Integração Servidor Comandas

Documento de integração para utilizar as API's do servidor comandas.

- API para buscar comanda a ser faturada.
- API para liberar as comandas após finalizar a venda.
- API para voltar o status da comanda para consumo após abortar operação.
- Requisitos Centralizador PDV's Fast

API para buscar comanda a ser faturada.

□□ API – Iniciar Faturamento de Comandas

Endpoint:
`POST /api/v1/mesas-comandas/iniciar-faturamento`

□□ Objetivo da API

Esta API permite que um **PDV** solicite ao **Servidor de Comandas** o início do faturamento de **uma ou mais comandas**, possibilitando que elas sejam pagas juntas em uma única venda.

O Servidor de Comandas:

- valida cada comanda solicitada;
- **trava** (coloca em `conferencia`) apenas as comandas que podem ser faturadas;
- retorna os **itens da comanda** para o PDV montar a venda;
- retorna também as comandas que **não puderam ser faturadas**, com o motivo.

⚠ **Importante:**
A API permite **sucesso parcial**.
Se 3 comandas forem enviadas e apenas 2 puderem ser faturadas, **essas 2 serão liberadas para faturamento** e a outra será retornada como bloqueada.

□□ Conceitos importantes (antes de integrar)

Status da comanda

Status	Significado
`consumo`	Comanda em uso, com itens lançados
`conferencia`	Comanda travada para faturamento em um PDV
`livre`	Comanda sem itens ou já finalizada
`transferencia`	Comanda em processo de transferência

Regras gerais

- **Nunca existe faturamento parcial de itens**: ou a comanda entra inteira ou não entra.
- **Itens cancelados sempre são retornados**, mas **não entram no total**.
- O PDV **não recalcula valores**: deve confiar nos valores retornados pela API.

□□ Autenticação

Segue o **padrão de autenticação do Servidor de Comandas** (definido pelo projeto).

> ❝ Caso a autenticação falhe, a API retornará erro HTTP (401/403).

□□ Request

Body (JSON)

Campo	Tipo	Obrigatório	Descrição
`numeroMesas`	`number[]`	✅	IDs das comandas que o PDV deseja faturar
`numeroPdv`	`string`	✅	Identificação do PDV solicitante
`operador_id`	`string`	✅	Identificação do operador

Exemplo de requisição

```
{
  "numeroMesas": [4069, 4070, 4075],
  "numeroPdv": "12",
  "operador_id": "1275"
}
```

⚙ O que o servidor faz ao receber a requisição

Para **cada comanda enviada**, o servidor avalia:

Situação da comanda	Resultado
Não existe	❌ Bloqueada (`NAO_ENCONTRADA`)
Já está em conferência	❌ Bloqueada (`EM_CONFERENCIA`)
Está livre	❌ Bloqueada (`STATUS_LIVRE`)
Está em consumo e há PDVs lançando itens	❌ Bloqueada (`EM_LANCAMENTO_ITENS`)
Está em transferência	❌ Bloqueada (`EM_TRANSFERENCIA`)
Está em consumo (sem bloqueios)	✅ Elegível → status muda para `conferencia`

Response – Sucesso (200)

Estrutura do retorno

```
{
  "comandasParaFaturar": [],
  "comandasBloqueadasParaFaturar": []
}
```

🧾 `comandasParaFaturar`

Lista das comandas que **foram liberadas para faturamento**.

Campos

Campo	Descrição
`idMesa`	ID da comanda
`codigoComanda`	Código exibido ao operador
`statusAlteradoPara`	Sempre `conferencia`
`totalComanda`	Soma dos itens **não cancelados**
`itens`	Lista de itens da comanda

Regras do total

- `totalComanda` = soma de `preco_total` **somente dos itens com** `cancelado = false`

Estrutura de um item

```
{
  "id": 34,
  "usuario_lancamento": "1275895",
  "produto_id": 9381649,
  "ean": "00000000100700",
  "descricao_resumida": "PROD TESTE 01",
  "quantidade": 1,
  "valorTotalAcrescimo": 10,
  "valorTotalDesconto": 0,
  "cancelado": false,
  "cancelado_por": null,
  "preco_unitario": 15,
  "preco_total": 25,
  "data_hora": "27/01/2026, 11:03:12"
}
```


⚠ Itens cancelados

- `cancelado = true`
- **Não entram no total**
- Servem apenas para **auditoria/relatórios**
- `preco_total` é apenas informativo

□□ `comandasBloqueadasParaFaturar`

Lista das comandas que **não puderam ser faturadas**.

Campos

Campo	Descrição
`idMesa`	ID da comanda (quando existir)
`codigoComanda`	Código da comanda
`statusAtual`	Status atual da comanda
`motivo`	Código do bloqueio

Campo	Descrição
`mensagem`	Texto explicativo
`numeroPdvEmConferencia`	PDV que está usando a comanda (quando aplicável)
`pos_lancando`	PDVs lançando itens (quando aplicável)

□□ Enum `motivo`

Motivo	Significado
`EM_CONFERENCIA`	Comanda já está sendo faturada
`STATUS_LIVRE`	Comanda está livre/sem itens
`NAO_ENCONTRADA`	Comanda inexistente
`EM_LANCAMENTO_ITENS`	Existem PDVs lançando itens
`EM_TRANSFERENCIA`	Comanda em transferência
`STATUS_INVALIDO`	Status não permitido
`SEM_ITENS`	Comanda sem itens (quando aplicável)

Response – Sucesso (200) contendo apenas comandas aptas a faturar.


```
{
 "comandasParaFaturar": [
   {
     "idMesa": 4069,
     "codigoComanda": "21",
     "statusAlteradoPara": "conferencia",
      "totalComanda": 31.50,
     "itens": [
```

```
{
  "id": 34,
  "usuario_lancamento": "1275895",
  "produto_id": 9381649,
  "ean": "00000000100700",
  "descricao_resumida": "PROD TESTE 01",
  "quantidade": 1,
  "valorTotalAcrescimo": 10,
  "valorTotalDesconto": 0,
  "cancelado": false,
  "cancelado_por": null,
  "preco_unitario": 15,
  "preco_total": 25,
  "data_hora": "27/01/2026, 11:03:12"

},
{
  "id": 35,
  "usuario_lancamento": "1275895",
  "produto_id": 9381647,
  "ean": "00000000100670",
  "descricao_resumida": "PROD TESTE 40",
  "quantidade": 1,
  "valorTotalAcrescimo": 0,
  "valorTotalDesconto": 0,
  "cancelado": false,
  "cancelado_por": null,
  "preco_unitario": 6.5,
  "preco_total": 6.5,
  "data_hora": "27/01/2026, 11:03:12"
},
{
  "id": 36,
  "usuario_lancamento": "1275895",
  "produto_id": 8104916,
  "ean": "00000001267532",
  "descricao_resumida": "10 OVO",
  "quantidade": 1,
  "valorTotalAcrescimo": 0,
```

```
          "valorTotalDesconto": 0,
          "cancelado": true,
          "cancelado_por": "1275895",
          "preco_unitario": 50,
          "preco_total": 50,
          "data_hora": "27/01/2026, 11:07:01"
        }
      ]
    }
 ],
 "comandasBloqueadasParaFaturar": []
}
```


Response – Sucesso (200) contendo comandas aptas e não aptas a faturar.


```
{
 "comandasParaFaturar": [
   {
     "idMesa": 4069,
     "codigoComanda": "21",
     "statusAlteradoPara": "conferencia",
      "totalComanda": 31.50,
     "itens": [
       {
         "id": 34,
         "usuario_lancamento": "1275895",
         "produto_id": 9381649,
         "ean": "00000000100700",
         "descricao_resumida": "PROD TESTE 01",
         "quantidade": 1,
         "valorTotalAcrescimo": 10,
```

```
    "valorTotalDesconto": 0,
    "cancelado": false,
    "cancelado_por": null,
    "preco_unitario": 15,
    "preco_total": 25,
    "data_hora": "27/01/2026, 11:03:12"

  },
  {
    "id": 35,
    "usuario_lancamento": "1275895",
    "produto_id": 9381647,
    "ean": "00000000100670",
    "descricao_resumida": "PROD TESTE 40",
    "quantidade": 1,
    "valorTotalAcrescimo": 0,
    "valorTotalDesconto": 0,
    "cancelado": false,
    "cancelado_por": null,
    "preco_unitario": 6.5,
    "preco_total": 6.5,
    "data_hora": "27/01/2026, 11:03:12"
  },
  {
    "id": 36,
    "usuario_lancamento": "1275895",
    "produto_id": 8104916,
    "ean": "00000001267532",
    "descricao_resumida": "10 OVO",
    "quantidade": 1,
    "valorTotalAcrescimo": 0,
    "valorTotalDesconto": 0,
    "cancelado": true,
    "cancelado_por": "1275895",
    "preco_unitario": 50,
    "preco_total": 50,
    "data_hora": "27/01/2026, 11:07:01"
  }
]
```

```
    }
  ],
  "comandasBloqueadasParaFaturar": [
    {
      "idMesa": 4070,
      "codigoComanda": "22",
      "statusAtual": "conferencia",
      "motivo": "EM_CONFERENCIA",
       "numeroPdvEmConferencia": "12",
      "mensagem": "Comanda já está em conferência."
    },
    {
      "idMesa": 4071,
      "codigoComanda": "23",
      "statusAtual": "livre",
      "motivo": "STATUS_LIVRE",
      "mensagem": "Comanda está livre e sem itens a faturar."
    },
    {
      "codigoComanda": "23",
      "mensagem": "Comanda não encontrada"
}
  ]
}
```


Response – 200 - Nenhuma comanda apta a faturar.

Quando acontece?

Quando **nenhuma comanda** enviada pode ser faturada.

Comportamento esperado do PDV

- **Não montar a venda**
- Exibir os motivos ao operador
- Encerrar o fluxo

Exemplo


```
{
"comandasParaFaturar": [],
"comandasBloqueadasParaFaturar": [
   {
     "idMesa": 4070,
     "codigoComanda": "22",
     "statusAtual": "conferencia",
     "motivo": "EM_CONFERENCIA",
      "numeroPdvEmConferencia": "12",
     "mensagem": "Comanda já está em conferência."
   },
   {
     "idMesa": 4072,
     "codigoComanda": "23",
     "statusAtual": "livre",
     "motivo": "STATUS_LIVRE",
     "mensagem": "Comanda está livre e sem itens a faturar."
   },
   {
     "idMesa": 4073,
     "codigoComanda": "24",
     "statusAtual": "consumo",
     "motivo": "EM_LANCAMENTO_ITENS",
     "pos_lancando": [1,2]
     "mensagem": "Comanda está em consumo tendo lançamento de itens."
   },
   {
```

```
      "idMesa": 4071,
      "codigoComanda": "25",
      "statusAtual": "transferencia",
      "motivo": "EM_TRANSFERENCIA",
      "mensagem": "Comanda está em transferência de itens."
    },

    {
      "codigoComanda": "1231",
      "motivo": "NAO_ENCONTRADA",
      "mensagem": "Comanda não encontrada."
    }
  ]
}
```

Boas práticas para quem integra (PDV)

- Sempre trate **sucesso parcial**
- Nunca recalcular valores
- Não tente faturar comandas bloqueadas
- Guarde localmente as comandas que entraram em `conferencia`
- Use essa lista depois para:
  - `confirmar-faturamento`
  - `cancelar-faturamento`

API para liberar as comandas após finalizar a venda.

□□ API – Confirmar Faturamento (Liberar/Resetar Comandas)

Endpoint:
`POST /api/v1/mesas-comandas/confirmar-faturamento`

□□ Objetivo da API

Esta API é utilizada pelo **PDV** ao final do atendimento para **liberar (resetar)** uma ou mais comandas que estavam em `conferencia`, tornando-as `livre` novamente para um novo cliente.

□ O que essa API faz no Servidor de Comandas (para cada comanda liberada):

- apaga **itens** associados à comanda;
- apaga **pagamentos** associados (se existirem);
- limpa/zera informações associadas à mesa/comanda (mesma lógica do `liberar-mesa`);
- altera o status para `livre`.

□□ O que essa API **NÃO faz**:

- não recebe dados fiscais;
- não recebe pagamentos;
- não valida NFC-e/NF-e;
- não emite documento fiscal.

> Pagamento e emissão fiscal são responsabilidades do PDV e não são enviados ao Servidor de Comandas.

□□ Quando usar (fluxo do integrador)

Use esta API **apenas após** o PDV ter:

1. chamado `POST /api/v1/mesas-comandas/iniciar-faturamento` (e recebido comandas em `conferencia`), e
2. finalizado o atendimento no PDV.

□□ **Recomendação prática**:
Envie em `confirmar-faturamento` **somente as comandas que o servidor retornou em** `comandasParaFaturar` (ou seja, as que realmente entraram em `conferencia`).

□□ Autenticação

Segue o **padrão de autenticação do Servidor de Comandas** (não detalhado no cartão).

> ❝ Quando você definir/passar o padrão, incluo aqui (JWT, x-api-key, etc.).

□□ Request

Body (JSON)

Campo	Tipo	Obrigatório	Descrição

`numeroMesas`	`number[]`	☐	Lista de mesas/comandas a serem liberadas/resetadas
`numeroPdv`	`string`	☐	Identificação do PDV solicitante
`operador_id`	`string`	☐	Identificação do operador solicitante

Exemplo

```
{
  "numeroMesas": [4069, 4070, 4075],
  "numeroPdv": "12",
  "operador_id": "1275"
}
```

⚙ Regras de validação (por comanda)

Para cada `idMesa` em `numeroMesas`:

1) Comanda não existe

- Vai para `comandasBloqueadasParaConfirmar`
- `motivo = NAO_ENCONTRADA`

2) Comanda está em `conferencia`

- Pode liberar, **desde que**:
  - se existir `numeroPdvEmConferencia` e ele for **diferente** do `numeroPdv` do request → bloquear:
    - `motivo = EM_CONFERENCIA_OUTRO_PDV`
    - retornar `numeroPdvEmConferencia`

- caso contrário → liberar com sucesso

3) Comanda está `livre`

- Bloquear:
  - `motivo = STATUS_LIVRE` (já está disponível)

4) Comanda está `consumo`

- Se tiver `pos_lancando` → bloquear:
  - `motivo = EM_LANCAMENTO_ITENS`
  - retornar `pos_lancando`
- Se **não** tiver `pos_lancando` → bloquear:
  - `motivo = NAO_ESTA_EM_CONFERENCIA`
  - (porque não é uma comanda travada para faturamento do PDV Linux)

5) Comanda está `transferencia`

- Bloquear:
  - `motivo = EM_TRANSFERENCIA`

6) Outros status

- Bloquear:
  - `motivo = STATUS_INVALIDO`

> Observação importante: mesmo parecendo “reset”, a regra do fluxo do PDV Linux é **liberar somente o que estava em conferência**, para reduzir risco de liberar mesa errada.

□□ Response – Sucesso (200)

Quando acontece?

Quando **pelo menos 1 comanda** foi liberada/resetada com sucesso.

Estrutura do retorno

A API sempre retorna duas listas:

```
{
  "comandasConfirmadasParaFaturar": [],
  "comandasBloqueadasParaConfirmar": []
}
```

`comandasConfirmadasParaFaturar`

Lista das comandas que foram **liberadas/resetadas** com sucesso.

Campos

Campo	Descrição
`idMesa`	ID da comanda
`codigoComanda`	Código exibido ao operador
`statusAlteradoPara`	Sempre `livre`
`mensagem`	Texto explicativo

Exemplo — 200 (todas liberadas)

```
{
  "comandasConfirmadasParaFaturar": [
    {
      "idMesa": 4069,
      "codigoComanda": "21",
```

```
      "statusAlteradoPara": "livre",
      "mensagem": "Comanda liberada com sucesso."
    },
    {
      "idMesa": 4070,
      "codigoComanda": "22",
      "statusAlteradoPara": "livre",
      "mensagem": "Comanda liberada com sucesso."
    }
  ],
  "comandasBloqueadasParaConfirmar": []
}
```

□□ `comandasBloqueadasParaConfirmar`

Lista das comandas que **não puderam** ser liberadas.

Campos

Campo	Descrição
`idMesa`	ID da comanda (quando existir)
`codigoComanda`	Código da comanda (quando existir)
`statusAtual`	Status atual da comanda (quando existir)
`motivo`	Código padronizado do bloqueio
`mensagem`	Mensagem explicativa
`numeroPdvEmConferencia`	Somente quando `motivo = EM_CONFERENCIA_OUTRO_PDV`
`pos_lancando`	Somente quando `motivo = EM_LANCAMENTO_ITENS`

Exemplo — 200 (parcial)

```
{
  "comandasConfirmadasParaFaturar": [
    {
      "idMesa": 4069,
      "codigoComanda": "21",
      "statusAlteradoPara": "livre",
      "mensagem": "Comanda liberada com sucesso."
    }
  ],
  "comandasBloqueadasParaConfirmar": [
    {
      "idMesa": 4070,
      "codigoComanda": "22",
      "statusAtual": "conferencia",
      "motivo": "EM_CONFERENCIA_OUTRO_PDV",
      "numeroPdvEmConferencia": "15",
      "mensagem": "Comanda está em conferência por outro PDV."
    },
    {
      "idMesa": 4073,
      "codigoComanda": "24",
      "statusAtual": "consumo",
      "motivo": "EM_LANCAMENTO_ITENS",
      "pos_lancando": [1, 2],
      "mensagem": "Comanda está em consumo tendo lançamento de itens."
    },
    {
      "codigoComanda": "1231",
      "motivo": "NAO_ENCONTRADA",
      "mensagem": "Comanda não encontrada."
    }
  ]
}
```

Response – (200 Quando **nenhuma comanda** enviada pôde ser liberada/resetada.)

Quando acontece?

Quando **nenhuma comanda** enviada pôde ser liberada/resetada.

Comportamento esperado do PDV

- considerar que **nenhuma mesa foi liberada**;
- exibir os motivos ao operador;
- seguir procedimento operacional (ex.: tratativa gerencial/manual quando necessário).

Exemplo — 200 (nenhuma liberada)


```
{
  "comandasConfirmadasParaFaturar": [],
  "comandasBloqueadasParaConfirmar": [
    {
      "idMesa": 4070,
      "codigoComanda": "22",
      "statusAtual": "conferencia",
      "motivo": "EM_CONFERENCIA_OUTRO_PDV",
      "numeroPdvEmConferencia": "15",
      "mensagem": "Comanda está em conferência por outro PDV."
    },
    {
      "idMesa": 4073,
      "codigoComanda": "24",
      "statusAtual": "consumo",
      "motivo": "EM_LANCAMENTO_ITENS",
```

```
      "pos_lancando": [1, 2],
      "mensagem": "Comanda está em consumo tendo lançamento de itens."
    },
    {
      "idMesa": 4071,
      "codigoComanda": "23",
      "statusAtual": "livre",
      "motivo": "STATUS_LIVRE",
      "mensagem": "Comanda já está livre."
    },
    {
      "codigoComanda": "1231",
      "motivo": "NAO_ENCONTRADA",
      "mensagem": "Comanda não encontrada."
    }
  ]
}
```

□□ Enum `motivo` (mínimo)

Motivo	Significado
`NAO_ENCONTRADA`	Comanda não existe
`STATUS_LIVRE`	Comanda já está livre
`NAO_ESTA_EM_CONFERENCIA`	Comanda não está em conferência (não é do fluxo de faturamento)
`EM_CONFERENCIA_OUTRO_PDV`	Comanda está em conferência por outro PDV
`EM_TRANSFERENCIA`	Comanda em transferência
`EM_LANCAMENTO_ITENS`	Comanda em consumo com lançamento de itens em andamento
`STATUS_INVALIDO`	Status não permitido

Boas práticas para quem integra (PDV)

- Chame `confirmar-faturamento` **somente** para comandas que você recebeu em `comandasParaFaturar` no `iniciar-faturamento`.
- Sempre trate **retorno parcial**:
  - confirme/libere as que vierem em `comandasConfirmadasParaFaturar`
  - exiba e trate operacionalmente as bloqueadas
- Não tente “forçar” liberação de comanda em conferência de outro PDV via esta API (isso deve ser endpoint administrativo/gerencial).

API para voltar o status da comanda para consumo após abortar operação.

□□ API – Cancelar Faturamento de Comandas

Endpoint:
`POST /api/v1/mesas-comandas/cancelar-faturamento`

□□ Para que serve

Esta API é usada quando o PDV **iniciou o faturamento** de uma ou mais comandas (elas ficaram em `conferencia`) mas **o atendimento NÃO foi concluído**.

Ao chamar esta API, o Servidor de Comandas:

- remove a trava de faturamento;
- altera o status da comanda de `conferencia` → `consumo`;
- registra auditoria do cancelamento.

□ Use quando:

- o operador cancelou a venda no PDV;
- houve erro antes de concluir o atendimento (ex.: falha de cadastro de item);
- o cliente desistiu antes de finalizar.

□□ Não use quando:

- a venda foi concluída e você quer liberar a comanda para novo cliente → use `confirmar-faturamento`.

□□ Onde entra no fluxo do PDV

Fluxo típico:

1. PDV chama `iniciar-faturamento` → recebe `comandasParaFaturar` (essas entram em `conferencia`)
2. PDV tenta concluir a venda (pagamento/emissão fiscal são do PDV)
3. Se desistiu/cancelou → chama `cancelar-faturamento` enviando **somente as mesas que entraram em conferência**

> ⚠ Boa prática obrigatória:
> Envie para `cancelar-faturamento` **somente** as comandas que retornaram em `comandasParaFaturar` no `iniciar-faturamento`.

□□ Autenticação

Segue o padrão de autenticação do Servidor de Comandas (definido pelo projeto).

> Se houver falha, o servidor retornará HTTP 401/403.

□□ Request

Body (JSON)

Campo	Tipo	Obrigatório	Descrição

`numeroMesas`	`number[]`		IDs das comandas a cancelar (em lote)
`numeroPdv`	`string`		Número/identificação do PDV solicitante
`operador_id`	`string`		Identificação do operador solicitante

Exemplo

```
{
  "numeroMesas": [4069, 4070, 4075],
  "numeroPdv": "12",
  "operador_id": "1275"
}
```

O que o servidor faz (regras por comanda)

O servidor avalia **cada idMesa individualmente** e pode cancelar algumas e bloquear outras (retorno parcial).

Quando a comanda é cancelada com sucesso

- Se `status = conferencia` **e** (se existir) `numeroPdvEmConferencia == numeroPdv` do request → servidor altera status para `consumo` e retorna em `comandasCanceladasParaFaturar`.

Quando a comanda é bloqueada

A comanda entra em `comandasBloqueadasParaCancelar` nos casos:

Situação	Motivo	Campo extra
Comanda não existe	`NAO_ENCONTRADA`	—
`status = conferencia` por outro PDV	`EM_CONFERENCIA_OUTRO_PDV`	`numeroPdvEmConferencia`
`status = consumo`	`NAO_ESTA_EM_CONFERENCIA`	—
`status = livre`	`STATUS_LIVRE`	—
`status = transferencia`	`EM_TRANSFERENCIA`	—
`status = consumo` com lançamento em andamento	`EM_LANCAMENTO_ITENS`	`pos_lancando`
qualquer outro status	`STATUS_INVALIDO`	—

□□ Response

□ 200 – Cancelamento total ou parcial

Retorna **200** quando **pelo menos 1 comanda** foi cancelada com sucesso.

Estrutura

```
{
  "comandasCanceladasParaFaturar": [],
  "comandasBloqueadasParaCancelar": []
}
```

`comandasCanceladasParaFaturar[]` (sucesso)

Lista de comandas que foram efetivamente destravadas:

Campo	Descrição
`idMesa`	ID da comanda
`codigoComanda`	Código/identificador da comanda

Campo	Descrição
`statusAlteradoPara`	Sempre `"consumo"`
`mensagem`	Mensagem para exibição/log

`comandasBloqueadasParaCancelar[]` (bloqueios)

Lista de comandas que não puderam ser canceladas:

Campo	Descrição
`idMesa`	ID da comanda (quando existir)
`codigoComanda`	Código da comanda (quando existir)
`statusAtual`	Status atual (quando existir)
`motivo`	Código do bloqueio
`mensagem`	Mensagem explicativa
`numeroPdvEmConferencia`	Apenas quando `EM_CONFERENCIA_OUTRO_PDV`
`pos_lancando`	Apenas quando `EM_LANCAMENTO_ITENS`

□ Exemplo — 200 (cancelamento completo)


```
{
  "comandasCanceladasParaFaturar": [
    {
      "idMesa": 4069,
      "codigoComanda": "21",
      "statusAlteradoPara": "consumo",
      "mensagem": "Faturamento cancelado com sucesso. Comanda retornou para consumo."
    },
    {
```

```
      "idMesa": 4070,
      "codigoComanda": "22",
      "statusAlteradoPara": "consumo",
      "mensagem": "Faturamento cancelado com sucesso. Comanda retornou para consumo."
    }
  ],
  "comandasBloqueadasParaCancelar": []
}
```

Exemplo — 200 (parcial)


```
{
  "comandasCanceladasParaFaturar": [
    {
      "idMesa": 4069,
      "codigoComanda": "21",
      "statusAlteradoPara": "consumo",
      "mensagem": "Faturamento cancelado com sucesso. Comanda retornou para consumo."
    }
  ],
  "comandasBloqueadasParaCancelar": [
    {
      "idMesa": 4070,
      "codigoComanda": "22",
      "statusAtual": "conferencia",
      "motivo": "EM_CONFERENCIA_OUTRO_PDV",
      "numeroPdvEmConferencia": "15",
      "mensagem": "Comanda está em conferência por outro PDV."
    },
    {
      "idMesa": 4072,
      "codigoComanda": "24",
      "statusAtual": "transferencia",
      "motivo": "EM_TRANSFERENCIA",
```

```
      "mensagem": "Comanda está em transferência de itens e não pode ser cancelada."
    }
  ]
}
```


---

200 — nenhuma comanda cancelável

Retorna erro quando **0 comandas** puderam ser canceladas.

Exemplo — 200


```
{
  "comandasCanceladasParaFaturar": [],
  "comandasBloqueadasParaCancelar": [
    {
      "idMesa": 4070,
      "codigoComanda": "22",
      "statusAtual": "livre",
      "motivo": "STATUS_LIVRE",
      "mensagem": "Comanda está livre e não possui faturamento para cancelar."
    },
    {
      "idMesa": 4073,
      "codigoComanda": "26",
      "statusAtual": "consumo",
      "motivo": "NAO_ESTA_EM_CONFERENCIA",
      "mensagem": "Comanda não está em conferência."
    },
    {
      "codigoComanda": "9999",
      "motivo": "NAO_ENCONTRADA",
      "mensagem": "Comanda não encontrada."
```

```
    }
  ]
}
```

□□ Enum `motivo` (mínimo)

- `NAO_ENCONTRADA`
- `NAO_ESTA_EM_CONFERENCIA`
- `STATUS_LIVRE`
- `EM_TRANSFERENCIA`
- `EM_LANCAMENTO_ITENS`
- `EM_CONFERENCIA_OUTRO_PDV`
- `STATUS_INVALIDO`

□ Boas práticas para integradores (PDV)

- Sempre trate **retorno parcial**.
- Envie apenas as comandas que você recebeu em `comandasParaFaturar` no `iniciar-faturamento`.
- Se vier `EM_CONFERENCIA_OUTRO_PDV`, isso é **tratativa gerencial/manual** (não tente "resolver" pelo PDV).
- Após cancelar, o PDV deve:
  - limpar o contexto da venda local;
  - permitir que a comanda seja faturada novamente.

Requisitos Centralizador PDV's Fast

AVANÇO INFORMÁTICA

Centralizador PDV's Fast

Guia de Requisitos para Implantação

Este documento descreve os requisitos de hardware, sistema operacional e infraestrutura de rede necessários para a instalação e operação do Centralizador PDV's Fast em ambiente do cliente.

1. Visão Geral da Solução

O Centralizador PDV's Fast é o componente central responsável por intermediar a comunicação entre os dispositivos do ambiente do cliente. Ele recebe e processa requisições provenientes de:

- App de Comandas / PDV Mobile (dispositivos SmartPOS) — comunicação via API REST sobre Wi-Fi.
- PDV Legado Desktop (Linux) — comunicação via rede cabeada.

Além disso, o Servidor Comandas precisa ter acesso ao servidor do Sistema Integral (ERP Linux), que é a máquina responsável pela gestão central dos dados do cliente.

Para que toda a solução funcione corretamente, é fundamental que a infraestrutura de rede e a máquina do Servidor Comandas atendam integralmente aos requisitos descritos neste documento.

2. Topologia de Comunicação e Requisitos de Rede

A ilustração abaixo representa a topologia de comunicação entre os componentes da solução:

Componente	Tipo de Conexão	Requisito de Rede
SmartPOS (App Comandas / PDV Mobile)	Wi-Fi	Deve estar conectado à rede Wi-Fi que permita acesso ao Servidor Comandas e também a internet.
PDV Legado Desktop (Linux)	Cabo de rede (Ethernet)	Deve estar conectado via cabo na rede que permita acesso ao Servidor Comandas e ao servidor do Sistema Integral (ERP) e também a internet.
Centralizador PDV's Fast (Windows)	Cabo de rede (Ethernet)	Deve estar na rede que permita comunicação com: SmartPOS (Wi-Fi), PDV Linux (cabeado) e servidor do Sistema Integral (ERP Linux) como também a internet.
Servidor do Sistema Integral (ERP Linux)	Cabo de rede (Ethernet)	Deve estar acessível pelo Servidor Comandas e pelo PDV Legado Linux.

2.1 Requisitos Detalhados de Rede

⚠ **Atenção:** A comunicação entre todos os componentes ocorre pela rede local (LAN) do cliente. O Servidor Comandas deve conseguir se comunicar simultaneamente com os SmartPOS via Wi-Fi e com o PDV Linux e o servidor do ERP via cabo.

Requisito	Descrição
Padrão de rede	Ethernet 10/100/1000 Mbps (Gigabit Ethernet recomendado).
Topologia	Todos os dispositivos devem estar na mesma rede local (LAN) ou com roteamento adequado entre sub-redes.
IP estático — Servidor Comandas	É obrigatório configurar um endereço IP fixo (estático) na máquina do Servidor Comandas, para que os SmartPOS e o PDV Linux sempre consigam localizá-lo.
Firewall / Bloqueios	As portas utilizadas pela aplicação Servidor Comandas devem estar liberadas no firewall da rede e no firewall do Windows da máquina. As portas específicas serão informadas pela equipe de implantação.
Wi-Fi (SmartPOS)	O roteador ou access point Wi-Fi deve estar na mesma rede ou ter roteamento para o Servidor Comandas. Sinal estável e sem isolamento de cliente (AP Isolation desativado).
Cabo de rede (PDV Linux)	Conexão física obrigatória via cabo Ethernet entre o PDV Linux e o switch ou roteador da rede do cliente.
Velocidade mínima recomendada	Rede local com pelo menos 100 Mbps entre os dispositivos.

3. Requisitos de Hardware — Máquina do Servidor Comandas

A máquina destinada a operar o Servidor Comandas deve atender às especificações mínimas a seguir. Recomendamos fortemente o uso das configurações recomendadas para garantir estabilidade e desempenho adequados.

Componente	Especificação Mínima	Especificação Recomendada
Sistema Operacional	Windows 11 — 64 bits	Windows 11 — 64 bits
Processador	Intel Core i3 (ou equivalente), clock mínimo de 2,3 GHz, pelo menos 4 núcleos.	Intel Core i5 ou superior, 4+ núcleos, clock ≥ 2,5 GHz.
Memória RAM	8 GB DDR3	12 GB DDR3 ou DDR4
Placa de Rede	Padrão 10/100/1000 Mbps (Gigabit Ethernet)	Padrão 10/100/1000 Mbps (Gigabit Ethernet)
Armazenamento	SSD SATA ou NVMe de no mínimo 240 GB (Kingston ou similar)	SSD SATA ou NVMe de no mínimo 240 GB (Kingston ou similar)

4. Requisitos de Sistema Operacional e Configuração da Máquina

4.1 Sistema Operacional

- Windows 11 — 64 bits
- A conta de usuário do Windows utilizada pela aplicação deve ter permissões de Administrador Local.

4.2 Configurações Importantes

- IP estático configurado na placa de rede (obrigatório — veja seção 2.1).
- Atualizações automáticas do Windows: recomendamos que reinicializações automáticas fora do horário de funcionamento sejam configuradas para evitar interrupções no serviço.
- Antivírus: caso haja antivírus instalado, os diretórios e processos do Servidor Comandas devem ser adicionados como exceção para evitar bloqueios indesejados.
- Firewall do Windows: as portas da aplicação devem ser liberadas (a equipe de implantação orientará sobre quais portas configurar).
- Ideal que essa máquina nunca seja desligada, já que ela realiza o envio das vendas ao servidor integral. Ou seja, ela precisa estar sempre em execução contínua.

5. Checklist de Pré-Implantação

Antes da visita de implantação, o cliente deve confirmar que todos os itens abaixo estão atendidos:

#	Item	Status
1	Máquina com Windows 11 — 64 bits disponível e ligada.	☐ Confirmado
2	Processador com pelo menos 4 núcleos e 2,3 GHz.	☐ Confirmado
3	Mínimo de 8 GB de RAM instalados.	☐ Confirmado

#	Item	Status
4	SSD de no mínimo 240 GB disponível.	☐ Confirmado
5	Placa de rede Gigabit (10/100/1000 Mbps).	☐ Confirmado
6	IP estático configurado (ou a ser configurado pela equipe Avanço).	☐ Confirmado
7	Cabo de rede conectando a máquina ao switch/roteador.	☐ Confirmado
8	SmartPOS com acesso à rede Wi-Fi que alcança o Servidor Comandas.	☐ Confirmado
9	PDV Linux (legado) com acesso via cabo à mesma rede do Servidor Comandas.	☐ Confirmado
10	Servidor do Sistema Integral (ERP Linux) acessível na rede.	☐ Confirmado
11	Firewall e antivírus configurados para permitir a aplicação (ou a configurar com apoio Avanço).	☐ Confirmado

6. Suporte e Contato

Em caso de dúvidas sobre os requisitos ou durante o processo de implantação, entre em contato com a equipe da Avanço Informática:

Avanço Informática — Suporte Técnico
Site: www.avancoinfo.com.br
Telefone: (31) 3025-1188