PWI
SISTEMAS

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: equipe.n1@pwi.com.br
VERSÃO DOCUMENTO:	Versão: 1.0.16

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 Baerer 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 8
Validações: 9
Exemplo de retorno 9
Consulta de saldo da conta consumo disponível 10
Explicação do método 10
Exemplo de retorno 10
Gerar venda e ingressos 10
Validações feitas pela API 10
Parâmetros de envio 11
Exemplo de envio 12
Exemplo de retorno 12
Gerar venda de ingressos com controle de sessão diária 13
Consulta de Ingressos 14
Explicação do método 14
Exemplo de retorno 15
Consulta de Ingressos da Venda 16
Explicação do método 16

Exemplo de retorno 16
Consulta de lista de vendas 17
Explicação do método 17
Exemplo de Retorno 18
Consulta de Venda por ID 18
Exemplo de retorno 19
Cancelar venda e ingressos gerados 20
Explicação do método 20
Exemplo de retorno 20
Sucesso 20
Falha 20
Lista de códigos de retorno 21
E-Ticket (Ingresso) 21
Código de barras / QR Code do E-Ticket (Ingresso) 21
..... 22
Roteiro de Homologação 22
1 - Autenticação 22
2- Consulta de produtos disponíveis para venda 22
3- Consulta de sessões disponíveis para um determinado produto 23
4 - Consulta de saldo da conta consumo disponível 23
5 - Gerar vendas e ingressos 23
5.1 - Gerar vendas e ingressos com sessão 25
6 - Consulta de ingressos 25
7 - Consulta lista de vendas 26
8 - Consulta de venda por ID 27
9 - Cancelar venda e ingressos gerados 27
10 – Logout [Obsoleto] 28
Log de alterações 28

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	https://api.{cliente}.com.br/prod/
HOMOLOGAÇÃO	https://api.{cliente}.com.br/hom/

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	apivendas.pwi.com.br/prod/

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

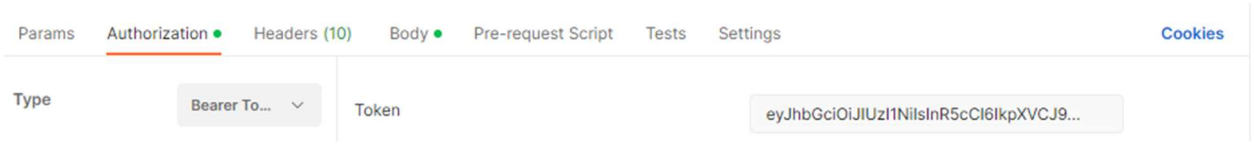
```
{
       "errors": [],
       "result": {
              "access_token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...",
              "inativo": false,
              "ApiVersion" : documentaçãoVersao 1.0.10

    }
}
```

Autorização Baerer 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”.



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 possem 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.

Exemplo de retorno


```
{
	"errors": [],
	"result": [
	{
		"id": "000004",
		"nome": "INGRESSO ADULTO",
		"preco": 170.000000,
		"visitaObrigatorio": false,
		"produtoComSessao": false,
		"dataValidadeInicio": null,
		"dataValidadeTermino": null,
		"qtDiasValidade": 5
	},
	{
		"id": "000004",
		"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/listaSessoes

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 da conta consumo disponível

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.

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 possuí 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.
- Valida se permite o uso da Conta Consumo;
- Valida se o saldo da Conta Consumo é insuficiente.

Parâmetros de envio

Nome	Descrição	Formato	Obrigatório
TerminalVenda	Nome do terminal de envio da requisição	String (20)	Sim
Usuario	Nome do usuário que enviou a requisição	String (30)	Sim
IdVendaOrigem	Seu ID de referência da venda	String (10)	Sim
DataHoraVenda	Data e hora da venda	Date	Sim
NrDocumentoCliente	Documento (CPF/RG) do cliente	String (20)	Não
NomeCliente	Nome do cliente	String (100)	Não
EmailCliente	E-mail do cliente	String (150)	Não
TelefoneCLiente	Telefone do cliente	String (20)	Não
Itens	Informações de itens	Array	Sim

IdProduto	Código do produto	String (15)	Sim
Qtde	Quantidade do item especificado da venda	int	Sim
DataPrevisaoVisita	Data de previsão de visita. A data de previsão de visita só deve ser preenchida se no item a mesma for obrigatória.	Datetime	Variável

	Qualquer outro caso dever ser passado com o valor nulo.		Definida de acordo com o item. Sempre verificar na rota de listagem de produtos.

Exemplo de envio

```
{
    "TerminalVenda": "Maq@Pwi",
    "Usuario": "Gustavo Pereira",
    "IdVendaOrigem": "15",
    "DataHoraVenda": "2022-11-08T13:51:27",
    "NrDocumentoCliente": "39409489857",
    "NomeCliente": "FELIPE AVELAR",
    "EmailCliente": "felipe.avelar@pwi.com.br",
    "TelefoneCliente": "19992062026",
    "Itens": [
        {
            "IdProduto": "000004",
            "DataPrevisaoVisita": "2022-11-08T00:00:00",
            "Qtde": 1
        },
        {
            "IdProduto": "000003",
            "Qtde": 1
        }
    ]
}
```

Exemplo de retorno

```
{
    "errors": [],
    "result": {
        "id": 156,
        "terminalVenda": " Maq@Pwi ",
        "usuario": "Gustavo Pereira",
        "idVendaOrigem": "15",
        "integrador": 2,
        "cadunico": 69,
        "dataHoraVenda": "2022-11-08T13:51:27",
        "dataHoraCancelado": null,
        "total": 340.000000,
```

```
        "itens": [
            {
                "id": 293,
                "idProduto": "000004",
                "qtde": 1,
                "valor": 170.00,
                "fkSessao": null,
                "ingressos": [
                    {
                        "id": 479994,
                        "numeroItem": 1,
                        "numeroIngresso": 70000000098,
                        "digitoIngresso": 8,
                        "dataPrevisaoVisita": "2022-11-08T00:00:00"
                    }
                ]
            },
            {
                "id": 294,
                "idProduto": "000004",
                "qtde": 1,
                "valor": 170.00,
                "fkSessao": null,
                "ingressos": [
                    {
                        "id": 479995,
                        "numeroItem": 2,
                        "numeroIngresso": 70000000099,
                        "digitoIngresso": 0
                    }
                ]
            }
        ]
    }
}
```


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 possem 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": 6114,
            "numeroItem": 1,
            "numeroPassaporte": 20000002007,
            "digitoPassaporte": 7,
            "dataPrevisaoVisita": "2022-11-12T00:00:00",
            "dataUtilizacao": "2022-11-08T10:12:19",
            "dataCancelamento": null,
            "isInativo": false,
            "codigoVenda": 11
        },
        {
            "id": 6126,
            "numeroItem": 1,
            "numeroPassaporte": 20000002019,
            "digitoPassaporte": 2,
            "dataPrevisaoVisita": "2022-11-12T00:00:00",
            "dataUtilizacao": "2022-11-08T10:16:18",
            "dataCancelamento": "2022-11-08T10:16:24.633",
            "isInativo": true,
            "codigoVenda": 14
        }
    ]
}
```


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": 6126,
            "numeroItem": 1,
            "numeroPassaporte": 20000002019,
            "digitoPassaporte": 2,
            "dataPrevisaoVisita": "2022-11-12T00:00:00",
            "dataUtilizacao": "2022-11-08T10:16:18",
            "dataCancelamento": "2022-11-08T10:16:24.633",
            "isInativo": true,
            "codigoVenda": 14
        },
```

```
        {
            "id": 6127,
            "numeroItem": 2,
            "numeroPassaporte": 20000002020,
            "digitoPassaporte": 7,
            "dataPrevisaoVisita": "2022-11-11T00:00:00",
            "dataUtilizacao": null,
            "dataCancelamento": "2022-11-08T10:16:24.633",
            "isInativo": true,
            "codigoVenda": 14
        }
    ]
}
```


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
        },
        "vendas": [
            {
                "id": 156,
                "terminalVenda": "MAQ@PWI",
                "usuario": "Gustavo Pereira",
                "idVendaOrigem": "15",
                "integrador": 2,
                "cadunico": 69,
                "dataHoraVenda": "2022-11-08T13:51:00",
                "dataHoraCancelado": null,
                "total": 340.00,
                "itens": [
                    {
                        "id": 293,
                        "idProduto": "000004",
                        "qtde": 1,
                        "valor": 170.00,
                        "fkSessao": null,
                        "ingressos": [
                            {
                                "id": 479994,
                                "numeroItem": 1,
                                "numeroIngresso": 70000000098,
                                "digitoIngresso": 8,
                                "dataPrevisaoVisita": "2022-11-08T00:00:00"
                            }
                        ]
                    }
                ]
            }
        ]
    }
}
```


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": 156,
        "terminalVenda": "Maq@Pwi",
        "usuario": "Gustavo Pereira",
        "idVendaOrigem": "15",
        "integrador": 2,
        "cadunico": 69,
        "dataHoraVenda": "2022-11-08T13:51:00",
        "dataHoraCancelado": null,
        "total": 340.00,
        "itens": [
            {
                "id": 293,
                "idProduto": "000004",
                "qtde": 1,
                "valor": 170.00,
                "fkSessao": null,
                "ingressos": [
                    {
                        "id": 479994,
                        "numeroItem": 1,
                        "numeroIngresso": 70000000098,
                        "digitoIngresso": 8,
                        "dataPrevisaoVisita": "2022-11-08T00:00:00"
                    }
                ]
            },
            {
                "id": 294,
                "idProduto": "000004",
                "qtde": 1,
                "valor": 170.00,
                "fkSessao": null,
                "ingressos": [
                    {
                        "id": 479995,
                        "numeroItem": 2,
                        "numeroIngresso": 70000000099,
                        "digitoIngresso": 0,
```

```
                        "dataPrevisaoVisita": "2022-11-27T00:00:00"
                    }
                ]
            }
        ]
    }
}
```


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:

- Acess 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 (Acess 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 (Acess 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 (Acess 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 (Acess 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
        - Digito 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
        - Digito 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 (Acess Token)

Body:{vazio}

O retorno deve conter as seguintes informações:

- ID ingresso
- Número do item
- Número do ingresso

- Digito 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 (Acess 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. Digito 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
        - Digito do Ingresso
        - Data de previsão da visita

9 - Cancelar venda e ingressos gerados

Método responsável por canelar uma venda e seus ingressos.

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

Authorization: Bearer Token (Acess 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 hearder 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.