### **Introduction**

This document describes the **Service Plans API** for managing and configuring service plans for users on the platform. The API allows clients to interact with user data and service plans, including creating, updating, and deleting plans based on user status and provider configurations. It is essential for integrating various services and ensuring proper management of user subscriptions.

#### **Purpose of the Specification**

This specification provides a detailed description of the **API endpoints**, including how to retrieve user data, manage service plans, and implement **Basic Authentication** for modifying user subscriptions. The **OpenAPI** format is used to standardize the API documentation, making it easier for developers to integrate and use the system.

#### **Who Should Use This Documentation?**

This document is intended for developers and technical teams who need to integrate with the API. It is useful for:

* Understanding the available API endpoints and their functionality.
* Integrating with the platform's service plans and user management system.
* Implementing authentication and handling the lifecycle of user subscriptions.

#### **Key Features of the API**

* **User Management**: Manage and retrieve data for users across various types (guest, basic, advanced, company).
* **Service Plan Management**: Create, update, and delete service plans tailored for each user type and provider.
* **Access Control**: Implement **Basic Authentication** for modifying user plans and data.
* **Pagination**: Support for paginated responses when querying large sets of user data.

**OpenAPI Specification - Service Plans API**

openapi: 3.0.0

info:

title: Service Plans API

version: 1.0.0

description: API for managing user service plans

paths:

/api/external/user:

get:

summary: Get user information

parameters:

- name: user\_id

in: query

description: Unique user identifier

required: false

schema:

type: string

example: "3453255"

- name: from

in: query

description: Pagination start timestamp

required: false

schema:

type: integer

example: 1625097600

- name: to

in: query

description: Pagination end timestamp

required: false

schema:

type: integer

example: 1625184000

- name: limit

in: query

description: Limit to records returned

required: false

schema:

type: integer

default: 1000

example: 100

- name: provider\_id

in: query

description: Unique partner identifier

required: false

schema:

type: string

example: "2323"

responses:

'200':

description: Successful response with user data

content:

application/json:

schema:

type: object

properties:

user\_id:

type: string

example: "3453255"

provider\_id:

type: string

example: "2323"

status:

type: string

enum: [active, inactive, idle]

example: "active"

type:

type: string

enum: [basic, guest, company, advanced, admin]

example: "basic"

sub\_status:

type: string

enum: [signed, unsigned, absent]

example: "signed"

zone:

type: string

example: "us"

'401':

description: Unauthorized - Invalid credentials

'404':

description: User not found

/api/partner:

get:

summary: Get partner information

parameters:

- name: provider\_id

in: query

description: Unique partner identifier

required: true

schema:

type: string

example: "2323"

- name: status

in: query

description: Partner operation state

required: false

schema:

type: string

enum: [active, stopped]

example: "active"

responses:

'200':

description: Successful response with partner data

content:

application/json:

schema:

type: object

properties:

provider\_id:

type: string

example: "2323"

status:

type: string

enum: [active, stopped]

example: "active"

'401':

description: Unauthorized - Invalid credentials

'404':

description: Partner not found

/api/external/plans:

post:

summary: Create a new service plan

security:

- basicAuth: []

requestBody:

description: Plan creation data

content:

application/json:

schema:

type: object

properties:

plan\_name:

type: string

example: "Premium Plan"

provider\_id:

type: string

example: "2323"

type:

type: string

enum: [basic, company, advanced]

example: "basic"

responses:

'201':

description: Plan created successfully

'400':

description: Bad request - Invalid plan data

'401':

description: Unauthorized - Invalid credentials

put:

summary: Update an existing service plan

security:

- basicAuth: []

requestBody:

description: Plan update data

content:

application/json:

schema:

type: object

properties:

plan\_name:

type: string

example: "Premium Plan Updated"

provider\_id:

type: string

example: "2323"

type:

type: string

enum: [basic, company, advanced]

example: "advanced"

responses:

'200':

description: Plan updated successfully

'400':

description: Bad request - Invalid plan data

'401':

description: Unauthorized - Invalid credentials

delete:

summary: Delete a service plan

security:

- basicAuth: []

parameters:

- name: plan\_id

in: query

description: Unique plan identifier

required: true

schema:

type: string

example: "plan12345"

responses:

'204':

description: Plan deleted successfully

'404':

description: Plan not found

'401':

description: Unauthorized - Invalid credentials

securitySchemes:

basicAuth:

type: http

scheme: basic