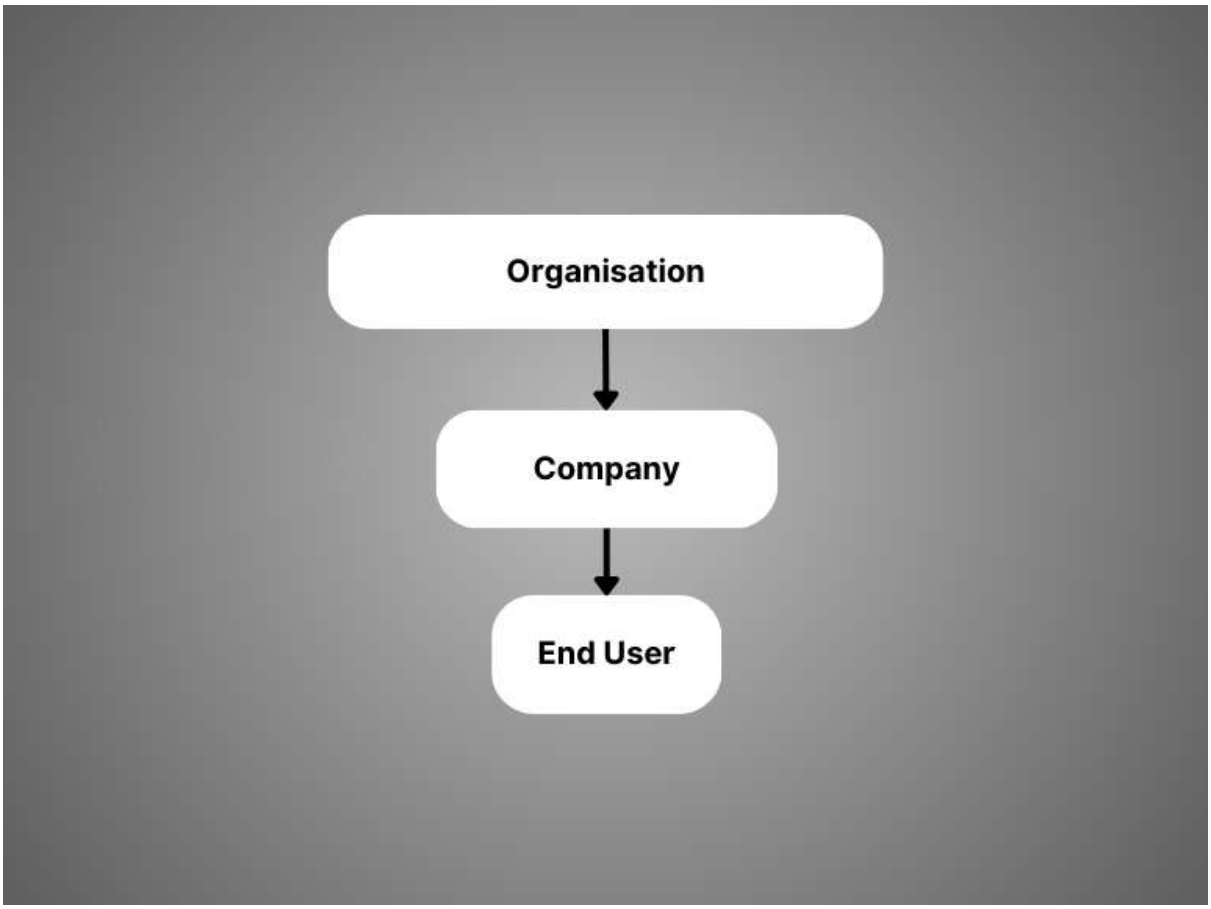
- The SIM cards sold on Amazon will be linked to this Company.
- End consumers who purchase these SIM cards will register on our platform and automatically be associated with the created Company “Amazon Band.”

3. End User

- Has login credentials (username, password, can set up 2FA)
- Is associated with a Company, which in turn is associated with an Organization
- End Users cannot configure or use rules. SIM card management rules can only be configured at the Organization or Company level
- End Users are linked to a Company, and the Company is linked to an Organization.

This structure is immutable. We cannot link an end user directly to an Organization. The End User is only linked to a Company, which in turn is linked to an Organization.

Sarl 1.0 Acola
E-mail : brc@bisatel.com
61 rue de Lyon, 75012 Paris
Standard : 09 67 40 91 57
WhatsApp : +330667207166
Siret : 749 812 558 / TVA : FR23749812558



Important

The platform has a clear hierarchy: there is an admin, which branches into Organizations, each Organization can see its Companies, and each Company can have End Users. This structure does not change!

Example:

An End User cannot be linked directly to an Organization. The end user is linked to a Company, and the Company is linked to an Organization.

- Organizations can see their companies
- Companies can have end users

SIMs

The SIMs page provides convenient access to essential information about each SIM card, including their statuses, SIM types, operators, associated companies, and end users.

This enables to easily track data usage, and remaining days on active plans, and view detailed information for each card, such as ICCID, operator, plan, company, and other relevant details.

Dictionary

SIM Types

1. Prepaid

- A SIM card with a certain balance. When the balance runs out, you can refill it yourself or through auto-refills, and the SIM card will continue to work.
- The client pays for the balance top-up, and the amount is added to the SIM card, making it usable.

- SIM cards have usage terms defined by operator contracts. For example, if there are no refills for three years, the SIM card may be deactivated. This is just an example, as details are always specified in contracts with operators.

2. Subscription

A subscription is a product type where the SIM card is used for a month, and at the end of the month, an invoice is issued for all expenses according to the plan (usage). Payment for the usage is deducted at the end of the month.

Types of Subscription:

- Static Pool
- Dynamic Pool
- Unlimited
- Monthly Bundle
- Pay per Megabyte

3. PAYGO

Pay GO (pay-as-you-go in real time) is a product that charges for data per megabyte.

PAYGO is a flexible data billing product that charges users per megabyte of data used, offering advanced features and seamless integration. Unlike standard pay-per-megabyte models, PAYGO allows operators to set prices not only based on total data consumption but also for specific countries.

SIM Statuses

- **Not available** – No integration.
- **Pending** – Being checked by the integrator because it was just added to the platform, or operations are being performed on it (e.g., SIM swap).
- **Blank** – SIM without a line.
- **Activation ready** – Has a line and a plan (linked to a company if not for store use).
- **Active** – Operating in normal mode.
- **Suspended** – Manually or systemically suspended (e.g., expired).
- **Invalid** – Something went wrong with the integration

General terms in the SIMs section

- **ICCID** - The unique identifier for each SIM card. For eSIMs, the number is displayed along with a QR code, making it easy to distinguish between physical SIMs and eSIMs.
- **In session**:

  Yes - the SIM is currently online and in use.
  No - the SIM is currently offline and not in use.
  N/A - the status of the SIM is undetermined because the integration does not support it.

- **Operator** - The mobile network provider associated with the SIM, such as Telefonica, Vodafone UK, HOT Mobile, Odido, Orange France, Partner, Plus, WWS.
- **Plan** - Shows the name of the current plan, which is automatically generated but can be customized if needed
- **Company** - Company name to which the SIM card is assigned.
- **End User** - The email address of the end user linked to the SIM.
- **Data left** (MB) - Remaining data allowance for the SIM.
- **Days left** - The remaining days before the plan or SIM expires.
- **This month** (MB) - Data usage for the current month.
- **Note** - A field for any additional notes or comments.
- **Rules** - Customizable rules applied to the SIM
- **Labels** - Customizable labels for organizing SIMs.
- **SIM type** - Indicates whether the SIM is Prepaid, Subscription, or PAYGO.

Sarl 1.0 Acola
E-mail : brc@bisatel.com
61 rue de Lyon, 75012 Paris
Standard : 09 67 40 91 57
WhatsApp : +330667207166
Siret : 749 812 558 / TVA : FR23749812558

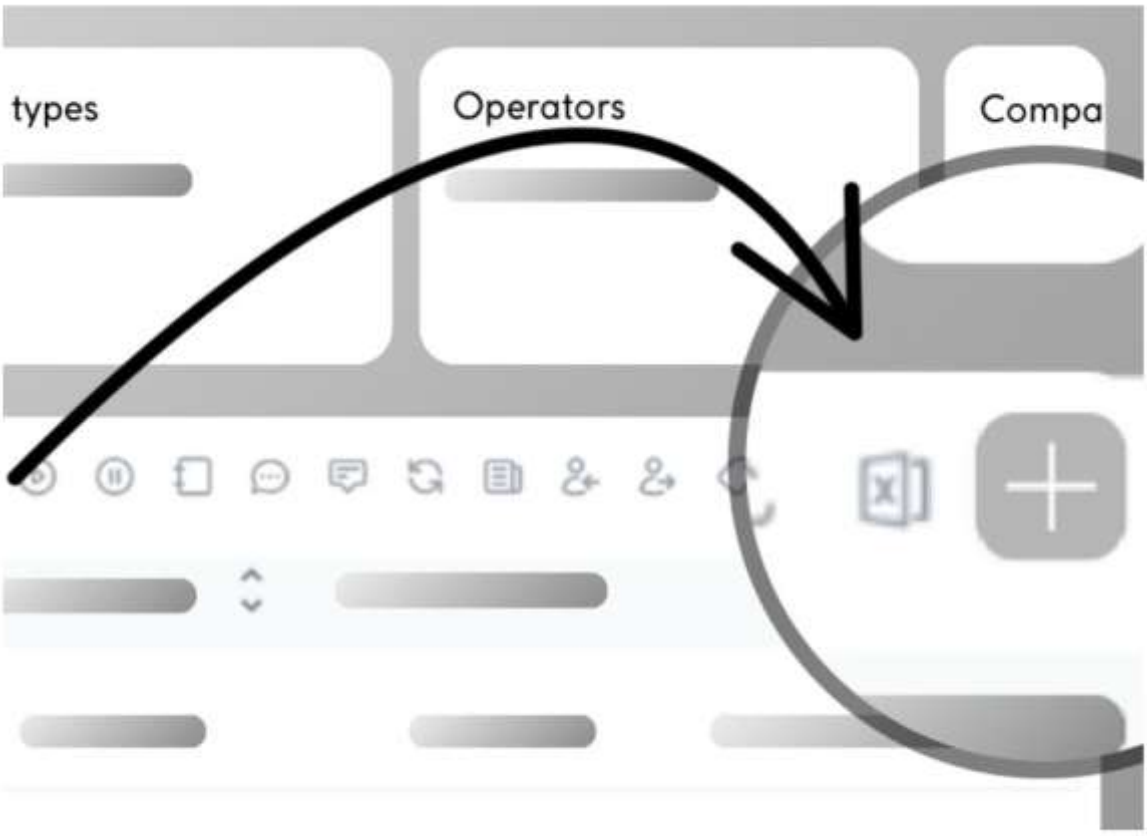
Key features

Key features of the page include:

1. **Dynamic Table:**

You can adjust the order of displayed data, add new columns from the existing list, or hide columns that are not needed. This flexibility allows you to customize the interface to fit your specific needs.

2. **By clicking the "+" button, you can perform 5 actions:**



- Upload new SIM card(s) into the system
- Create New Prepaid SIM card(s)
- Create New Subscription SIM card(s)
- Create New PAYGO SIM card(s)
- Prepare an eSIM(s) for the store

3. **Bulk Operations** - You can perform bulk actions such as:

- Activate: Quickly activate multiple SIM cards at once.

- Suspend: Suspend a group of SIM cards, temporarily halting their services.
- Replace Plan: Update the data plan for several SIMs in one go.
- Send SMS: Send a message to the end users associated with the selected SIM cards.
- Change Note: Modify or update notes for multiple SIM cards to include new information or instructions.
- Refresh Usage: Refresh the usage data for all selected SIMs to ensure up-to-date statistics.
- Rules: Apply or modify automation rules for the selected SIMs, such as data limits or notifications.
- Assign Company: Assign multiple SIM cards to a company for better organization.
- Unassign Company: Remove the company association from selected SIM cards.
- Labels: Add or modify labels to categorize and organize multiple SIM cards at once.
- Cancel SIM: Cancel the selected SIM cards and remove them from the system.
- Upload Details: Upload specific details or configurations in bulk for selected SIM cards.
- Export: Export the information of selected SIM cards to a file for further analysis or reporting.

4. **Detailed SIM Card Information:** Here you can click on any SIM card number (ICCID) to access more detailed information, organized into three main tabs:

- Details: Manage the SIM card status, view data left, add labels for bulk management, assign the SIM to an end user, set up specific rules, send SMS messages, and more.
- Usage: Check daily data usage and see information about the countries where the SIM is being used.
- Log: Displays a history of changes with date, modified fields, old and new values, and the user who made the changes. This is useful for auditing and tracking adjustments.

It helps track the status and usage of SIM cards, ensuring full control over mobile connections. This is crucial for maintaining seamless operations, avoiding overages on mobile expenses, and responding promptly to any issues or changes.

Q&A

1. **What does the Activation Ready status mean?**

This status means that the SIM is active with the operator, but there has been no usage yet. Once the SIM card is used, the status will change to "Active."

2. **Why can't I add SIM Card to the eSIM store**?

You are unable to add a SIM card to the eSIM store due to the following reasons:

- The necessary integration for the SIM is missing.
- The SIM card is not an eSIM. Only eSIMs can be added to the eSIM store, so please ensure you are adding an eSIM and not a regular SIM card.

3. **How do I refill?**

You have two options for refilling your balance:

1. **Manual Refill**

You can refill your balance manually through your account page on the Platform. Simply log in to your account and follow the prompts to add funds.

2. **Auto-Refill**

Alternatively, you can set up an auto-refill option. This allows for automatic balance top-ups when your data falls below a specified threshold. To use this feature, you will need to provide a credit card on your account page.

Regardless of the method you choose, your data will be added immediately in real-time upon successful completion of the refill.

Pools

On the Pools page, you can manage data pools. You can also add preset configurations to ensure consistent data pool management.

Dictionary

1) Static Pool

A static pool is a pool with a fixed price, independent of the amount of SIM cards in the pool.

Functionality:

You can assign multiple SIM cards to the pool without impacting the overall cost, making it useful for situations where usage needs to be controlled while maintaining a predictable budget.

2) Dynamic Pool

Dynamic pool - every new sim added to the pool adds to the total amount of data and increases the pool price. Also known as "SIM cards with pooled data between them".

Key features

Key features of the page include:

- **Dynamic Table**: You can customize the table by adding existing or hiding columns according to your needs. This adaptability ensures that you can view the most relevant information for your management tasks.

- **Bulk Operations**: You can perform bulk operations to manage multiple pools efficiently.

- **Detailed Information**: By clicking on a specific row, you can access more detailed information organized into several tabs:

- Details: View detailed metrics such as data usage, available data, allowance, pre-bill information, assigned plans, rules, and labels. This tab provides an in-depth look at how the pool is utilized and managed.

- General: Includes basic information about the pool, such as its name, type, currency, and associated company.

- Billing: Provides details on overage size and overage price, helping to manage and anticipate additional costs.

- Usage: Shows monthly and daily data usage statistics, allowing for detailed tracking and analysis of data consumption.

- Log: Displays a history of changes with date, modified fields, old and new values, and the user who made the changes. This is useful for auditing and tracking adjustments.

The **Pools** page helps optimize data allocation, control costs, and ensure efficient use of mobile data resources.

Plans

On the Plans page, you can create and customize data bundles, assign SIM cards to Plans, and monitor Plan details. The table presents essential information about each Plan, including its name, type, associated operator, company, details, labels, and whether the Plan is prorated. You can also add preset configurations for quick and consistent Plan setup.

Dictionary

Plan types:

1. Prepaid

A Prepaid Plan for a SIM card comes with a set balance, and once that balance is depleted, the client can refill it manually or through auto-refills.

How It Works:

- Balance Tracking: the balance is monitored in real-time. When the balance reaches zero, the SIM card is deactivated.
- Refilling: The organization can configure/create refills.

Usage Terms:

Prepaid SIM cards have specific terms based on the operator's contract. For example, if a SIM card remains inactive (no refills) for three years, it may be deactivated. These details vary and are specified in the operator's contract.

Additional Prepaid Options:

Auto-Refill: This feature automatically tops up the balance when it falls below a certain threshold (e.g., below 100 MB). The system charges either the company's card or the end user's card to the organization's account, and upon successful payment, the balance is updated instantly.

Wallet Balance: This option allows the client to store a larger sum in advance for auto-refills. For instance, instead of topping up multiple SIM cards frequently with small amounts, the client can deposit $1,000 to cover auto-refills as needed.

Product Expiry:

Each Prepaid product comes with a usage term. For example, 1 GB of data may be valid for 365 days. If the balance is not refilled within that period, any remaining balance will expire. However, with every refill, the validity period extends from the refill date, ensuring the product can be used for a longer time as long as the balance is regularly topped up.

2. Subscription

A subscription is a Plan type where the SIM card is used for a month, and at the end of the month, an invoice is issued for all expenses according to the Plan (usage). Payment for the usage is deducted at the end of the month.

The subscription is designed for companies rather than end consumers. The end consumer may be linked to the SIM card and see the usage, but cannot manage the SIM card. Management and payment are handled by the company responsible for the SIM card.

Types of Subscription Plan:
1) Static Pool
A static pool is a pool with a fixed price, independent of the amount of SIM cards in the pool.

2) Dynamic Pool
Dynamic pool - every new sim added to the pool adds to the total amount of data and increases the pool price. Also known as "SIM cards with pooled data between them

3) Unlimited
This is a subscription with no usage limits. After paying a fixed cost you can have access to unlimited usage without additional fees.

4) Monthly Bundle
This bundle is controlled by only one SIM card. You will pay for a fixed amount of data per month. If the entire amount is not used, the funds are not refunded.
If usage exceeds the bundle limit, additional charges for the extra data apply.

5) Pay per Megabyte
In this product, the user pays for each megabyte of data used. The more megabytes the user consumes, the more they pay.

3. PAYGO

PAYGO is a real-time product that charges users for data on a per-megabyte basis. While it operates similarly to pay-per-megabyte systems, it provides advanced features and supports seamless integrations. Operators can set prices not only for total data usage but also specifically for different countries. If the operator supplies country-specific information or imposes restrictions in certain regions, corresponding pricing can be established.

Product Classification

PAYGO is classified as a Prepaid service, which means it operates based on a balance. Users must maintain a positive balance to utilize the service; it cannot be accessed if the balance is zero.

PAYGO plans work exclusively with the Wallet feature, so if you don't have any companies with Wallet enabled, the company list will be empty when creating a PAYGO plan.

How It Works

1. **Data Collection**

The system gathers information from the SIM card, including:

- The amount of megabytes used
- The country in which the card was utilized

2. **Pricing Determination**

The system identifies the applicable price based on the collected data and the specific country.

3. **Balance Deduction**

The necessary amount is instantly deducted from the user's balance.

Real-Time Processing

As new information is received, the system performs calculations and deducts funds, ensuring that the balance decreases continuously in real time.

Key features

Key features of the page include:

Dynamic Table: The table is fully customizable, enabling you to add new columns or hide existing ones based on your preferences. This flexibility allows for a more personalized and efficient management experience.

Bulk Operations: You can perform bulk actions to streamline plan management, such as applying changes to multiple SIM cards simultaneously.

Detailed SIM Card Information: By clicking on a SIM card number (ICCID), you can access detailed information organized into two main tabs:

Details: View and manage the SIMs assigned to each plan, track refill history, manage labels, and see lifetime refill statistics. This tab provides a comprehensive view of how each SIM card is using its data plan.

Log: Displays a history of changes with date, modified fields, old and new values, and the user who made the changes. This is useful for auditing and tracking adjustments.

This page meets the needs of businesses and organizations that require efficient control over their mobile data plans. By allowing customization and real-time monitoring of data plans, it ensures that mobile data resources are allocated and used effectively.

The ability to track changes and manage multiple SIM cards simultaneously helps to maintain oversight, prevent data overages, and optimize costs, making it an essential tool for managing mobile connectivity.

Q&A

1. **Why can't I select a company when creating a PAYGO plan?**

The company list includes only companies that have Wallet enabled. PAYGO plans work exclusively with the Wallet feature, so if you don't have any companies with Wallet enabled, the company list will be empty when creating a PAYGO plan.

Companies

The Companies page is designed to meet the administrative needs of managing multiple Companies within the platform.

Dictionary

General Terms from Companies page:

- Name - The name of the Company.
- End Users - Number of end users associated with the Company.
- Prepaid - Number of prepaid SIM cards.
- Subscription - Number of subscription SIM cards.
- PAYGO - Number of PAYGO SIM cards.
- Staff - Total number of staff members.
- Pool - Number of pools associated with the Company.
- Wallet -The balance of the Company's wallet.
- Labels - Tags or categories for easy identification.

Key features

Detailed Information (Clicking on a row): Provides additional details about the selected Company:

- General: Overview of prepaid SIMs, subscription SIMs, PAYGO SIMs, end user information, and financial data, including the amount paid this month and lifetime payments.
- Wallet: Detailed wallet information, including the starting balance, total refilled amount, total spent, and the ending balance. A table with transaction data is also provided, showing the date, description, and amount of each transaction.
- Staff: List of staff members associated with the Company, including first name, last name, email, and type.
- Company Details: Information about the Company name, owner, email, notification settings, and currency preferences.

The **Companies** page provides a clear and organized view of each Company's SIM card usage, financial transactions, and associated staff, helping administrators maintain control and oversight.

End Users

Here you as an Organization can manage information about End Users, including their associated Companies, SIM cards, and financial details. This page is intended to streamline user management by offering detailed insights into End User accounts, SIM usage, and payment settings.

Dictionary

General Terms from End Users page:

- Name - For end users, the name is shown if available; otherwise, the email is displayed when the name is not provided.
- Company - The company associated with the end user.

- Prepaid - Number of prepaid SIM cards assigned to the end user.
- Subscription - Number of subscription SIM cards assigned.
- PAYGO - Number of PAYGO SIM cards assigned.
- Wallet - Current wallet balance of the end user.
- Labels - Tags or categories for easy identification and filtering.

Key features

Detailed Information (Clicking on a row): Provides further details about the selected End User:

- General: Overview of SIM card assignments (prepaid, subscription, PAYGO), wallet information including paid amounts (monthly and lifetime), and general settings such as notification preferences and whether the End User pays for new purchases.
- Permissions: Defines the actions the End User can perform, such as adding, deleting, editing, or viewing information. This includes managing payment methods, staff-related actions, and SIM card management (e.g., updating notes, auto-refill options, refilling).
- Log: Displays a history of changes with date, modified fields, old and new values, and the user who made the changes. This is useful for auditing and tracking adjustments.

The **End Users** page allows for efficient management of SIM card assignments, tracking of wallet balances, and configuration of End User permissions and settings.

Automation

The Automation Rules page allows you to create and manage rules that automate various processes. This functionality helps streamline operations, improve efficiency, and ensure that specific actions are triggered automatically based on predefined criteria.

Dictionary

General Terms from the Automation page:

- Name - The identifier for the rule.
- Type - The type of entity the rule applies to (e.g., SIM, Pool, Account).
- Period - The timeframe or cycle during which the rule is active (e.g., billing cycle, permanent, refill cycle).
- Trigger - The event or condition that activates the rule (e.g., status change, specific date, remaining balance).
- Plan Type - The type of plan to which the rule applies.
- Condition - The specific criteria that must be met for the rule to trigger.
- Action - The action that will be taken when the rule is triggered (e.g., suspend, activate, send email notification).

- Created By - The initiator of the rule.

Key features

1. **Detailed Information:** Clicking on a specific row provides additional details:

- SIMs: Displays the number of SIMs affected by the rule.
- General: Provides an overview including the rule's name, period, type, plan type, and description.
- Condition: Details the trigger, condition, value, and specific action to be taken.
- Action: Specifies what actions will be performed when the rule is triggered, including email and push notifications. Includes options for email notifications (on/off), push API notifications (on/off), and additional email settings for end users and companies.

2. **Rule Logs**

- Tracks the history of rule executions with:
- Date: When the action was triggered.
- Destination: The recipient of the notification or action.
- Trigger: The condition or event that activated the rule.
- Rule: The name or identifier of the rule. This field is available only when the transaction was made manually.
- Action: The action is taken when the rule is triggered.

Rules can help to automate repetitive tasks and ensure timely responses based on specific conditions. By setting up automation rules, you can efficiently manage operations, reduce manual effort, and ensure that actions and notifications are handled consistently and automatically.

This helps in optimizing workflow, maintaining operational efficiency, and ensuring that critical processes are executed without manual intervention.

Transactions

The Transactions page allows efficient tracking and manages financial transactions, including payments and refills, across different billing cycles and payment methods.

Dictionary

General Terms from Transactions page:

- Date - The date the transaction occurred.

- Type - The type of transaction ((e)SIM refill, refill wallet).
- Destination - The recipient or purpose of the transaction.
- Amount - The total amount involved in the transaction.
- Payment Method - The method used for payment (e.g., Credit Card via Stripe, iDEAL, Manual).
- Payer - The entity responsible for making the payment.
- Billing Cycle - The billing period associated with the transaction.
- Labels - Custom labels that can be used for categorization.

Key features

Detailed Information: Clicking on a specific row provides additional details, including:

- Labels: Option to add new labels for better categorization.
- Company Info: Details about the company related to the transaction.
- Payer Info: Information about the payer, including their name and email.
- Transaction Details:

- Type: The type of transaction.
- Destination: The recipient or purpose.
- Amount: The amount of the transaction.
- Payment Method: The method used.
- Transaction ID: A unique identifier for the transaction.
- Reason: The reason for the transaction.

Refill Amount (MB): For transactions related to data refills, the amount of data refilled.

By offering comprehensive transaction details and the ability to categorize and manage labels, the page supports efficient financial oversight and helps prevent errors or discrepancies in billing and payment processing.

SETTINGS

The Settings page provides access to essential configuration options for managing your account and platform functionality.

Account Settings

The Account Settings page serves as a comprehensive hub for managing essential information related to your account. This section is organized into four key areas: Billing Info, Address Info, Contact Info, and Default Settings.

1. Billing Info

In the Billing Info section, users can enter and update their company's financial details. This includes:

- **Company Name:** The official name of your organization as it appears on billing statements and legal documents.
- **VAT Number:** A unique identification number assigned to your business for value-added tax purposes, necessary for invoicing and tax compliance.

This information is crucial for ensuring accurate billing and compliance with financial regulations.

2. Address Info

The Address Info section allows users to provide their organization's physical address for billing and communication purposes. The fields include:

- Postal Code: The postal code for your business location, ensuring accurate mail delivery.
- Country: The country where your organization is located.
- Street: The street address of your business.
- House Number / Apartment: Specific house or apartment number to pinpoint your location.
- City: The city where your organization operates.
- State: The state or region of your business address, if applicable.

Having accurate address information is essential for effective communication and service delivery.

3. Contact Info

The Contact Info section is designed to maintain up-to-date contact information for the primary point of contact within your organization. This includes:

- Contact Name: The name of the person responsible for managing account-related inquiries.

- Contact Email: The email address for the primary contact, ensuring prompt communication regarding account matters.
- Phone Number: The primary phone number for reaching the contact person.
- Phone Number 2: An optional secondary phone number for additional contact methods.

This section ensures that communication is streamlined and that support can be provided efficiently.

4. Default Settings

The Default section allows users to set preferences for their account. This includes:

- Currency: The default currency is set so that you don't have to select it repeatedly when creating a plan.
- Email for Notifications: The email address designated to receive notifications related to account activity, such as billing alerts, service updates, and other important communications.

Company Staff

The Staff page allows administrators to register team members and manage their access rights on the platform. This functionality ensures that each staff member has the appropriate permissions to perform their tasks while maintaining security and control over sensitive information.

1. Automation (Allow / Restrict)

This section governs the automation rules for staff. Administrators can set permissions for various actions related to automation:

Rule: Control access to automation rules by allowing or restricting staff from:

- Add: Create new automation rules.
- View: Access existing rules.
- Edit: Modify existing rules.
- Delete: Remove rules from the system.

2. Company (Allow / Restrict)

This section focuses on permissions related to company management:

- Payment Method: Manage payment methods for the company by allowing or restricting actions to:

- Add: Introduce new payment methods.
- View: Access current payment methods.
- Delete: Remove existing payment methods.

- Staff: Control access to staff management features:

- Add: Create new staff profiles.
- View: Access staff information.
- Edit: Modify staff profiles.
- Delete: Remove staff members.
- Log: View action logs of staff activities.
- Permissions: Manage individual permissions for staff.
- Reset: Reset staff passwords.

- Webhook: Set permissions for webhook management:

- View: Access existing webhooks.
- Update: Modify webhook settings.

3. eSIM Store (Allow / Restrict)

In this section, administrators can control access to the eSIM store:

- View: Allow or restrict staff from viewing eSIM offerings.
- Update: Enable or disable staff from updating eSIM inventory or pricing.

4. End-User (Allow / Restrict)

This section manages permissions related to end-user management:

- Add: Create new end-user profiles.
- View: Access end-user details.
- Edit: Modify existing end-user profiles.
- Delete: Remove end-users from the platform.
- Permissions: Manage permissions for end-users.
- Payment Method: Control access to end-user payment methods:
- Add: Create new payment methods.
- View: Access current payment methods.
- Delete: Remove payment methods.
- Staff: Manage staff assigned to specific end-users with similar permission controls as listed above.

5. SIM Card (Allow / Restrict)

This section governs the management of SIM cards:

- Add: Create new SIM card entries.
- View: Access details of existing SIM cards.
- Edit: Modify SIM card information.
- Delete: Remove SIM cards from the system.
- Actions: Control various SIM card actions, including:

- Assign/Unassign Company: Manage company assignments for SIM cards.
- Assign/Unassign End-User: Manage end-user assignments.
- Assign/Unassign Plan: Link SIM cards to specific plans.
- Assign/Unassign Rule: Apply automation rules to SIM cards.
- Assign/Unassign Label: Tag SIM cards for better organization.
- Cancel Location: Manage location settings for SIM cards.
- Change Status: Modify the status of SIM cards.
- Send SMS: Control SMS functionality for SIM cards.
- SIM Swap: Manage SIM swapping procedures.
- Update Note: Add notes to SIM card records.
- Manual Refill: Perform manual refills for prepaid SIM cards.

6. Invoice Software (Allow / Restrict)

This section controls access to invoice management features:

- Add: Create new invoice templates.
- View: Access existing invoices.
- Edit: Modify invoice details.
- Delete: Remove invoices from the system.
- Parameters: Set permissions for viewing and editing invoice parameters.

7. Operator (Allow / Restrict)

Manage telecom operator details:

- Add: Create new operator profiles.
- View: Access information about existing operators.
- Edit: Modify operator details.
- Delete: Remove operators from the system.

8. Organization (Allow / Restrict)

This section allows control over organizational settings:

- View: Access organization details.
- Edit: Modify organizational information.
- Staff: Manage staff related to the organization with similar permissions as outlined in previous sections.
- Webhook: Manage webhooks specific to the organization.

9. Payment (Allow / Restrict)

This section controls access to payment management:

- View: Allow or restrict staff from accessing payment details.

10. Billing (Allow / Restrict)

Manage billing settings and permissions:

- View: Access billing information.
- Edit: Modify billing settings.
- Advance Billing Enable: Control whether advance billing options are available.
- Create: Allow or restrict staff from creating billing entries.

11. Plan (Allow / Restrict)

Control access to service plans:

- Add: Create new service plans.
- View: Access details of existing plans.
- Edit: Modify plan information.
- Delete: Remove plans from the system.

12. Pool (Allow / Restrict)

Manage access to resource pools:

- Pool: Control permissions for adding, viewing, editing, and deleting resource pools.

13. PSP (Allow / Restrict)

Manage Payment Service Provider settings:

- Add: Create new PSP entries.
- View: Access details of existing PSPs.
- Edit: Modify PSP information.
- Delete: Remove PSPs from the system.

Operators

The Operators section is dedicated to managing telecom operators associated with the platform.

A centralized management approach simplifies the oversight of multiple operators, ensuring efficient operations and easy adaptation to changing business requirements.

Operator Name	Status
FloLive	Integrated ✅
HOT Mobile	Integrated ✅
KPN	Integrated ✅
Odido	Integrated ✅
Partner	Integrated ✅
Pelephone	Integrated ✅
Plus	Integrated ✅
Rogers	Integrated ✅
Tele2 Sweden	Integrated ✅
Telefonica	Integrated ✅
Vodafone	Integrated ✅
WWS	Integrated ✅

Q&A

How to add new operators?

1. Navigate to Settings.
2. Select "Operators" from the menu.
3. Click the "+" button on the right side of the page to add a new operator.
4. On the "Add a New Operator" page, select an operator from the provided list.
5. Enter any additional details, if required.

6. Click the Create button to finalize.

If the operator you need isn't listed, please reach out to our manager for assistance.

Rules

Rules allow organizations and companies to customize when and how they receive notifications. **End users do not manage these rules.**

Type	Levels	Description
Rules	Organizations Companies	Customizable settings that organizations or companies configure to manage when and how notifications are sent

- Who can set rules: Only organizations and companies.
- Example: "Notify me when the balance drops below 500MB."

These rules provide flexibility, enabling organizations to receive additional notifications based on their specific requirements.

Notifications

Our platform provides a few types of notifications, allowing you to stay informed about key updates and service changes. Here's how they work for different service types and how you can customize them to fit your needs.

Type	Levels	Description
Notifications	Organizations Companies End Users	Automatic alerts are sent to end users or companies. They can be categorized as mandatory and optional
Webhook Notifications	Organizations Companies End Users	Automatically send real-time updates to your system through webhooks. This allows seamless integration with your internal processes.

1. Prepaid Services Notifications

For Prepaid services, end users will receive reminders when their balance is about to expire:

- 10 days remaining: "You have 10 days left, and your balance will expire."
- 5 days remaining: "You have 5 days left, and your balance will expire."

These notifications help end users avoid service interruptions by giving them enough time to top up their accounts.

2. Subscription and PAYGO Services

For subscription and PAYGO (pay-as-you-go) services, you can choose where notifications are sent:

- Set a default email where notifications will always be delivered.
- Send notifications to specific emails within the company.
- Use the end user's email if needed.

This ensures that both internal teams and customers are informed at the right time.

3. Rules for Notifications

Rules allow organizations and companies to customize when and how they receive notifications. **End users do not manage these rules.**

- Who can set rules: Only organizations and companies.
- Example: "Notify me when the balance drops below 500MB."

These rules provide flexibility, enabling organizations to receive additional notifications based on their specific requirements.

Billing Integrations

The Billing Integrations page is designed to manage and monitor financial transactions through integrated payment service providers (PSPs) and invoicing systems.

1. PSPs (Payment Service Providers)

This section provides an overview of transaction data from various payment service providers integrated into the platform. For each PSP, you can view both monthly and total transaction details, presented in EUR and transaction count.

- Stripe:

- Transactions This Month: Amount (EUR) and Count
- Total Transactions: Amount (EUR) and Count

- PayPal:

- Transactions This Month: Amount (EUR) and Count
- Total Transactions: Amount (EUR) and Count

- Cardknox:

- Transactions This Month: Amount (EUR) and Count
- Total Transactions: Amount (EUR) and Count

- Add payment providers: The platform also supports the addition of new payment service providers and the modification of their credentials, providing enhanced flexibility and control over payment integrations.

2. Invoice Integration

This section allows you to add and manage third-party invoicing systems to streamline your billing process. By clicking the "+" button, you can integrate with supported invoicing platforms:

- WeFact: A cloud-based invoicing system.
- Xero: A popular accounting software for small businesses.

Once integrated, these systems allow for automated invoice generation and management, enhancing billing efficiency on the platform.

You as Organization can connect all your products there and on the platform using SKU, and customers using Identifier.

Important

If a Company or End User wants to make a payment, they can only use the payment methods linked by the organization. If the organization has not linked any payment systems, it means that the company and end user cannot make any payments (it all depends on the product type being used).

Communication

On this page, you, as an organization, can set up direct links to make it easier for your customers to connect with you. Here's what you can do:

1. Contact Us

- Add a link to your external contact page.
- Once you set it up, your customers will see a new menu item that takes them directly to this link.
- You can easily replace or remove the link whenever needed.

2. Report an Issue

- Provide a link where your customers can report any issues.
- This could be a Google Form, a page on your website that sends requests to your email, or any other platform you prefer.
- After you set it up, your customers will see a new menu item directing them to this link.
- You're free to change or delete the link at any time.

3. Order New SIM

- Add a link where customers can request a new SIM card.
- Once added, your customers will see a menu item that points them to this link.
- You can update or remove this link whenever you want.

WALLET

The Wallet page comprehensively overviews your company's financial transactions and wallet management.

1. General

- Enable Wallet: This toggle feature allows users to turn the wallet functionality on or off, providing flexibility based on the company's financial management preferences
- Minimum Balance: Can either be positive or negative. A positive minimum balance indicates that a certain amount must be maintained in the account, while a negative minimum balance allows for a deficit, meaning the account can go below zero up to a specified limit.

2. Overview of Transactions

- Starting Balance: Displays the initial amount available in the wallet before any transactions are made, providing a baseline for financial tracking.
- Total Refilled: This shows the total amount added to the wallet over a specified period, helping users monitor funding activity.
- Total Spent: Indicates the cumulative amount spent from the wallet, allowing users to track expenditures efficiently.
- Ending Balance: Reflects the current wallet balance after accounting for all refills and expenditures, ensuring users have an accurate picture of their financial status.

3. Transaction History

- Date: Lists the date of each transaction, providing a timeline for financial activities.
- Description: Offers a brief summary of each transaction, such as refills or payments made, to help users understand the context of each entry.
- Amount: Displays the monetary value of each transaction, allowing users to assess their financial movements at a glance.

Sarl 1.0 Acola
E-mail : brc@bisatel.com
61 rue de Lyon, 75012 Paris
Standard : 09 67 40 91 57
WhatsApp : +330667207166
Siret : 749 812 558 / TVA : FR23749812558

ONBOARDING GUIDE

Mandatory settings

1. Integrate your Telecoms

Steps:

1. Go to Setting
2. Choose Operators
3. “Operator” tab will appear -> press the “plus” button on your right
4. It will reveal the page where you need to choose the operator from the drop-down list
5. After you choose the operator, depending on the integration type, you need to fill in the details =>

- **Option 1** – Operator is not integrated, which means that you can add SIMs and manage them, but won’t be able to have real-time information about data usage.

If integration is needed, you need to contact the operator and provide us with the API documentation and credentials for integration. That can take us up to 2 weeks to integrate

- **Option 2** – there is integration with an operator, which you need to choose from the drop-down list. In the “integration” credentials needed (CERT-file and key, or additional fields like license key and API prefix, or login and password, etc.)

---

2. Upload your SIMs

1. Go to the “SIMs” button on your left list of the Side navigation menu
2. Press “plus” button on your right
3. A wizard will appear
4. Choose the necessary from the list:

4.1 Upload new SIM(s) - (SIMs that have no plan yet)

1. Choose the operator and company this SIM(s) belongs to
2. choose if you want to add SIM ICCID manually through a comma or upload via file

4.2 New Prepaid SIM(s)

1. Choose the company and prepaid plan created for this exact company. You may as well assign it to the end user if needed
2. Choose if you want to add SIM ICCID manually through a comma or upload via file

4.3 New Subscription SIM(s)

1. Choose the company and subscription plan you created for this company
2. Choose if you want to add SIM ICCID manually through a comma or upload via file

4.4 New PAYGO SIM(s)

1. Choose the company and PAYGO plan created for this company
2. Choose if you want to add SIM ICCID manually through a comma or upload via file

⚠ IMPORTANT

Before adding/uploading SIMs, please check with your operator if they are active. Sometimes you need to activate SIM(s) with the operator first.

"Activation ready" status means traffic is monitored on a SIM from the moment you attach a plan to it

---

3. Create Company

Steps:

1. Go to the "Companies" button on your left list of the Side navigation menu
2. On your right press, the "plus" button to create a company
3. A wizard will appear and you will need to fill in all the required information

---

Sarl 1.0 Acola
E-mail : brc@bisatel.com
61 rue de Lyon, 75012 Paris
Standard : 09 67 40 91 57
WhatsApp : +330667207166
Siret : 749 812 558 / TVA : FR23749812558

4. Create Plans

Steps:

1. Go to the “Plans” button on your left list of the Side navigation menu
2. On your right press the “plus” button and choose the plan type you want to create
3. When a new wizard appears
4. Fill in all essential fields.

General plan - all created companies in this organization will be automatically assigned to this plan
Company-specific plan - will be assigned to the selected companies

⚠ IMPORTANT:

The price should be not less than 1.0 when creating a plan

---

5. Create your "Product"

1. Assign Plan and Company to SIM
2. Go to the “SIMs” button on your left list of the Side navigation menu
3. Choose SIM/SIMs from the SIMs list you have
4. Choose from the toolbar necessary action (assign company/plan etc.)

Optional settings

1. Account Settings

For VAT collecting

Sarl 1.0 Acola
E-mail : brc@bisatel.com
61 rue de Lyon, 75012 Paris
Standard : 09 67 40 91 57
WhatsApp : +330667207166
Siret : 749 812 558 / TVA : FR23749812558

Steps:

- Go to Settings
- Choose “Account”

Fill in all information in the form (if you need VAT to be added to the account please include this information as well / Address / Contact info etc.)

2. User Profile

For the security of the account

Steps:

- Go to choose the user profile icon
- Fill in: personal info / Multi-factor authentication/time zone
- If you want to change your password
- Press on the password field
- A wizard will appear where you can change the password
- Follow the instructions

3. PSP Integration

(Payment Service Provider Integration) - For payment processing

Steps:

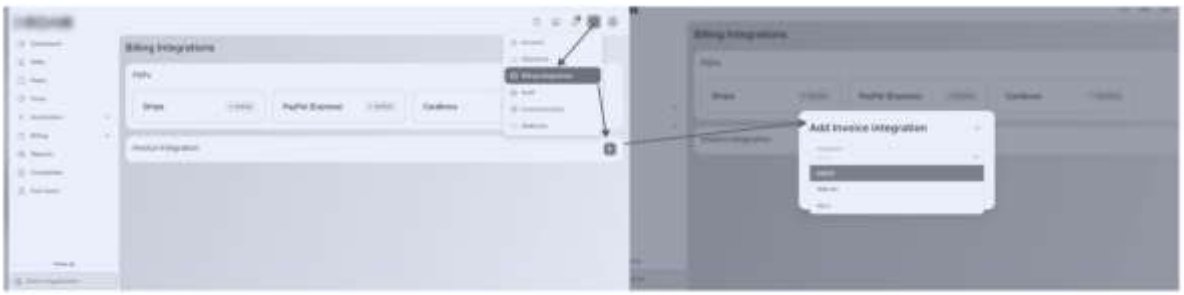
- Go to Settings
- Choose “Billing integrations”
- In the newly opened window choose PSPs you will use for Prepaid & PAYGO.

- Stripe
- PayPal
- Cardknox
- Mollie

If you plan to use only Subscription - PSP is not needed

4. Invoice Software Integration

For Invoicing via your Software



Steps:

- Go to Settings
- Choose “Billing integrations”
- Press the “plus” button on your right
- In the wizard window choose invoice integration (you may choose only one from the drop-down list)

Option 1 - WeFact
Option 2 - Xero

- You should add this in case you want to create documents via WeFact or Xero, but this is not obligatory, as you may create drafts of the invoices in our platform without using additional integrations.

- As our client, you may send requests for integration of the platform with new payment systems, if the needed one is not included in the list above

- If your invoicing software is not integrated, please use the Export feature to export all invoices to your invoicing software

5. Create Pools

Steps:

- Go to the “Pools” button on your left list of the Side navigation menu
- On your left press the “plus” button and choose the pool type you want to create

- One pool may be assigned to one company only

- To be able to create pools, you need to have plans already created a plan with this type of "subscription static pool"

6. Create e-SIM Store

In the eSIM Store one can publish only the Prepaid plan type

To have an e-SIM store enabled within the company, you need to create a plan(s) for e-SIMs.

Steps:

- Go to the “Plans” button on your left list of the side navigation menu
- On your right press the “plus” button and choose “New Prepaid Plan”
- In the wizard window fill in empty fields

⚠ IMPORTANT

To be able to have an e-SIM Store following conditions need to be taken:

- **The operator has to work with e-SIMs**
- **“Coverage” field has to be filled in**
- **You have to choose the “Recommended Plan” that will be set up for the SIM after its’ sale**
- **You need to turn on the switcher “e-SIM Store Plan”**

7. Invite Staff

This action can be done at any point of using the platform.
You may invite staff in case not only the account owner will work with SIMs

Steps:

- Go to Settings
- Choose Staff -> Staff tab will appear
- Press “plus” button on your right
- In the “Invite Staff” wizard type-in the e-mail address to send the invitation to join
- The invited staff member will need to follow instructions from the received mail to create an account

8. Customize Columns

This action can be done at any point of using the platform.

Steps:

- Go to the “SIMs” button on your left list of the Side navigation menu
- Press “plus” on the table header -> a wizard with the list of all names of the columns will appear
- Choose the ones you want to primarily see in your chart. In the same wizard window, you may also choose the sequence of the columns.

WEBHOOKS

This section briefly describes the concept of webhooks and how to use them with the platform. It provides basic terms and definitions related to the topic of webhooks, as well as the main entities and examples of payloads that your application can receive.

TERMS USED

This page uses generally accepted and newly coined terms. Definitions of the most commonly used terms in this document are provided below.

Push API – is the tool that allows websites to send instant messages (webhooks) to your Endpoint URL, on something triggers like creating a new account, changing SIM usage, etc. This tool sends a webhook from one application to another application when something event was triggered. For example, you can set up your endpoint URL to send a webhook notification when something Prepaid SIM Card has 200 MB of usage remaining.

All webhooks described in this docs use the standard JSON syntax.

Webhook – is the HTTP POST request to your endpoint URL, called in Push API process.

Endpoint URL – is a URL link to your-side web server that implements a handler of webhook with different payloads. For example, a handler of the "echo" webhook payload is necessarily required to work with Push API (for more details, see section 2).

Webhook Payload – is a body of Webhook. This docs define the set of default payloads that may occur in webhooks (for more information about webhook payloads, see section 5).

Webhook Payload Model – is a typical JSON that can occur in different webhook payloads. Models have required and optional fields, and are designed to unify sets of parameters of default objects like SIM Card, Account, etc. (for more information about webhook payload models, see section 4).

ENDPOINT VALIDATION

A prerequisite for using Push API is the validation of your endpoint URL. You must implement a webhook request handler of the “echo” type for this.

To validate your endpoint URL, the System will send an "echo" type request, with the “echo_data” parameter containing some short pseudo-random text. The size of the text will not exceed 50 characters. The text can include Latin letters, Arabic numbers, and the special character “:”.

In the response, the System will expect to receive a response of the “echo” type, with the “echo_data” parameter, the content of which is identical to what was in the request. You can simply copy the request body into your response, without any additional operations.

You can see examples of HTTP requests and responses (Listing 2.1, 2.2).

Listing 2.1 – Example of webhook HTTP request

```
POST /your_endpoint HTTP/1.1
Host: example.your_host.com
User-Agent: GuzzleHttp/7
Content-Type: application/json
Content-Length: 53

{"type":"echo","echo_data":"something:Echo:data:123"}
```

Listing 2.2 – Example of HTTP response

```
HTTP/1.1 200 OK
...
Content-Type: application/json

{"type":"echo","echo_data":"something:Echo:123"}
```

The implementation of this handler is required for working with the Push API.

Validation of the endpoint URL is carried out when it is created on the platform, and may also be repeated periodically during its use.

If you do not have the opportunity to implement a handler for the "echo" webhook payload, you can use the manual endpoint URL validation feature. This action is extremely undesirable and is permissible only in special cases!

If you use the manual validation feature, your endpoint URL will continue to be marked as "Invalid", but webhooks will still try to be sent to it. You also assume responsibility for any incorrect URL links. We reserve the right to block invalid URLs

---

AUTHENTICATION

To enhance security against potential attacks like spam and DDoS on your endpoint, we recommend implementing authentication for all incoming requests. For this purpose, we include a hash and timestamp in the headers of webhooks. The hash is generated using the SHA256 cryptographic algorithm, following the format:

api_secret:timestamp,

where:

api_secret – is a secret key that you can get by sending us a request for it,

timestamp – is a UNIX timestamp in seconds.

Listing 3.1 – Example of auth hash calculating (pseudocode)

```
var apiSecret = "96f3fbc76a921307f1c90d42a2203c9d";
var timestamp = 1704067200; // 2024-01-01 00:00:00
var hash = sha256(apiSecret + ":" + timestamp);
// hash = "68361c09cc50993ca6e0486e1f530c1d4e36a8aca9c8d20eb0d3aafbe47d2d5d"
```

It is recommended to always authenticate all incoming webhooks.

WEBHOOK PAYLOAD MODELS

Account Balance ModelWebhook Payload Models are unified JSON objects that can occur in different webhook payloads.

Models have required and optional fields. Required fields always exist in the model. Optional fields are parameters that may be absent for different reasons.

Below are defined all the models that can be found in webhook payloads.

Account Model

Account Model - is a base model for show platform account parameters. The fields of the Account model are presented in the table (Table 4.1), with an example below (Listing 4.1). There are the following types of this model (Table 4.2).

Table 4.1 - Fields of Account Model

№	Key	Type	Exists	Description
1	uuid	uuid v4, string	required	Main identifier of account
2	name	string	required	Public name of account
3	status	Status Model	optional	Platform Account Status
4	currency_code	string	optional	"EUR" , "USD" , etc.
5	email_for_notification	string	optional	If account has this parameter
6	vat_number	string	optional	Unique identifier of taxable person
7	address	Account Address Model	optional	If account has any of address parameters

Listing 4.1 – Example of Account Model (JSON)

```
{
   "uuid":                "ff472de7-59b2-425d-96ec-9d1ff4bab684",
   "name":                 "Example Company",
   "status":               {...//Status Model},
   "currency_code":          "EUR",
   "email_for_notification":  "email.notif@gmail.com",
   "vat_number":            "123456789",
   "address":              {...//Account Address Model}
}
```

Table 4.2 - Account Type

№	Title	Description
1	Admin	Has highest rules
2	Organisation	Has more rules than Companies and End Users and manages them
3	Company	Has more rules than End Users and manages them
4	End User	Main account of user with basic rules

End User Model

End User Model - is an extended model from Account Model for show platform end user parameters. The fields of the End User model are presented in the table (Table 4.3), with an example below (Listing 4.2).

Table 4.3 - Fields of End User Model

№	Key	Type	Exists
1-7		*Field of Account Model ...*	
8	owner	User Model	optional
9	company	Account Model	optional

Listing 4.2 - Example of End User Model (JSON)

```
{
   "uuid":                "ff472de7-59b2-425d-96ec-9d1ff4bab684",
   "name":                 "Example End User",
   "status":               {...//Status Model},
   "currency_code":          "EUR",
   "email_for_notification":  "email.notif@gmail.com",
   "vat_number":             "1234567890",
   "address":               {...//Account Address Model},
   "owner":                 {...//User Model},
   "company":                 {...//Account Model}
}
```

Account Address Model

The Account Address Model is the model that is most often found within the Account Model and contains information about its contact address. Its fields are presented in the table (Table 4.4), with an example (Listing 4.3).

Table 4.4 – Fields of Account Address Model

Table 4.4 – Fields of Account Address Model

№	Key	Type	Exists
1	house_number	string	optional
2	street	string	optional
3	addition	string	optional
4	city	string	optional
5	state	string	optional
6	country_code	string	optional
7	postal_code	string	optional

Listing 4.3 – Example of Account Address Model (JSON)

```
{
   "house_number":  "52/1",
   "street":       "Green Street",
   "addition":     "Office 23",
   "city":         "Example City",
   "state":        "Example State",
   "country_code":  "US",
   "postal_code":   "17000"
}
```

Wallet Model

The Wallet Model is a model that displays the wallet balance and currency of a specific account. Its fields are presented in the table (Table 4.5), with an example below (Listing 4.4).

Table 4.5 – Fields of Wallet Model

№	Key	Type	Exists	Description
1	remaining_balance	float	required	Remaining founds, 2 decimals
2	currency_code	string	required	Balance currency. "EUR", "USD", etc.

Listing 4.4 – Example of Wallet Model (JSON)

```
{
  "remaining_balance":  "52.99",
  "currency_code":     "EUR"
}
```

Plan Model

The Plan Model – is a base model for showing platform plan parameters. Model fields are presented in the table (Table 4.6), with an example below (Listing 4.5). There are the following types of this model (Table 4.7).

Table 4.6 – Fields of Plan Model

№	Key	Type	Exists	Description
1	uuid	uuid v4, string	required	Main identifier of plan
2	type	string	required	Type of the plan
3	name	string	required	Public name of the plan
4	operator_title	string	optional	Title of operator with integration, if it exists
5	prepaid_refills	array	optional	Available refills for this plan

Listing 4.5 – Example of Plan Model (JSON)


```
{
  "uuid":          "271295c0-2019-470d-b7d9-b79bc2686cfd",
  "type":          "prepaid",
  "name":           "Plan Name",
  "operator_title":  "Test Operator (Integration)"
  "prepaid_refills":        [
    {...//Refill Model},
    {...//Refill Model},
    ...other refills
  ]
}
```


Table 4.7 – Plan Types

№	Title	Description
1	Monthly bundle	Pay after using. One of subscription subtype
2	Pay per MB	There is static price for MB. One of subscription subtype
3	Unlimited	Usage is unlimited. One of subscription subtype
4	Dynamic pool	Pool usage can change. One of subscription subtype
5	Static pool	Pool usage can't change. One of subscription subtype
6	Prepaid	Need to pay before using
7	PAYGO	One of subscription subtype

Pool Model

The Pool Model – is a base model for showing platform pool parameters. Models fields are presented in the table (Table 4.8), with an example below (Listing 4.6).

Table 4.8 – Fields of Pool Model

№	Key	Type	Exists	Description
1	uuid	uuid v4, string	required	Main identifier of pool
2	name	string	required	Public name of pool
3	current_usage_kb	integer	optional	Current pool usage Listing

Listing 4.6 – Example of Pool Model (JSON)

```
{
  "uuid":          "34160e3a-3492-4f6f-bbf6-c9a9be15ea55",
  "name":           "Pool Name",
  "current_usage_kb":  "51200"
}
```

SIM Card Model

The SIM Card Model – is a base model for show platform SIM card parameters. Model fields are presented in the table (Table 4.9), with an example below (Listing 4.7). There are the following types of this model (Table 4.10).

Table 4.9 – Fields of SIM Card Model

№	Key	Type	Exists	Description
1	iccid	string	required	Identifier of SIM Card
2	sim_card_type	string	optional	Type of SIM, if it exists
3	msisdn	string	optional	Identifier of SIM line
4	imei	string	optional	Identifier of SIM device, if it exists
5	is_e_sim	boolean	optional	Shows whether it is an eSIM card
6	status	Status Model	optional	Current SIM Card status
7	plan	Plan Model	optional	SIM Plan, if it exists
8	company	Account Model	optional	Assigned SIM Company
9	end_user	Account Model	optional	Assigned SIM End User
10	details	array	optional	Details about SIM Card like usage, etc.

Listing 4.7 – Example of SIM Card Model (JSON)

```
{
  "iccid":         "8920000012345678900",
  "sim_card_type":   "prepaid",
  "msisdn":          "01234567890",
  "imei":           "86-000000-111111-2",
  "is_e_sim":        false,
  "status":         {...//Status Model},
  "plan":           {...//Plan Model},
  "company":          {...//Account Model},
  "end_user":         {...//End User Model},
  "details":         ["current_usage_kb" => 512, ...]
}
```

Table 4.10 – Sim Card Types

№	Title	Description
1	Prepaid	SIM Card with Prepaid Plan
2	Subscription	SIM Card with one of Subscription Plans
3	PAYGO	SIM Card with PAYGO Plan
4	Without Type	Blank SIM Card. May be unavailable

Status Model

The Status Model – is a model for show platform status parameters. Model fields are presented in the table (Table 4.11), with an example below (Listing 4.8). There are the following types of this model (Table 4.12).

Table 4.11 – Fields of Status Model

№	Key	Type	Exists	Description
1	id	id, integer	required	Status identifier
2	title	string	required	Status title

Listing 4.8 – Example of Status Model (JSON)

```
{
  "id":    "1",
  "title":  "pending",
}
```

Table 4.12 – Status Types

№	Title	Description
1	Pending	Waiting for activation
2	Invalid	Broken
3	Active	In use
4	Suspended	I pause for some reason. Waiting for reactivation.
5	Not Available	Some problems with SIM
6	Blank	Need to assign one of the Plans
7	Activation	Ready Active, wait for first usage
8	Archived	Deleted or SIM Swaped

User Model

The User Model – is a base model for showing platform user parameters. Models fields are presented in the table (Table 4.13), with an example below (Listing 4.9).

Table 4.13 – Fields of User Model

№	Key	Type	Exists	Description
1	uuid	uuid v4, string	required	Main identifier of user
2	email	string	required	User email
3	first_name	string	optional	User first name, if it set
4	last_name	string	optional	User last name, if it set
5	status	Status Model	optional	Current user status
6	company	Account Model	optional	If End User does not exist
7	end_user	End User Model	optional	Main account for user
8	is_two_factor_enabled	boolean	optional	Shows user 2FA state
9	timezone	string	optional	Shows user setting timezone
10	format_date_time	string	optional	User format for date and time
11	format_number	string	optional	User format for numbers

Listing 4.9 – Example of User Model (JSON)

```
{
  "uuid":                "b528bcb9-86cf-2d72834d02ab",
  "email":               "user_email@gmail.com",
  "first_name":          "John",
  "last_name":           "Johnson",
  "status":              {...//Status Model},
  "company":             {...//Account Model},
  "end_user":            {...//End User Model},
  "is_two_factor_enabled":  true,
  "timezone":            "UTC",
  "format_date_time":     "Y-m-d H:i:s",
  "format_number":       "fr-CA"
}
```

Prepaid Refill Model

The Prepaid Refill Model – is a model for showing platform plan prepaid refill parameters. Model fields are presented in the table (Table 4.14), with an example below (Listing 4.10).

Table 4.14 – Fields of Refill Model

№	Key	Type	Exists	Description
1	uuid	uuid v4, string	required	Main identifier of refill
2	amount_mb	integer	required	MB amount for refill
3	price	float	required	Price for refill (with VAT if it exists)
4	currency_code	string	required	"EUR", "USD", etc.

Listing 4.10 – Example of Refill Model (JSON)

```
{
  "uuid":        "b528bcb9-86cf-2d72834d02ab",
  "amount_mb":    "315",
  "price":        "90.95",
  "currency_code":  "USD"
}
```

Date Time Model

The Date Time Model – is a model for showing date time parameters like timestamp, time, date, and timezone. Model fields are presented in the table (Table 4.15), with an example below (Listing 4.11).

Table 4.15 – Fields of Date Time Model

№	Key	Type	Exists	Description
1	unix_timestamp	integer	required	Current date time timestamp
2	human_time	string	required	Time in format 'H:i:s'
3	human_date	string	required	Date in format 'Y-m-d'
4	timezone	string	required	Always UTC

Listing 4.11 – Example of Date Time Model (JSON)

```
{
  "unix_timestamp":  "1711929600",
  "human_time":     "17:25:11",
  "human_date":     "2024-06-12",
  "timezone":       "UTC"
}
```

Rule Model

Rule Model – is a model for showing platform rule parameters. Model fields are presented in the table (Table 4.16), with an example below (Listing 4.12).

Table 4.16 – Fields of Rule Model

№	Key	Type	Exists	Description
1	uuid	string	required	Identificator of current rule
2	name	string	required	Rule name
3	description	string	required	Short info about the rule

Listing 4.12 – Example of Rule Model (JSON)

```
{
  "uuid":        "1711929600",
  "name":         "Rule for refills",
  "description":   "Send notification to email when usage reaches 100MB"
}
```

Payment Model

The Payment Model – is a model for show platform payment parameters. Model fields are presented in the table (Table 4.17), with an example below (Listing 4.13). Types of Payment Methods describes in the Table 4.18 and types of Refill Types – in the table 4.19. { "unix_timestamp": "1711929600", "human_time": "17:25:11", "human_date": "2024-06-12", "timezone": "UTC" } { "uuid": "1711929600", "name": "Rule for refills", "description": "Send notif to email when usage reaches 100MB" } Sheets Doc. No. Date Sheet Rev. Push API Documentation 18 3 27 PPAD.01

Table 4.17 – Fields of Payment Model

№	Key	Type	Exists	Description
1	uuid	uuid v4, string	required	Main identifier of payment
2	ransaction_id	string	required	Identificator of transaction
3	amount	float	required	Money amount for payment
4	currency_code	string	required	EUR, USD, etc
5	payment_method	string	required	Wallet, card, etc.
6	refill_type	string	required	Payment object – refill balance, sim or etc.
7	credit_card	Credit Card Model	optional	If payment was made with card
8	email	string	optional	Purchaser email
9	name	string	optional	Purchaser name
10	company	Account Model	optional	Purchaser company if end user does not exist
11	end_user	End User Model	optional	Purchaser end user
12	receipt_url	string	optional	Link to see original receipt
13	metadata	array	optional	Additional info about payment
14	payment_at	Date Time	required	Operation date

Listing 4.13 – Example of Payment Model (JSON)

```
{
  "uuid":          "b528bcb9-86cf-2d72834d02ab",
  "transaction_id":  "txn_322Knxiwkaxi4d26770123zOgB",
  "amount":         "35.20",
  "currency_code":   "EUR",
  "payment_method":  "card",
  "refill_type":     "refill_sim",
  "credit_card":     {...//Credit Card Model},
  "email":          "purchaser@gmail.com",
  "name":           "Purchaser Name",
  "company":        {...//Account Model},
  "end_user":       {...//End User Model},
  "receipt_url":     "https://payment.com/jajkxawjj...",
  "metadata":       ["sim_iccid" => "8991712410", ...],
  "payment_at":      {...//Date Time Model},
}
```

Table 4.18 – Payment Types

№	Title	Description
1	wallet	Money for payment was used from Wallet
2	bancontact	One of Internet payment types. Depends on your payment integration
3	card	Money for payment was used from assigned to the Account credit card
4	cardknox	One of Internet payment types. Depends on your payment integration
5	ideal	One of Internet payment types. Depends on your payment integration
6	manual	Payment was made by admin manually
7	paypal	One of Internet payment types. Depends on your payment integration
8	stripe	One of Internet payment types. Depends on your payment integration

Table 4.19 – Refill Types

№	Title	Description
1	Refill SIM	SIM Card MB amount was refilled
2	Refill eSIM	eSIM Card MB amount was refilled
3	Refill Wallet	Wallet money amount was refilled
4	Buying an eSIM	eSIM Purchase

Credit Card Model

The Credit Card Model – is a model for show platform credit card parameters. Model fields are presented in the table (Table 4.20), with an example below (Listing 4.14).

Table 4.20 – Fields of Credit Card Model

№	Key	Type	Exists	Description
1	uuid	string	required	Identificator of current rule
2	name	string	required	Rule name
3	description	string	required	Short info about the rule

Listing 4.14 – Example of Credit Card Model (JSON)

```
{
   "uuid":        "1711929600",
   "name":        "Rule for refills",
   "description":  "Send notification to email when usage reaches 100MB"
}
```

WEBHOOK PAYLOAD TYPES

Payload Types are data arrays with defined keys and data, some of which are required and others are not.

You need to be prepared to process different payload types and know which data they contain.

"Optional" marks fields that must not be present in the body of the webhook that is sent. Fields without 'optional' marks are required.

The listed types may change, and new ones may be added. Please follow the latest revisions of this documentation.

Technical Payloads

These payloads are designed to perform certain system operations, such as endpoint validation, for example.

echo – a payload that is sent to an endpoint for validation. The endpoint must respond with the same payload (Listing 5.1).

Listing 5.1 – Example of echo Payload (JSON)

```
{
   "type":         "echo",
   "data":         {
      "echo_data":      "echo:Ariel:1715854344"
   },
   "created_at":    {...//Date Time Model}
}
```

Event Payloads

Event Payloads are payloads that are sent to the endpoint when a certain event occurs, such as a user being deleted or company data being changed.

Company Events

Company Events are triggered by certain actions aimed at the company or elements related to it.

company.added – occurs when an Organization adds a company to itself and its owner completes registration, not only invited (Listing 5.2).

Listing 5.2 – Example of company.added payload (JSON)

```
// Some code
```

company.deleted – occurs when the Organization deletes its own company. The company must be confirmed and has a confirmed owner, not only invited (Listing 5.3).

Listing 5.3 – Example of company.deleted Payload (JSON)

```
{
  "type":      "company.added",
  "data":      {
    "company":    {...//Account Model}
  },
  "created_at":  {...//Date Time Model}
}
```

company.updated – occurs when certain data is updated in an organization's Company, such as a name or currency, for example (Listing 5.4).

Listing 5.4 – Example of company.updated Payload (JSON)

```
{
  "type":      "company.updated",
  "data":      {
    "company":    {...//Account Model}
  },
  "created_at":  {...//Date Time Model}
}
```

Company Staff Events

Company Staff Events are triggered by certain actions aimed at the company's employees (staff).

company.staff.added – occurs when the Company adds a Staff user to itself. The staff user must be registered, not only invited (Listing 5.5).

Listing 5.5 – Example of company.staff.added Payload (JSON)

```
{
```

```
  "type":    "company.staff.added",
  "data":    {
    "user":    {...//User Model}
  },
  "created_at":  {...//Date Time Model}
}
```

company.staff.deleted – occurs when Company deletes its own Staff user. The user must be registered, not only invited (Listing 5.6).

Listing 5.6 – Example of company.staff.deleted Payload (JSON)

```
{
  "type":    "company.staff.deleted",
  "data":    {
    "user":    {...//User Model}
  },
  "created_at":  {...//Date Time Model}
}
```

company.staff.updated – occurs when certain data is updated in an company's Staff user, such as a name or email, for example (Listing 5.7).

Listing 5.7 – Example of company.staff.updated Payload (JSON)

```
{
  "type":    "company.staff.updated",
  "data":    {
    "user":    {...//User Model}
  },
  "created_at":  {...//Date Time Model}
}
```

End User Events

End User Events are triggered by certain actions aimed at the End User or elements related to it.

end_user.added – occurs when Company or Organization adds an End User to themselves and the end user confirms registration, not only be invited (Listing 5.8).

Listing 5.8 – Example of end_user.added Payload (JSON)

```
{
  "type":      "end_user.added",
  "data":      {
    "end_user":    {...//End User Model}
  },
```

```
  "created_at":  {...//Date Time Model}
}
```

end_user.deleted – occurs when Company or Organization deletes their own End User. The end user must be registered, not only invited (Listing 5.9).

Listing 5.9 – Example of end_user.deleted Payload (JSON)

```
{
  "type":       "end_user.deleted",
  "data":       {
    "end_user":    {...//End User Model}
  },
  "created_at":  {...//Date Time Model}
}
```

end_user.updated – occurs when certain data is updated in an organization's/company's End User, such as a name or email, for example (Listing 5.10).

Listing 5.10 – Example of end_user.updated Payload (JSON)

```
{
  "type":       "end_user.updated",
  "data":       {
    "end_user":    {...//End User Model}
  },
  "created_at":   {...//Date Time Model}
}
```

SIM Card Events
SIM Card Events are triggered by certain actions aimed at the SIM Card.

sim_card.assigned_to_end_user – occurs when a blank SIM Card has been assigned to the End User. There are two situations: the end user registers on the platform with a bought SIM Card or the Organization / Company assigns its own SIM card to its end user (Listing 5.11).

Listing 5.11 – Example of sim_card.assigned_to_end_user Payload (JSON)

```
{
  "type":       "sim_card.assigned_to_end_user",
  "data":       {
    "sim_card":    {...//Sim Card Model}
  },
  "created_at":   {...//Date Time Model}
}
```

sim_card.first_usage – occurs when the SIM was first used (Listing 5.12).

Listing 5.12 – Example of sim_card.first_usage Payload (JSON)

```
{
  "type":       "sim_card.first_usage",
  "data":       {
    "sim_card":    {...//Sim Card Model}
  },
  "created_at":   {...//Date Time Model}
}
```

Payment Payloads

payment.created – occurs when some payment has been made (Listing 5.13). The payload contains only either the company or the end user.

Listing 5.13 – Example of payment.created Payload (JSON)

```
{
  "type":       "payment.created",
  "data":       {
    "payment":     {...//Payment Model}
  },
  "created_at":   {...//Date Time Model}
}
```

Rule Payloads

These payloads are sent to the endpoint when certain rules that have been created for the account are triggered.

SIM Card Rule Payloads

It occurs when certain rules related to the SIM have been triggered.

rule.sim_card.current_usage.triggered – occurs when SUBSCRIPTION SIM Card current usage has been changed (Listing 5.14).

Listing 5.14 – Example of rule.sim_card.current_usage.triggered Payload (JSON)

```
{
  "type":       "rule.sim_card.current_usage.triggered",
  "data":       {
    "rule":       {...//Rule Model},
```

```
    "sim_card":   {...//Sim Card Model}
  },
  "created_at":  {...//Date Time Model}
}
```

rule.sim_card.status.changed – occurs when SIM Card status has been changed. SIM Card has been activated for example (Listing 5.15).

Listing 5.15 – Example of rule.sim_card.status.changed Payload (JSON)

```
{
  "type":      "rule.sim_card.status.changed",
  "data":      {
    "rule":      {...//Rule Model},
    "sim_card":   {...//Sim Card Model}
  },
  "created_at":  {...//Date Time Model}
}
```

rule.sim_card.remaining_usage.triggered – occurs when PREPAID SIM Card remaining usage has been changed (Listing 5.16).

Listing 5.16 – Example of rule.sim_card.remaining_usage_triggered Payload (JSON)

```
{
  "type":      "rule.sim_card.remaining_usage.triggered",
  "data":      {
    "rule":      {...//Rule Model},
    "sim_card":   {...//Sim Card Model}
  },
  "created_at":  {...//Date Time Model}
}
```

rule.sim_card.imei.changed – occurs when SIM Card IMEI has been changed (Listing 5.17).

Listing 5.17 – Example of rule.sim_card.imei.changed Payload (JSON)

```
{
  "type":      "rule.sim_card.imei.changed",
  "data":      {
    "rule":      {...//Rule Model},
    "sim_card":    {...//Sim Card Model}
  },
  "created_at":  {...//Date Time Model}
}
```

rule.sim_card.date.triggered – occurs when Rule “Date Triggered” has been happened (Listing 5.18).

Listing 5.18 – Example of rule.sim_card.date.triggered Payload (JSON)

```
{
  "type":     "rule.sim_card.date.triggered",
  "data":     {
    "rule":      {...//Rule Model},
    "sim_card":   {...//Sim Card Model}
  },
  "created_at":  {...//Date Time Model}
}
```

SIM Pools Payloads

Occurs when certain rules related to the SIM Pool has been triggered.

rule.pool.usage.triggered – occurs when Pool usage has been changed (Listing 5.19).

Listing 5.19 – Example of rule.pool.usage.triggered Payload (JSON)

```
{
  "type":    "rule.pool.usage.triggered",
  "data":    {
     "rule":    {...//Rule Model},
     "pool":    {...//Pool Model}
  },
  "created_at":  {...//Date Time Model}
}
```

Wallet Payloads

rule.company.wallet.triggered – occurs when the Wallet has been refilled or spent (Listing 5.20). The payload contains only either the company or the end user.

Listing 5.20 – Example of rule.company.wallet.triggered Payload (JSON)

```
{
  "type":      "rule.company.wallet.triggered",
  "data":      {
     "rule":      {...//Rule Model},
     "company":    {...//Account Model},
     "wallet":     {...//Wallet Model}
  },
   "created_at":  {...//Date Time Model}
}
```