



## Documento técnico de integração da API de Vendas de ingressos

PROJETO	
Projeto	API para integração de geração de ingressos
Gerente do Projeto Designado	Marcos Vinicius / Lourivaldo Pereira dos Anjos
Equipe de projeto (PWI)	EQUIPE N1: <a href="mailto:equipe.n1@pwi.com.br">equipe.n1@pwi.com.br</a>
VERSÃO DOCUMENTO:	Versão: 1.0.21

Informações Gerais .....	5
Finalidade.....	5
Necessidades iniciais para uso da API de integração.....	5
Informações técnicas .....	5
URLs de comunicação .....	5
Ambiente PWI.....	5
Formato de comunicação .....	5
Limite de requisições .....	6
Autenticação de acesso .....	6
Exemplo de envio .....	6
Exemplo de retorno.....	6
Autorização Bearer Token.....	6
Métodos da API para manipulação de ingressos .....	7
Produtos disponíveis para venda .....	7
Validações:.....	7
Exemplo de retorno.....	8
Sessões disponíveis para venda de um determinado produto .....	9
Validações:.....	9
Exemplo de retorno.....	10
Consulta de saldo disponível.....	10
Explicação do método .....	11
Exemplo de retorno.....	11
Gerar venda e ingressos.....	11
Validações feitas pela API.....	11
Combos de produtos na venda:.....	12
O que é um passaporte?.....	12
Parâmetros de envio.....	12
Gerar venda de ingressos com controle de sessão diária .....	16
Consulta de Ingressos .....	17
Explicação do método .....	17
Exemplo de retorno.....	18
Consulta de Ingressos da Venda .....	19
Explicação do método .....	19

Exemplo de retorno .....	19
Consulta de lista de vendas .....	21
Explicação do método .....	21
Exemplo de Retorno .....	21
Consulta de Venda por ID .....	24
Exemplo de retorno .....	24
Cancelar venda e ingressos gerados .....	26
Explicação do método .....	26
Exemplo de retorno .....	27
Sucesso .....	27
Falha .....	27
Lista de códigos de retorno .....	27
E-Ticket (Ingresso).....	27
Código de barras / QR Code do E-Ticket (Ingresso).....	28
.....	28
Roteiro de Homologação.....	28
1 - Autenticação .....	28
2- Consulta de produtos disponíveis para venda .....	29
3- Consulta de sessões disponíveis para um determinado produto .....	29
4 - Consulta de saldo da conta consumo disponível .....	29
5 - Gerar vendas e ingressos .....	30
5.1 - Gerar vendas e ingressos com sessão.....	31
6 - Consulta de ingressos.....	32
7 - Consulta lista de vendas.....	32
8 - Consulta de venda por ID .....	33
9 - Cancelar venda e ingressos gerados .....	34
10 – Logout [obsoleto] .....	34
Log de alterações .....	34

## Informações Gerais

### Finalidade

A API de Vendas tem como objetivo integrar pedidos de venda e geração de ingressos, diretamente no sistema ERP do parque, através de requisições HTTP.

O que é possível fazer?

- Consultar os produtos disponíveis para venda pelo parceiro do parque
- Consultar o saldo de conta consumo, para parceiros que trabalhem com essa modalidade
- Incluir pedidos de venda no sistema ERP do parque
- Gerar os ingressos a partir dos pedidos incluídos no sistema
- Consultar pedidos gerados no sistema ERP via API
- Consultar os dados dos ingressos gerados no sistema ERP via API
- Cancelar os pedidos de vendas gerados no sistema ERP via API (consultar regras de cancelamento diretamente com o parque)

### Necessidades iniciais para uso da API de integração

Para se iniciar os trabalhos de integração com o sistema ERP do parque via API, faz-se necessário o cadastro junto aos responsáveis comercial do parque, para que seja criado o registro do parceiro e a emissão da senha de acesso à API. Somente após o cadastro do parceiro feito pela equipe do parque, será possível realizar o consumo dos métodos descritos nesse manual.

Com o cadastro realizado e a senha disponibilizada pela equipe do parque, recomendamos o consumo da API em ambiente de homologação, para que se possa validar todos os métodos descritos nesse documento.

## Informações técnicas

### URLs de comunicação

PRODUÇÃO	<a href="https://api.{cliente}.com.br/prod/">https://api.{cliente}.com.br/prod/</a>
HOMOLOGAÇÃO	<a href="https://api.{cliente}.com.br/hom/">https://api.{cliente}.com.br/hom/</a>

### Ambiente PWI

Caso o cliente não possua o ambiente de homologação, a PWI disponibiliza um ambiente interno para realizar os testes.

Todos os passos continuaram semelhantes.

PWI	<a href="http://apivendas.pwi.com.br/prod/">apivendas.pwi.com.br/prod/</a>
-----	--

### Formato de comunicação

Toda a comunicação da API de Vendas é através do formato **JSON**

```
Content-Type: "Application/JSON"
```

```
Accept: "Application/JSON"
```

## Limite de requisições

Para evitar gargalos e imprevistos que possam influenciar sua operação, a API permite até 8 requisições por segundo.

## Autenticação de acesso

Os parâmetros de autenticação da API são enviados no body da requisição, para efetuar qualquer transação será necessário à autenticação antes. Segue abaixo na descrição dos métodos, informações dos parâmetros do método autenticar.

### POST: {url}/api/auth/login

Nome	Descrição	Formato	Obrigatório
Usuario	CNPJ do integrador	Decimal	Sim
Senha	Senha fornecida no cadastro do integrador	String	Sim

### Exemplo de envio

```
{  
  "USUARIO":58108408000161,  
  "SENHA":"senha123"  
}
```

### Exemplo de retorno

```
{  
  "errors": [],  
  "result": {  
    "access_token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...",  
    "inativo": false,  
    "versaoAtual": "1.0.18.0",  
    "linkDocumentacao": "https://apivendas.pwi.com.br/prod"  
  }  
}
```

## Autorização Bearer Token

Para a utilização dos demais métodos é necessária a autenticação através do padrão Http Bearer Token.

O campo "access\_token" recebido no JSON de retorno do método de autenticação é utilizado para este fim e deve ser enviado no cabeçalho "Authorization".

Params Authorization Headers (10) Body Pre-request Script Tests Settings Cookies

Type Bearer To... Token eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...

## Métodos da API para manipulação de ingressos

### Produtos disponíveis para venda

Consulta todos os produtos disponíveis para venda do integrador.

#### GET: {url}/api/produto/lista

Nome	Descrição	Formato	Obrigatório
idProduto	Código de produto que buscamos	texto	Não
dataVisita	Data que cliente pretende visitar o parque	yyyy-mm-dd	Não

Ambos os parâmetros possuem a função de filtrar a tabela de produtos, de maneira que retorne um resultado mais rápido e preciso.

Comumente é possível enviar uma lista de parâmetros e as bibliotecas já possuem preparo para recebê-los. Porém também podem ser enviados juntos a URL, separando por “?” (Exemplo abaixo).

Exemplos de envio:

- Sem nenhum parâmetro: produto/lista
- Apenas com idProduto: produto/lista?idProduto=a595
- Apenas com dataVisita: produto/lista?dataVisita=2023-04-10
- Ambos os parâmetros: produto/lista?idProduto=a595&dataVisita=2023-04-10

#### Validações:

- O sistema verifica automaticamente se o produto possui uma sessão e, caso possua, requer que o usuário informe a data de visita.
- O sistema verifica automaticamente se o produto está na lista de produtos disponíveis e retorna uma mensagem de erro caso contrário.
- O sistema verifica automaticamente se o integrador está autorizado a vender o produto e retorna uma mensagem de erro caso contrário.
- Quando o parâmetro de data de visita é informado
  - O sistema verifica automaticamente se o produto pode ser vendido no dia da semana e no dia do mês informados. Caso contrário, o sistema retorna uma mensagem de erro.
  - O sistema verifica automaticamente se o parque estará fechado na data informada e retorna uma mensagem de erro caso contrário.
  - O sistema verifica automaticamente se o produto não estará disponível na data de previsão de visita informada e retorna uma mensagem de erro caso contrário.
  - O sistema verifica automaticamente se o produto estará vencido na data de previsão de visita informada e retorna uma mensagem de erro caso contrário.

- **Verificação de grupo terminal:** O sistema realiza automaticamente a verificação de disponibilidade do produto para o grupo terminal de vendas da API. Caso o produto não esteja associado a nenhum grupo terminal, ele será exibido para todos os terminais. Caso contrário, se não estiver liberado para o grupo terminal específico, o produto não será exibido.
- **Verificação de empresa:** O sistema verifica automaticamente se o produto está associado a uma empresa específica. Caso o produto não esteja vinculado a nenhuma empresa, ele será exibido para todos os terminais. Caso não esteja liberado para uma empresa específica, o produto será ocultado.
- **Verificação de disponibilidade de compra:** O sistema verifica a disponibilidade de compra do produto, podendo ser:
  - **Antecipado:** O produto será exibido apenas para datas de visita futuras, após a data atual.
  - **Dia atual:** O produto será exibido somente para visitas agendadas para o mesmo dia.
  - **Qualquer dia:** O produto será exibido independentemente da data da visita.

### Exemplo de retorno

```
{
  "errors": [],
  "result": [
    {
      "id": "000004",
      "nome": "INGRESSO ADULTO",
      "preco": 170.000000,
      "visitaObrigatorio": false,
      "produtoComSessao": false,
      "dataValidadeInicio": null,
      "dataValidadeTermino": null,
      "qtDiasValidade": 5
    },
    {
      "id": "000005",
      "nome": "INGRESSO ANUAL",
      "preco": 230.000000,
      "visitaObrigatorio": true,
      "produtoComSessao": false,
      "dataValidadeInicio": "2022-01-01T00:00:00",
      "dataValidadeTermino": "2025-12-31T00:00:00",
      "qtDiasValidade": 0
    }
  ]
}
```

#### Sobre o retorno:

- "errors": Array vazio que indica a lista de erros encontrados pelo sistema durante a validação dos dados. Caso não haja erros, esse campo estará vazio.

- "result": Array contendo todos os produtos válidos. Cada objeto dentro do array representa um produto e contém os seguintes campos:
  - "id": Identificador único do produto.
  - "nome": Nome do produto.
  - "preco": Preço do produto.
  - "visitaObrigatorio": Indica se a visita é obrigatória para a compra deste produto.
  - "produtoComSessao": Indica se o produto possui sessão.
  - "dataValidadeInicio": Data de início de validade do produto (opcional).
  - "dataValidadeTermino": Data de término de validade do produto (opcional).
  - "qtDiasValidade": Quantidade de dias de validade do produto a partir da data de compra (0 se não houver validade).

## Sessões disponíveis para venda de um determinado produto

Consulta todos as datas e os horários de sessões disponíveis para venda de um produto dentro de um período de até 30 dias.

**GET:** {url}/api/produto/listaSesoes

Nome	Descrição	Formato	Obrigatório
idProduto	Código de produto que buscamos	texto	Sim
dataInicio	Data e hora de início do período cujas sessões estamos buscando.	yyyy-mm-dd hh:mm:ss	Sim
dataFim	Data e hora de término do período cujas sessões estamos buscando.	yyyy-mm-dd hh:mm:ss	Sim

A transmissão de parâmetros é uma prática comum, facilitada pelo preparo das bibliotecas para recebê-los. Alternativamente, esses parâmetros podem ser anexados à URL, separados por um ponto de interrogação (como exemplificado abaixo).

Exemplos de envio: produto/lista? idProduto=999&dataInicio=2024-01-01T09:00:00&dataFim=2024-01-30T23:59:00

### Validações:

- O sistema verifica automaticamente se a data de início é posterior à data final. Se essa condição for verdadeira, retorna uma mensagem de erro ao usuário;
- O sistema verifica se o período entre a data de início e de fim é superior a 30. Em caso afirmativo, retorna uma mensagem de erro ao usuário;
- O sistema verifica automaticamente se o produto está incluído na lista de produtos disponíveis e, caso contrário, retorna uma mensagem notificando a não existência de um produto condizente com o ID informado;
- O sistema verifica automaticamente se existem sessões disponíveis para o produto dentro do período especificado. Se a lista de sessões retornada pelo serviço for nula ou vazia, informa-se que não foi detectada uma sessão referente ao produto e ao período especificados.

## Exemplo de retorno

```
{
  "errors": [],
  "result": [
    {
      "id": "1",
      "dhSessao": "2024-01-01T00:00:00",
      "qtDisponivel": 150
    },
    {
      "id": "2",
      "dhSessao": "2024-01-02T09:00:00",
      "qtDisponivel": 150
    },
    {
      "id": "3",
      "dhSessao": "2024-01-02T10:30:00",
      "qtDisponivel": 200
    }
  ]
}
```

Sobre o retorno:

- "errors": Array que indica a lista de erros encontrados pelo sistema durante a validação dos dados. Caso não haja erros, esse campo estará vazio.
- "result": Array contendo todos os produtos válidos. Cada objeto dentro do array representa um produto e contém os seguintes campos:
  - "id": Identificador único da sessão;
  - "dhSessao": Data e horário em que a sessão ocorrerá.;
  - "qtDisponivel": Quantidade de vagas disponíveis para a sessão em questão.

## Consulta de saldo disponível

- **Conta de consumo**  
A modalidade Conta Consumo é uma forma de pagamento pré-paga, consulte com o administrador do parque se esta modalidade está habilitada para o seu cadastro.
- **Dinheiro**  
A modalidade Dinheiro é uma forma de pagamento pós-pago, é adicionado o limite de compra antecipado para o parceiro. Consulte com o administrador do parque se esta modalidade está habilitada para o seu cadastro.

## Explicação do método

Consulta o saldo da conta consumo do Integrador.

**GET:** {url}/api/creditos/saldo

## Exemplo de retorno

```
{
  "errors": [],
  "result": {
    "saldo": "100.40"
  }
}
```

## Gerar venda e ingressos

Responsável por incluir as vendas para gerar os ingressos.

**POST:** {url}/api/venda/incluir

## Validações feitas pela API

Responsáveis por validar a inclusão das vendas e a geração de ingressos.

- Validações da venda:
  - Validação para verificar se o parceiro possui saldo suficiente para finalizar a venda.
  - Validação para verificar se o parceiro está ativo.
  - Validação se a data da venda é posterior à data de hoje, ignorando os segundos.
  - Validações dos itens da venda:
    - O sistema verifica automaticamente se o produto possui uma sessão e, caso possua, requer que o usuário informe a data de visita.
    - O sistema verifica automaticamente se o produto está na lista de produtos disponíveis e retorna uma mensagem de erro caso contrário.
    - O sistema verifica automaticamente se o integrador está autorizado a vender o produto e retorna uma mensagem de erro caso contrário.
    - Quando o parâmetro de data de visita é informado
      - O sistema verifica automaticamente se o produto pode ser vendido no dia da semana e no dia do mês informados. Caso contrário, o sistema retorna uma mensagem de erro.
      - O sistema verifica automaticamente se o parque estará fechado na data informada e retorna uma mensagem de erro caso contrário.
      - O sistema verifica automaticamente se o produto não estará disponível na data de previsão de visita informada e retorna uma mensagem de erro caso contrário.
      - O sistema verifica automaticamente se o produto estará vencido na data de previsão de visita informada e retorna uma mensagem de erro caso contrário.

- **Verificação de grupo terminal:** O sistema realiza automaticamente a verificação de disponibilidade do produto para o grupo terminal de vendas da API. Caso o produto não esteja associado a nenhum grupo terminal, ele será exibido para todos os terminais. Caso contrário, se não estiver liberado para o grupo terminal específico, o produto não será exibido.
  - **Verificação de empresa:** O sistema verifica automaticamente se o produto está associado a uma empresa específica. Caso o produto não esteja vinculado a nenhuma empresa, ele será exibido para todos os terminais. Caso não esteja liberado para uma empresa específica, o produto será ocultado.
  - **Verificação de disponibilidade de compra:** O sistema verifica a disponibilidade de compra do produto, podendo ser:
    - **Antecipado:** O produto será exibido apenas para datas de visita futuras, após a data atual.
    - **Dia atual:** O produto será exibido somente para visitas agendadas para o mesmo dia.
    - **Qualquer dia:** O produto será exibido independentemente da data da visita.
- Valida se permite o uso da Conta Consumo;
  - Valida se o saldo da Conta Consumo é insuficiente.
  - Valida se o saldo do limite em dinheiro é insuficiente.

### Combos de produtos na venda:

É possível adicionar um combo entre os produtos disponíveis para venda. Nesses casos, ao consultar os itens da venda, o combo será retornado como uma lista de itens individuais, detalhando cada produto que o compõe. Para identificar se um item pertence a um combo, utilize o campo `IsItemCombo`, que indicará se o item é parte de um conjunto. O campo `IdProdutoCombo` trará o ID correspondente ao combo do qual o item faz parte.

### O que é um passaporte?

Um passaporte é um conjunto de um ou mais ingressos, representado por um único QR Code, o que facilita a impressão para o cliente e o uso nas atrações. Ele permite controlar a quantidade de acessos em uma catraca através do campo `QtMultiplo`, que define o número de vezes que o passaporte pode ser utilizado.

### Parâmetros de envio

Nome	Descrição	Formato	Obrigatório
<code>terminalVenda</code>	Identificador do terminal onde a venda foi realizada	String	Sim
<code>usuario</code>	Usuário responsável pela venda	String	Sim

idVendaOrigem	Identificador da venda no sistema de origem	String	Sim
dataHoraVenda	Data e hora em que a venda foi efetuada	DateTime (yyyy-MM-ddTHH:mm:ss)	Sim
itens	Lista de itens vendidos	Array<ItemVenda>	Sim
idProduto	Identificador do produto/ingresso	String	Sim
qtde	Quantidade de ingressos do produto	Int	Sim
dataPrevisaoVisita	Data prevista para utilização do ingresso	Date (yyyy-MM-dd)	variável Definida de acordo com o item. Sempre verificar na rota de listagem de produtos.
ingressosInfo	Dados dos portadores dos ingressos	Array<IngressoInfo>	Não
nome	Nome completo do visitante	String	Não
email	E-mail do visitante	String	Não
nascimento	Data de nascimento do visitante	Date (yyyy-MM-dd)	Não
telefone	Telefone de contato do visitante	String	Não
documento	Número do documento do visitante	String	Não
tipoDocumento	Tipo do documento informado (CPF, RG, Passaporte, etc.)	String	Não

#### Exemplo de envio

```
{
  "terminalVenda": "CLIENTEPWI",
  "usuario": "OPERADOR01",
```

```
"idVendaOrigem": "VENDAINFO1",
"dataHoraVenda": "2026-05-03T15:40:00",
"itens": [
  {
    "idProduto": "000002",
    "qtde": 2,
    "dataPrevisaoVisita": "2026-06-15",
    "ingressosInfo": [
      {
        "nome": "Cliente info teste 1",
        "email": " Cliente1@email.com",
        "nascimento": "2000-01-01",
        "telefone": "11999999999",
        "documento": "19119119100",
        "tipoDocumento": "CPF"
      },
      {
        "nome": " Cliente info teste 2",
        "email": " Cliente 2@email.com",
        "nascimento": "1992-03-20",
        "telefone": "11988888888",
        "documento": "19119119100",
        "tipoDocumento": "CPF"
      }
    ]
  }
]
```

### Exemplo de retorno

```
{
  "errors": [],
  "result": {
    "id": 457291,
    "terminalVenda": " CLIENTEPWI",
    "usuario": "OPERADOR01",
    "idVendaOrigem": "VENDAINFO1",
    "integrador": 1,
    "cadunico": 2,
    "dataHoraVenda": "2026-05-03T15:40:00",
    "dataHoraCancelado": null,
    "total": 108.00,
    "itens": [
      {
        "id": 541329,
        "idProduto": "000002",
        "descricaoProduto": "INGRESSO INTEIRA INTERNO",
```

```
"qtde": 2,  
"valor": 108.0000,  
"isItemCombo": false,  
"idProdutoCombo": "",  
"descricaoProdutoCombo": null,  
"isInativo": false,  
"fkSessao": 0,  
"ingressos": [  
  {  
    "id": 980329,  
    "numeroItem": 1,  
    "numeroPassaporte": 2480000312167,  
    "digitoPassaporte": 6,  
    "isPassaporte": false,  
    "qtMultiplo": 1,  
    "dataPrevisaoVisita": "2026-06-15T00:00:00",  
    "dataUtilizacao": null,  
    "dataCancelamento": null,  
    "isInativo": false,  
    "codigoVenda": 457291  
  },  
  {  
    "id": 980329,  
    "numeroItem": 1,  
    "numeroPassaporte": 2480000312167,  
    "digitoPassaporte": 6,  
    "isPassaporte": false,  
    "qtMultiplo": 1,  
    "dataPrevisaoVisita": "2026-06-15T00:00:00",  
    "dataUtilizacao": null,  
    "dataCancelamento": null,  
    "isInativo": false,  
    "codigoVenda": 457291  
  },  
  {  
    "id": 980330,  
    "numeroItem": 2,  
    "numeroPassaporte": 1330000312175,  
    "digitoPassaporte": 9,  
    "isPassaporte": false,  
    "qtMultiplo": 1,  
    "dataPrevisaoVisita": "2026-06-15T00:00:00",  
    "dataUtilizacao": null,  
    "dataCancelamento": null,  
    "isInativo": false,  
    "codigoVenda": 457291  
  },  
]
```

```
{
  "id": 980330,
  "numeroItem": 2,
  "numeroPassaporte": 1330000312175,
  "digitoPassaporte": 9,
  "isPassaporte": false,
  "qtMultiplo": 1,
  "dataPrevisaoVisita": "2026-06-15T00:00:00",
  "dataUtilizacao": null,
  "dataCancelamento": null,
  "isInativo": false,
  "codigoVenda": 457291
}
],
"ingressosInfo": null
}
],
"nrDocumentoCliente": "",
"nrCnpj": 0.0,
"nomeCliente": "",
"emailCliente": "",
"telefoneCliente": ""
}
}
```

### Gerar venda de ingressos com controle de sessão diária

Em determinadas situações, as regras referentes às datas disponíveis de um produto são determinadas por um controle de sessão, esse controle é denominado Grupo de Sessão. Nesses casos, a API irá verificar no momento da geração do pedido, se a data e o produto preterido para compra estão disponíveis tanto para a data selecionada como em quantidade de ingressos.

OBS: Verifique com o parque os produtos que possuem em seu cadastro os Grupos de Sessão habilitados.

Nos casos em que a venda do item seja bem-sucedida, a API retornará o campo FK Sessão com o ID da sessão utilizada no item.

```
"id": 294,
"idProduto": "000004",
"qtde": 1,
"valor": 170.00,
"fkSessao": 354,
"ingressos": [
  {
    "id": 479995,
    "numeroItem": 2,
```

```

        "numeroIngresso": 70000000099,
        "digitoIngresso": 0,
        "dataPrevisaoVisita": "2022-11-27T00:00:00"
    }
}
]

```

Nos casos em que o dia da venda não estiver disponível para o Grupo de Sessão do item, a API retornará a seguinte mensagem.

```

"errors": [
  "Sessão não encontrada para a data da compra: 12/11/2023"
],
"result": null

```

E por fim nos casos em que a quantidade de ingressos disponíveis for menor do que a quantidade solicitada pela venda, a API retornará a seguinte mensagem:

```

"errors": [
  "Produto sem saldo de sessão disponível"
],
"result": null

```

## Consulta de Ingressos

Retorna dados necessários para verificar ingressos, também pode ser utilizado para impressão.

### Explicação do método

Retorna uma lista com todos os ingressos gerados a partir de uma venda.

### GET: {url}/api/ingresso/index

Nome	Descrição	Formato	Obrigatório
ID	ID da Venda	Int	Não
dataVisita	Data de visita do ingresso	Date (yyyy-mm-dd)	Não
dataPedido	Data que o pedido foi realizado	Date (yyyy-mm-dd)	Não
dataCatraca	Data que o ingresso passou na catraca	Date (yyyy-mm-dd)	Não

Todos os parâmetros possuem a função de filtrar a tabela de ingressos, de maneira que retorne um resultado mais rápido e preciso.

Ao menos um dos filtros deve estar preenchido, caso contrário receberá o erro indicando:

"É necessário ao menos um filtro para retorno de ingresso"

Exemplos de envio:

- ingresso/index?id=6&dataVisita=2022-11-12&dataPedido=2022-11-11

- ingresso/index?dataCatraca=2022-11-11

### Exemplo de retorno

```
{
  "errors": [],
  "result": [
    {
      "id": 29690,
      "numeroItem": 4,
      "numeroPassaporte": 1350000145035,
      "digitoPassaporte": 1,
      "isPassaporte": false,
      "qtMultiplo": 2,
      "dataPrevisaoVisita": "2024-11-13T00:00:00",
      "dataUtilizacao": null,
      "dataCancelamento": null,
      "isInativo": false,
      "codigoVenda": 47942
    },
    {
      "id": 29692,
      "numeroItem": 1,
      "numeroPassaporte": 1100100100173,
      "digitoPassaporte": 2,
      "isPassaporte": true,
      "qtMultiplo": 2,
      "dataPrevisaoVisita": "2024-11-12T00:00:00",
      "dataUtilizacao": null,
      "dataCancelamento": null,
      "isInativo": false,
      "codigoVenda": 47942
    },
    {
      "id": 29691,
      "numeroItem": 1,
      "numeroPassaporte": 2100100100172,
      "digitoPassaporte": 1,
      "isPassaporte": true,
      "qtMultiplo": 2,
      "dataPrevisaoVisita": "2024-11-12T00:00:00",
      "dataUtilizacao": null,
      "dataCancelamento": null,
      "isInativo": false,
      "codigoVenda": 47942
    }
  ]
}
```

Sobre o retorno:

- **id**: O ID do ingresso, identificador único para cada ingresso.
- **numeroItem**: O número do item do ingresso. Pode ser usado para distinguir diferentes itens em uma venda.
- **numeroPassaporte**: O número do passaporte associado ao ingresso. É um identificador para a pessoa que utilizará o ingresso.
- **digitoPassaporte**: O dígito verificador do número do passaporte. Pode ser usado para validar a integridade do número do passaporte.
- **dataPrevisaoVisita**: A data prevista para a visita. Indica a data em que o ingresso foi planejado para ser utilizado.
- **dataUtilizacao**: A data em que o ingresso foi utilizado. Indica quando o ingresso foi efetivamente utilizado para acessar o local ou serviço associado.
- **dataCancelamento**: A data em que o ingresso foi cancelado. Se o ingresso não foi cancelado, esse campo pode ser nulo.
- **isInativo**: Um indicador booleano que informa se o ingresso está inativo. Se estiver **true** significa que está inativo, ou seja o ingresso não pode mais ser utilizado.
- **codigoVenda**: O código da venda associada ao ingresso. Pode ser útil para rastrear o histórico de vendas e relacionar o ingresso a uma transação de compra.

## Consulta de Ingressos da Venda

Retorna os dados necessários para gerar as impressões dos ingressos a partir do ID da Venda.

### Explicação do método

Retorna uma lista com todos os ingressos gerados a partir de uma venda.

### GET: {url}/api/ingresso/{id}

Nome	Descrição	Formato	Obrigatório
ID	ID da Venda	int	Sim

### Exemplo de retorno

```
{
  "errors": [],
  "result": [
    {
      "id": 29690,
      "numeroItem": 4,
      "numeroPassaporte": 1350000145035,
      "digitoPassaporte": 1,
      "isPassaporte": false,
      "qtMultiplo": 2,
      "dataPrevisaoVisita": "2024-11-13T00:00:00",
      "dataUtilizacao": null,
      "dataCancelamento": null,
      "isInativo": false,
    }
  ]
}
```

```
        "codigoVenda": 47942
    },
    {
        "id": 29692,
        "numeroItem": 1,
        "numeroPassaporte": 1100100100173,
        "digitoPassaporte": 2,
        "isPassaporte": true,
        "qtMultiplo": 2,
        "dataPrevisaoVisita": "2024-11-12T00:00:00",
        "dataUtilizacao": null,
        "dataCancelamento": null,
        "isInativo": false,
        "codigoVenda": 47942
    },
    {
        "id": 29691,
        "numeroItem": 1,
        "numeroPassaporte": 2100100100172,
        "digitoPassaporte": 1,
        "isPassaporte": true,
        "qtMultiplo": 2,
        "dataPrevisaoVisita": "2024-11-12T00:00:00",
        "dataUtilizacao": null,
        "dataCancelamento": null,
        "isInativo": false,
        "codigoVenda": 47942
    }
]
}
```

#### Sobre o retorno:

- id: O ID do ingresso, identificador único para cada ingresso.
- numeroItem: O número do item do ingresso. Pode ser usado para distinguir diferentes itens em uma venda.
- numeroPassaporte: O número do passaporte associado ao ingresso. É um identificador para a pessoa que utilizará o ingresso.
- digitoPassaporte: O dígito verificador do número do passaporte. Pode ser usado para validar a integridade do número do passaporte.
- dataPrevisaoVisita: A data prevista para a visita. Indica a data em que o ingresso foi planejado para ser utilizado.
- dataUtilizacao: A data em que o ingresso foi utilizado. Indica quando o ingresso foi efetivamente utilizado para acessar o local ou serviço associado.
- dataCancelamento: A data em que o ingresso foi cancelado. Se o ingresso não foi cancelado, esse campo pode ser nulo.
- isInativo: Um indicador booleano que informa se o ingresso está inativo. Se estiver **true** significa que está inativo, ou seja o ingresso não pode mais ser utilizado.

- **codigoVenda:** O código da venda associada ao ingresso. Pode ser útil para rastrear o histórico de vendas e relacionar o ingresso a uma transação de compra.

## Consulta de lista de vendas

Retorna os dados completos de vendas paginadas especificadas nos parâmetros da requisição.

### Explicação do método

Retorna uma lista de vendas dentro do intervalo de datas informadas via parâmetro.

**GET:** `{url}/api/venda/lista/{dataInicio}/{DataFim}/{pag}`

Nome	Descrição	Formato	Obrigatório
dataInicio	Data inicial da venda na consulta	YYYYMMDD	Sim
dataFim	Data final da venda na consulta	YYYYMMDD	Sim
Pag	Número da página para a consulta	Int	Opcional

### Exemplo de Retorno

```
{
  "errors": [],
  "result": {
    "paginacao": {
      "pagina": 1,
      "totalPaginas": 1
    },
    "pedidos": [
      {
        "id": 47732,
        "terminalVenda": " Terminal_05",
        "usuario": "PWI",
        "idVendaOrigem": "pwi_avelar_32",
        "integrador": 1,
        "cadunico": 2,
        "dataHoraVenda": "2024-10-23T10:00:00",
        "dataHoraCancelado": null,
        "total": 307.9,
        "itens": [
          {
            "id": 32804,
            "idProduto": "20240320",
            "qtde": 1,
            "valor": 59.9000,
            "isItemCombo": false,
            "idProdutoCombo": "",
            "isInativo": false,
            "fkSessao": 50197,

```

```
"ingressos": [  
  {  
    "id": 29690,  
    "numeroItem": 4,  
    "numeroPassaporte": 1350000145035,  
    "digitoPassaporte": 1,  
    "isPassaporte": false,  
    "qtMultiplo": 2,  
    "dataPrevisaoVisita": "2024-11-13T00:00:00",  
    "dataUtilizacao": null,  
    "dataCancelamento": null,  
    "isInativo": false,  
    "codigoVenda": 47942  
  }  
],  
{  
  "id": 32805,  
  "idProduto": "2024101802",  
  "qtde": 2,  
  "valor": 200.0000,  
  "isItemCombo": true,  
  "idProdutoCombo": "2024101801",  
  "isInativo": false,  
  "fkSessao": 0,  
  "ingressos": [  
    {  
      "id": 29692,  
      "numeroItem": 1,  
      "numeroPassaporte": 1100100100173,  
      "digitoPassaporte": 2,  
      "isPassaporte": true,  
      "qtMultiplo": 2,  
      "dataPrevisaoVisita": "2024-11-12T00:00:00",  
      "dataUtilizacao": null,  
      "dataCancelamento": null,  
      "isInativo": false,  
      "codigoVenda": 47942  
    },  
    {  
      "id": 29691,  
      "numeroItem": 1,  
      "numeroPassaporte": 2100100100172,  
      "digitoPassaporte": 1,  
      "isPassaporte": true,  
      "qtMultiplo": 2,  
      "dataPrevisaoVisita": "2024-11-12T00:00:00",  
      "dataUtilizacao": null,  
    }  
  ]  
}
```

```
        "dataCancelamento": null,  
        "isInativo": false,  
        "codigoVenda": 47942  
    }  
  ],  
},  
{  
  "id": 32806,  
  "idProduto": "2024101803",  
  "qtde": 2,  
  "valor": 48.0000,  
  "isItemCombo": true,  
  "idProdutoCombo": "2024101801",  
  "isInativo": false,  
  "fkSessao": 0,  
  "ingressos": []  
},  
"nrDocumentoCliente": "19119119155",  
"nomeCliente": "JOSE DA SILVA",  
"emailCliente": "josedasilva@meuemail.com.br",  
"telefoneCliente": "11988457720",  
},  
{  
  "id": 47731,  
  "terminalVenda": " Terminal_05",  
  "usuario": "PWI",  
  "idVendaOrigem": "pwi_avelar_31",  
  "integrador": 1,  
  "cadunico": 2,  
  "dataHoraVenda": "2024-10-23T10:00:00",  
  "dataHoraCancelado": null,  
  "total": 443.20,  
  "itens": [  
    {  
      "id": 32565,  
      "idProduto": "TESTE1",  
      "qtde": 2,  
      "valor": 443.2000,  
      "isInativo": false,  
      "fkSessao": 0,  
      "ingressos": [  
        {  
          "id": 29646,  
          "numeroItem": 4,  
          "numeroPassaporte": 6590000144599,  
          "digitoPassaporte": 6,  
          "isPassaporte": true,  
        }  
      ]  
    }  
  ]  
}
```

```

        "dataPrevisaoVisita": "2024-11-12T00:00:00",
        "dataUtilizacao": null,
        "dataCancelamento": null,
        "isInativo": false,
        "codigoVenda": 47732
    }
  ]
},
"nrDocumentoCliente": "19119119155",
"nomeCliente": "FELIPE AVELAR",
"emailCliente": "josedasilva@meuemail.com.br",
"telefoneCliente": "11955487762"
}
]
}

```

### Consulta de Venda por ID

Retorna os dados de uma venda específica através de seu ID.

**GET:** {url}/api/venda/{id}

Nome	Descrição	Formato	Obrigatório
ID	ID da Venda	int	Sim

### Exemplo de retorno

```

{
  "errors": [],
  "result": {
    "id": 47732,
    "terminalVenda": "TERMINAL_02",
    "usuario": "PWI",
    "idVendaOrigem": "pwi_avelar_32",
    "integrador": 1,
    "cadunico": 2,
    "dataHoraVenda": "2024-10-23T10:00:00",
    "dataHoraCancelado": null,
    "total": 307.90,
    "itens": [
      {
        "id": 32804,
        "idProduto": "20240320",
        "qtde": 1,
        "valor": 59.9000,
        "isItemCombo": false,
        "idProdutoCombo": ""
      }
    ]
  }
}

```

```
"isInativo": false,
"fkSessao": 50197,
"ingressos": [
  {
    "id": 29690,
    "numeroItem": 4,
    "numeroPassaporte": 1350000145035,
    "digitoPassaporte": 1,
    "isPassaporte": false,
    "qtMultiplo": 2,
    "dataPrevisaoVisita": "2024-11-13T00:00:00",
    "dataUtilizacao": null,
    "dataCancelamento": null,
    "isInativo": false,
    "codigoVenda": 47942
  }
],
{
  "id": 32805,
  "idProduto": "2024101802",
  "qtde": 2,
  "valor": 200.0000,
  "isItemCombo": true,
  "idProdutoCombo": "2024101801",
  "isInativo": false,
  "fkSessao": 0,
  "ingressos": [
    {
      "id": 29692,
      "numeroItem": 1,
      "numeroPassaporte": 1100100100173,
      "digitoPassaporte": 2,
      "isPassaporte": true,
      "qtMultiplo": 2,
      "dataPrevisaoVisita": "2024-11-12T00:00:00",
      "dataUtilizacao": null,
      "dataCancelamento": null,
      "isInativo": false,
      "codigoVenda": 47942
    },
    {
      "id": 29691,
      "numeroItem": 1,
      "numeroPassaporte": 2100100100172,
      "digitoPassaporte": 1,
      "isPassaporte": true,
      "qtMultiplo": 2,
```

```

        "dataPrevisaoVisita": "2024-11-12T00:00:00",
        "dataUtilizacao": null,
        "dataCancelamento": null,
        "isInativo": false,
        "codigoVenda": 47942
    }
  ],
  {
    "id": 32806,
    "idProduto": "2024101803",
    "qtde": 2,
    "valor": 48.0000,
    "isItemCombo": true,
    "idProdutoCombo": "2024101801",
    "isInativo": false,
    "fkSessao": 0,
    "ingressos": []
  }
],
"nrDocumentoCliente": "19119119155",
"nomeCliente": "FELIPE AVELAR",
"emailCliente": "josedasilva@meuemail.com.br",
"telefoneCliente": "11955487762"
}
}

```

## Cancelar venda e ingressos gerados

Responsável por cancelar uma venda e seus ingressos. Verifique com o administrador do parque as regras de cancelamento.

Uma vez cancelado, o pedido não poderá voltar ao status anterior de ativo. Os ingressos são cancelados e ficarão impossibilitados de serem validados na catraca do parque.

Apenas pedidos gerados via API, poderão ser cancelados por esse método.

### Explicação do método

Ao informar um ID válido de venda, caso o pedido esteja apto a ser cancelado, assim será feito com o cancelamento de seus ingressos.

### PUT: {url}/api/venda/cancelar/{id}

Nome	Descrição	Formato	Obrigatório
Id	Id da venda	int	Sim

## Exemplo de retorno

### Sucesso

```
{
  "errors": [],
  "result": {
    "codigo": 156,
    "mensagem": "CANCELAMENTO EFETUADO COM SUCESSO."
  }
}
```

### Falha

```
{
  "errors": [],
  "result": {
    "codigo": -1,
    "mensagem": "INGRESSO NÃO LOCALIZADO."
  }
}
```

## Lista de códigos de retorno

Código	Descrição
1	Cancelamento efetuado com sucesso.
-1	Ingresso não localizado.
-2	Tipo de cancelamento inválido.
-3	Ingresso já cancelado.
-4	Ingresso já utilizado.
-5	Ingresso não cancelado: Geração não realizada através da API.
-6	Integrador não encontrado.
-7	Data permitida para o cancelamento ultrapassada.

## E-Ticket (Ingresso)

A geração do e-ticket (Ingresso) poderá ser elaborada com qualquer layout definido pelo integrador desde contenha os seguintes dados:

- 1) Identificação que os ingressos são referentes ao parque.
- 2) Descrição e quantidade dos ingressos adquiridos.
- 3) Impressão de código de barras e/ou Impressão QR Code(verifique com o administrador do parque qual código deverá ser apresentado).
- 4) O código de barras gerado deverá estar próximo a descrição do ingresso para facilitar a leitura na catraca.

### Código de barras / QR Code do E-Ticket (Ingresso)

O código de barras gerado para cada ingresso deverá seguir o padrão "2 de 5 intercalado" e será composto pelo número do ingresso gerado mais o dígito verificador.

O QR Code será composto pelo número do ingresso gerado mais o dígito verificador. Será aceito apenas campos numéricos na geração do QR Code.

Exemplo:

Nro. Ingresso: 12341234

Digito Verificador: 9

Código de barras / QR Code: 123412349



## Roteiro de Homologação

Criado para auxiliar a integração do vendedor PJ com o parque. Tem a função de garantir que o consumo da API foi implementado corretamente no software do vendedor. Recomendamos que salve os envios e retornos para que possa auxiliar na verificação do passo-a-passo.

### 1 - Autenticação

Método Login, que recebe como parâmetro usuário e senha, fornecido pelo parque.

**POST:** {url}/api/auth/login

**Token:** {vazio}

**Body:**

```
{
  "USUARIO": "CNPJ",
  "SENHA": "Senha"
}
```

O retorno deve conter a seguinte informação:

- Access token

## 2- Consulta de produtos disponíveis para venda

Consulta os produtos que estão disponíveis para o Integrador.

**GET:** {url}/api/produto/lista

**Parâmetros:** idProduto (opcional), dataVisita (opcional)

**Authorization:** Bearer Token (Access Token)

**Body:** {vazio}

O retorno deve conter as seguintes informações:

- ID do produto
- Nome do produto
- Preço do produto
- Se data de visita é obrigatória
- Se produto possui uma sessão
- Data de validade inicial do produto
- Data de validade final do produto
- Quantidade de dias de validade

## 3- Consulta de sessões disponíveis para um determinado produto

Consulta os produtos que estão disponíveis para a venda dentro de um determinado período.

**GET:** {url}/api/produto/listaSessoes

**Parâmetros:** idProduto, dataInicio, dataFim

**Authorization:** Bearer Token (Access Token)

**Body:** {vazio}

O retorno deve conter as seguintes informações:

- ID da sessão
- Data de hora da sessão
- Quantidade de vagas disponíveis na sessão

## 4 - Consulta de saldo da conta consumo disponível

Consulta o saldo da conta consumo do Integrador, se disponível.

**GET:** {url}/api/creditos/saldo

**Authorization:** Bearer Token (Access Token)

**Body:** {vazio}

O retorno deve conter a seguinte informação:

- Saldo disponível

## 5 - Gerar vendas e ingressos

Método responsável por gerar vendas e ingressos.

**POST:** {url}/api/venda/incluir

**Authorization:** Bearer Token (Access Token)

**Body:**

```
{
  "TerminalVenda": "Terminal",
  "Usuario": "Usuario",
  "IdVendaOrigem": "ID Venda",
  "DataHoraVenda": "DataHora Venda",
  "Itens": [
    {
      "IdProduto": "ID Produto",
      "DataPrevisaoVisita": "DataHora Previsão Visita",
      "Qtde": Quantidade
    },
    {
      "IdProduto": "ID Produto",
      "DataPrevisaoVisita": "DataHora Previsão Visita",
      "Qtde": Quantidade
    }
  ]
}
```

O retorno deve conter as seguintes informações:

- ID Venda
- Terminal Venda
- Usuario
- IdVendaOrigem
- Integrador
- Cadunico

- Data e hora da venda
- Data e hora do cancelamento
- Valor total
- Itens da Venda(conjunto)
  - ID Item
  - ID Produto
  - Quantidade Produto
  - Valor do produto
  - Sessão do produto (Pode ou não estar preenchido)
  - Ingressos
    - ID Ingresso
    - Número Item
    - Número do Ingresso
    - Dígito do Ingresso
    - Data de previsão da visita

## 5.1 - Gerar vendas e ingressos com sessão

No mesmo método utilizado anteriormente, caso a venda contenha um item com limitação diária de venda, será consultado a sessão por data da visita do item para incluir a venda com sucesso.

O retorno deve conter as seguintes informações:

- ID Venda
- Terminal Venda
- Usuario
- IdVendaOrigem
- Integrador
- Cadunico
- Data e hora da venda
- Data e hora do cancelamento
- Valor total
- Itens da Venda(conjunto)
  - ID Item
  - ID Produto
  - Quantidade Produto
  - Valor do produto
  - Sessão do produto (preenchido)
  - Ingressos
    - ID Ingresso
    - Número Item
    - Número do Ingresso
    - Dígito do Ingresso
    - Data de previsão da visita

## 6 - Consulta de ingressos

Método responsável por consultar os ingressos de uma venda.

**GET:** {url}/api/ingresso/{id}

**Authorization:** Bearer Token (Access Token)

**Body:**{vazio}

O retorno deve conter as seguintes informações:

- ID ingresso
- Número do item
- Número do ingresso
- Dígito do ingresso
- Data previsão de visita

## 7 - Consulta lista de vendas

Método responsável por retornar uma lista de vendas filtradas por data.

**GET:** {url}/api/venda/lista/{dataInicio}/{DataFim}/{pag}

**Authorization:** Bearer Token (Access Token)

**Body:** {vazio}

O retorno deve conter as seguintes informações:

- Paginação
  - Página
  - Total Páginas
- Vendas (conjunto)
  - ID Venda
  - Terminal Venda
  - Usuario
  - IdVendaOrigem
  - Integrador
  - Cadunico
  - Data e hora da venda
  - Data e hora do cancelamento

- Valor total
- Itens da Venda(conjunto)
  1. ID Item
  2. ID Produto
  3. Quantidade Produto
  4. Valor do produto
  5. Sessão do produto
  6. Ingressos
    1. ID Ingresso
    2. Número Item
    3. Número do Ingresso
    4. Dígito do Ingresso
    5. Data de previsão da visita

## 8 - Consulta de venda por ID

Método que retorna os dados de uma venda através de seu ID.

**GET:** {url}/api/venda/{id}

**Authorization:** Bearer Token (Acess Token)

**Body:** {vazio}

O retorno deve conter as seguintes informações:

- ID Venda
- Terminal Venda
- Usuario
- IdVendaOrigem
- Integrador
- Cadunico
- Data e hora da venda
- Data e hora do cancelamento
- Valor total
- Itens da Venda(conjunto)
  - ID Item
  - ID Produto
  - Quantidade Produto
  - Valor do produto
  - Sessão do produto
  - Ingressos
    - ID Ingresso

- Número Item
- Número do Ingresso
- Dígito do Ingresso
- Data de previsão da visita

## 9 - Cancelar venda e ingressos gerados

Método responsável por cancelar uma venda e seus ingressos.

**PUT:** {url}/api/venda/cancelar/{id}

**Authorization:** Bearer Token (Access Token)

**Body:** {vazio}

O retorno deve conter as seguintes informações:

- Código
- Mensagem

## 10 - Logout [obsoleto]

Método responsável temporariamente por limpar o token do usuário quando houver o erro de header very much long

**Post:** {url}/api/auth/logout/

**Body:**

```
{
  "USUARIO": "CNPJ",
  "SENHA": "Senha"
}
```

O retorno deve conter status code de 200 (OK)

## Log de alterações

Versão	Data	Observações
1.0.0	17/01/2020	Criação inicial do projeto
1.0.1	08/01/2021	Ajuste na url de acesso
1.0.2	10/11/2022	Reformulação do documento
1.0.3	29/11/2022	Nova rota logout
1.0.4	16/12/2022	Adicionando Validações feitas pela API

1.0.5	05/01/2023	Melhorando texto dos parâmetros do método de consulta de lista de vendas
1.0.6	06/01/2023	Adicionando venda de ingressos com sessão
1.0.7	08/02/2023	Correção no método de logout de Put para Post
1.0.8	17/02/2023	Adicionado ambiente interno PWI
1.0.9	23/02/2023	Alterando ambiente interno PWI
1.0.10	03/04/2023	Adicionado novos parâmetros para o filtro de listagem de produtos Adicionado novos campos de retorno na listagem de produtos Melhoria nas validações de saldo e integrador na venda
1.0.11	15/05/2023	Adicionando campos para dados do cliente/comprador na venda.
1.0.12	22/05/2023	Adicionado nova rota de ingressos Adicionado nova validação na venda em relação a data da venda (somente doc.)
1.0.13	05/06/2023	Adicionado novos campos no retorno de ingressos: dataCancelamento, isInativo, codigoVenda
1.0.14	29/12/2023	Controle de versões
1.0.15	20/03/2024	Incluindo rota de consulta das sessões disponíveis para um determinado produto e período
1.0.16	01/07/2024	Ajuste na validação da data da venda para ignorar os segundos ao incluir uma venda. Atualização das bibliotecas para melhorar o desempenho e a segurança da API.
1.0.17	28/10/2024	Novas validações na restrição de produtos impactando na listar produtos e na inclusão de produtos. Incluindo novo campo no retorno dos itens da venda isInativo, nas rotas: Inclusão de venda, buscar venda por id, listar vendas. A Api agora passa a incluir financeiros. Melhorias na tabela de pedidos, movimentos, conta consumo.
1.0.18	11/11/2024	Inclusão da possibilidade de retornar passaportes na API de vendas; Melhorias no retorno de Ingresso nas rotas: <ul style="list-style-type: none"> <li>• Consulta de lista de vendas (venda/lista/)</li> <li>• Consulta de Venda por ID (venda/{id})</li> </ul> Inclusão de isPassaporte no objeto de Ingresso nas rotas: <ul style="list-style-type: none"> <li>• Gerar venda e ingressos (venda/incluir)</li> <li>• Consulta de lista de vendas (venda/lista/)</li> <li>• Consulta de Venda por ID (venda/{id})</li> <li>• Consulta de Ingressos (ingresso/index)</li> <li>• Consulta de Ingressos da Venda (ingresso/{id})</li> </ul>
1.0.19	12/11/2024	A API passou a trabalhar com combos, explodindo os itens (movimentos) ao finalizar o pedido. Alteração no retorno das rotas de vendas, adicionados os campos: <ul style="list-style-type: none"> <li>• Itens: <ul style="list-style-type: none"> <li>○ IsItemCombo</li> </ul> </li> </ul>

		<ul style="list-style-type: none"><li>○ IdProdutoCombo</li><li>● Ingressos:<ul style="list-style-type: none"><li>○ QtMultiplo</li></ul></li></ul> Mudanças no retorno nas rotas de ingressos, adicionado o campo: <ul style="list-style-type: none"><li>○ QtMultiplo</li></ul>
<b>1.0.20</b>	26/11/2024	Melhorias nas rotas de vendas e ingressos: <ul style="list-style-type: none"><li>● Melhoria na segurança e desempenho da busca de vendas.</li></ul>
<b>1.0.21</b>	24/01/2025	Acrescentando na rota de Creditos/Saldo, a consulta do saldo em dinheiro. Api tem a possibilidade de multiempresas