**ENDPOINTS GUIDE**

**Summary**

1. Orientations
2. Controllers
3. **Orientations**

In this documentation, we specified all controllers and their respective endpoints. Each endpoint has its SQL Query that will have to become a LINQ Query on the api application.

*SQL Filters*

The payload we’re sending are specified between “{}” with the green color. If the filter has a “?” included, it means that it is optional. See the example below:

**{userName: string}** – Required parameter on payload to be used on the query.

**{phoneNumber?: string}** – Optional parameter on payload to be used on the query.

Filters that comes from JWT authenticated user are specified with “{}” with the orange color.

They follow the same rule of payload for optional filters. See the example below:

**{userId: int}** – Required parameter on JWT authentication to be used on the query.

**{userEmail?: string}** – Optional parameter on JWT authentication to be used on the query.

1. **Controllers**

**Home Controller**

**Controller Route:** /v1/home/

**Authentication:** Authorize

**Endpoint:**

* /get-totals – GET

Select Query:

-- Red Channel

select Count(\*)

from protocolo p

inner join unidade\_empresa ue on ue.cod\_unidade\_empresa = p.cod\_unidade\_empresa

inner join nivel\_Servico ns on ns.cod\_nivel\_servico = p.cod\_nivel\_servico

inner join transportadora t on t.cod\_transportadora = p.cod\_transportadora

left join prazo\_entrega pe on pe.cod\_prazo\_entrega = p.cod\_prazo\_entrega

left join protocolo\_cotacao pc on pc.cod\_protocolo = p.cod\_protocolo

where exists (select 0 from aprovacao\_alcada ap

inner join alcada\_aprovacao al on al.cod\_alcada\_aprovacao = ap.cod\_alcada\_aprovacao

where ap.cod\_protocolo = p.cod\_protocolo

and al.COD\_TIPO\_ALCADA\_APROVACAO = 2 and ap.dat\_aprovacao is null)

-- Ocurrences

select distinct p.COD\_PROTOCOLO,(select top 1 nom\_cliente from evento\_erp erp where erp.cod\_protocolo = p.cod\_protocolo) as [cliente],

mop.DSC\_MOTIVO\_OCORRENCIA\_PADRAO,t.nom\_fantasia, convert(nvarchar, r.dat\_saida, 103) as [dat\_saida], convert(nvarchar, p.dat\_limite\_entrega, 103) as [dat\_limite\_entrega],

(p.VAL\_FRETE\_PESO + p.VAL\_TAXA\_TOTAL) as [VALOR],

p.VALIDACAO\_ACAO\_TRANSPORTADORA,

u.nom\_usuario,

l.sgl\_unidade\_federacao

from protocolo p

inner join unidade\_empresa ue on ue.cod\_unidade\_empresa = p.cod\_unidade\_empresa

inner join ocorrencia\_transporte ot on ot.cod\_ocorrencia\_transporte = p.cod\_ocorrencia\_transporte\_acao

inner join aprovacao\_alcada aa on aa.cod\_protocolo = p.cod\_protocolo and aa.cod\_ocorrencia\_transporte = p.COD\_OCORRENCIA\_TRANSPORTE\_ACAO

inner join romaneio r on r.cod\_romaneio = p.cod\_romaneio

inner join localidade l on l.cod\_localidade = p.cod\_localidade

inner join motivo\_ocorrencia\_padrao mop on mop.cod\_motivo\_ocorrencia\_padrao = ot.cod\_motivo\_ocorrencia\_padrao

inner join transportadora t on t.cod\_transportadora = r.cod\_transportadora

inner join usuario u on u.cod\_usuario = aa.cod\_usuario\_validador

where aa.DAT\_APROVACAO is null

and aa.cod\_alcada\_aprovacao IN ( select ao.cod\_alcada\_aprovacao from aprovador\_ocorrencia ao where ao.cod\_usuario = 855 and ao.cod\_unidade\_empresa = ue.cod\_unidade\_empresa)

-- Como esta query tem distinct, o count é a quaintidade de registros retornado na consulta à base

-- 855 em cod\_usuario é o ID do usuário que está navegando

-- COLETA INFRUTIFERA

select count(\*)

from protocolo p

inner join romaneio r on r.cod\_romaneio = p.cod\_romaneio

where exists (select 0 from aprovacao\_alcada ap

inner join alcada\_aprovacao al on al.cod\_alcada\_aprovacao = ap.cod\_alcada\_aprovacao

where ap.cod\_protocolo = p.cod\_protocolo

and al.COD\_TIPO\_ALCADA\_APROVACAO = 3 and ap.dat\_aprovacao is null)

-- Protocol Weight

select Count(\*)

from protocolo p

inner join unidade\_empresa ue on ue.cod\_unidade\_empresa = p.cod\_unidade\_empresa

where exists (select 0 from aprovacao\_alcada ap

inner join alcada\_aprovacao al on al.cod\_alcada\_aprovacao = ap.cod\_alcada\_aprovacao

where ap.cod\_protocolo = p.cod\_protocolo

and al.COD\_TIPO\_ALCADA\_APROVACAO = 4 and ap.dat\_aprovacao is null)

-- Expensive Shipping

select Count(\*)

from protocolo p

inner join unidade\_empresa ue on ue.cod\_unidade\_empresa = p.cod\_unidade\_empresa

inner join nivel\_Servico ns on ns.cod\_nivel\_servico = p.cod\_nivel\_servico

inner join transportadora t on t.cod\_transportadora = p.cod\_transportadora

left join prazo\_entrega pe on pe.cod\_prazo\_entrega = p.cod\_prazo\_entrega

left join protocolo\_cotacao pc on pc.cod\_protocolo = p.cod\_protocolo

where exists (select 0 from aprovacao\_alcada ap

inner join alcada\_aprovacao al on al.cod\_alcada\_aprovacao = ap.cod\_alcada\_aprovacao

where ap.cod\_protocolo = p.cod\_protocolo

and al.COD\_TIPO\_ALCADA\_APROVACAO = 5 and ap.dat\_aprovacao is null)

-- Revalidation

select count(\*)

from protocolo p

inner join localidade l on l.cod\_localidade = p.cod\_localidade

inner join unidade\_empresa ue on ue.cod\_unidade\_empresa = p.cod\_unidade\_empresa

inner join ocorrencia\_transporte ot on ot.cod\_ocorrencia\_transporte = p.cod\_ocorrencia\_transporte\_acao

inner join motivo\_ocorrencia\_padrao mop on mop.cod\_motivo\_ocorrencia\_padrao = ot.cod\_motivo\_ocorrencia\_padrao

inner join transportadora t on t.cod\_transportadora = p.cod\_transportadora

inner join romaneio r on r.cod\_romaneio = p.cod\_romaneio

where isNull(p.ind\_acao\_transportadora,0) = 1 and p.dat\_acao\_transportadora is null

AND isNull(p.VALIDACAO\_ACAO\_TRANSPORTADORA,'') = ''

**Company Controller**

**Controller Route:** /v1/company/

**Authentication:** Authorize

**Endpoint:**

* GET - /

Select Query:

SELECT nom\_fantasia, cod\_empresa

FROM empresa

ORDER BY nom\_fantasia

**Company Unit Controller**

**Controller Route:** /v1/company-unit/

**Authentication:** Authorize

**Endpoint:**

* GET - /

Select Query:

SELECT CONCAT (

NOM\_UNIDADE\_EMPRESA

,' - '

,e.nom\_fantasia

) AS NOM\_UNIDADE\_EMPRESA

,COD\_UNIDADE\_EMPRESA

FROM UNIDADE\_EMPRESA ue

INNER JOIN empresa e ON e.cod\_empresa = ue.cod\_empresa

WHERE cod\_unidade\_empresa\_expedidora IS NULL

ORDER BY ue.NOM\_UNIDADE\_EMPRESA

**Channel Controller**

**Controller Route:** /v1/channel/

**Authentication:** Authorize

**Endpoint:**

* GET - /

Select Query:

SELECT DSC\_CANAL\_VENDA, COD\_CANAL\_VENDA

FROM canal\_venda c

ORDER BY DSC\_CANAL\_VENDA

**Carrier Controller**

**Controller Route:** /v1/carrier/

**Authentication:** Authorize

**Endpoint:**

* GET - /

Select Query:

SELECT cod\_transportadora, nom\_fantasia

FROM transportadora t

ORDER BY nom\_fantasia

**Protocol Controller**

**Controller Route:** /v1/protocol/

**Authentication:** Authorize

**Endpoints:**

* GET - /get-red-channels

Select Query:

SELECT ns.DSC\_NIVEL\_SERVICO

,p.cod\_protocolo

,t.nom\_fantasia

,isNull(VAL\_FRETE\_PESO, 0) + isNull(VAL\_TAXA\_TOTAL, 0) AS val\_frete

,p.DSC\_JUSTIFICATIVA\_CANAL\_VERMELHO

,(

SELECT TOP 1 DSC\_ALCADA\_APROVACAO

FROM alcada\_aprovacao aa

INNER JOIN aprovacao\_alcada aal ON aal.cod\_alcada\_aprovacao = aa.cod\_alcada\_aprovacao

WHERE aal.cod\_protocolo = p.cod\_protocolo

AND aa.cod\_tipo\_alcada\_aprovacao = 2

AND aal.dat\_aprovacao IS NULL

ORDER BY cod\_aprovacao\_alcada

) AS [ALCADA]

,isNull(pe.qtd\_dia\_prazo, pc.qtd\_dia\_prazo) AS qtd\_dia\_prazo

FROM protocolo p

INNER JOIN unidade\_empresa ue ON ue.cod\_unidade\_empresa = p.cod\_unidade\_empresa

INNER JOIN nivel\_Servico ns ON ns.cod\_nivel\_servico = p.cod\_nivel\_servico

INNER JOIN transportadora t ON t.cod\_transportadora = p.cod\_transportadora

LEFT JOIN prazo\_entrega pe ON pe.cod\_prazo\_entrega = p.cod\_prazo\_entrega

LEFT JOIN protocolo\_cotacao pc ON pc.cod\_protocolo = p.cod\_protocolo

WHERE EXISTS (

SELECT 0

FROM aprovacao\_alcada ap

INNER JOIN alcada\_aprovacao al ON al.cod\_alcada\_aprovacao = ap.cod\_alcada\_aprovacao

WHERE ap.cod\_protocolo = p.cod\_protocolo

AND al.COD\_TIPO\_ALCADA\_APROVACAO = 2

AND ap.dat\_aprovacao IS NULL

)

and p.cod\_unidade\_empresa = **{cod\_unidade\_empresa?: int}**

and ue.cod\_empresa = **{cod\_empresa?: int}**

and p.cod\_transportadora = **{cod\_transportadora?: int}**

and p.cod\_localidade in (

select cod\_localidade from localidade

where sgl\_unidade\_federacao in (**{sgl\_unidade\_federacao?: string[]}**)

)

and exists (

select 0

from evento\_erp erp

where erp.cod\_protocolo = p.cod\_protocolo

and erp.COD\_CANAL\_VENDA = **{cod\_canal\_venda?: int}**

)

* GET - /get-unsuccessful-collection

Select Query:

SELECT p.COD\_PROTOCOLO

,(

SELECT TOP 1 nom\_cliente

FROM evento\_erp erp

WHERE erp.cod\_protocolo = p.cod\_protocolo

) AS [cliente]

,t.nom\_fantasia

,convert(NVARCHAR, r.dat\_saida, 103) AS [dat\_saida]

,convert(NVARCHAR, p.dat\_limite\_entrega, 103) AS [dat\_limite\_entrega]

,CONCAT (

'R$ '

,replace(format((p.VAL\_FRETE\_PESO + p.VAL\_TAXA\_TOTAL), '#.00'), '.', ',')

) AS [VALOR]

,STUFF((

SELECT CONCAT (

','

,convert(INTEGER, replace(erp.num\_nota\_fiscal, 'A', ''))

)

FROM evento\_erp erp

WHERE erp.cod\_protocolo = p.cod\_protocolo

FOR XML PATH('')

), 1, 1, '') AS [notas]

,CONCAT (

'R$ '

,replace(format(((p.VAL\_FRETE\_PESO + p.VAL\_TAXA\_TOTAL) \* 0.2), '#.00'), '.', ',')

) AS [VALOR\_ACAO]

,(

SELECT TOP 1 DSC\_ALCADA\_APROVACAO

FROM alcada\_aprovacao aa

INNER JOIN aprovacao\_alcada aal ON aal.cod\_alcada\_aprovacao = aa.cod\_alcada\_aprovacao

WHERE aal.cod\_protocolo = p.cod\_protocolo

AND aal.cod\_ocorrencia\_transporte = p.COD\_OCORRENCIA\_TRANSPORTE\_ACAO

AND aal.dat\_aprovacao IS NULL

ORDER BY cod\_aprovacao\_alcada

) AS [ALCADA]

FROM protocolo p

INNER JOIN unidade\_empresa ue ON ue.cod\_unidade\_empresa = p.cod\_unidade\_empresa

INNER JOIN transportadora t ON t.cod\_transportadora = p.cod\_transportadora

INNER JOIN romaneio r ON r.cod\_romaneio = p.cod\_romaneio

WHERE EXISTS (

SELECT 0

FROM aprovacao\_alcada ap

INNER JOIN alcada\_aprovacao al ON al.cod\_alcada\_aprovacao = ap.cod\_alcada\_aprovacao

WHERE ap.cod\_protocolo = p.cod\_protocolo

AND al.COD\_TIPO\_ALCADA\_APROVACAO = 3

AND ap.dat\_aprovacao IS NULL

)

and p.cod\_unidade\_empresa = **{cod\_unidade\_empresa?: int}**

and ue.cod\_empresa = **{cod\_empresa?: int}**

and p.cod\_transportadora = **{cod\_transportadora?: int}**

and p.cod\_localidade in (

select cod\_localidade from localidade

where sgl\_unidade\_federacao in (**{sgl\_unidade\_federacao?: string[]}**)

)

and exists (

select 0

from evento\_erp erp

where erp.cod\_protocolo = p.cod\_protocolo

and erp.COD\_CANAL\_VENDA = **{cod\_canal\_venda?: int}**

)

* GET - /get-expensive-shipping

Select Query:

SELECT ns.DSC\_NIVEL\_SERVICO

,p.cod\_protocolo

,t.nom\_fantasia

,isNull(VAL\_FRETE\_PESO, 0) + isNull(VAL\_TAXA\_TOTAL, 0) AS val\_frete

,(

SELECT TOP 1 DSC\_ALCADA\_APROVACAO

FROM alcada\_aprovacao aa

INNER JOIN aprovacao\_alcada aal ON aal.cod\_alcada\_aprovacao = aa.cod\_alcada\_aprovacao

WHERE aal.cod\_protocolo = p.cod\_protocolo

AND aa.cod\_tipo\_alcada\_aprovacao = 2

AND aal.dat\_aprovacao IS NULL

ORDER BY cod\_aprovacao\_alcada

) AS [ALCADA]

,isNull(pe.qtd\_dia\_prazo, pc.qtd\_dia\_prazo) AS qtd\_dia\_prazo

,p.cod\_romaneio

FROM protocolo p

INNER JOIN unidade\_empresa ue ON ue.cod\_unidade\_empresa = p.cod\_unidade\_empresa

INNER JOIN nivel\_Servico ns ON ns.cod\_nivel\_servico = p.cod\_nivel\_servico

INNER JOIN transportadora t ON t.cod\_transportadora = p.cod\_transportadora

LEFT JOIN prazo\_entrega pe ON pe.cod\_prazo\_entrega = p.cod\_prazo\_entrega

LEFT JOIN protocolo\_cotacao pc ON pc.cod\_protocolo = p.cod\_protocolo

WHERE EXISTS (

SELECT 0

FROM aprovacao\_alcada ap

INNER JOIN alcada\_aprovacao al ON al.cod\_alcada\_aprovacao = ap.cod\_alcada\_aprovacao

WHERE ap.cod\_protocolo = p.cod\_protocolo

AND al.COD\_TIPO\_ALCADA\_APROVACAO = 5

AND ap.dat\_aprovacao IS NULL

)

and p.cod\_unidade\_empresa = **{cod\_unidade\_empresa?: int}**

and ue.cod\_empresa = **{cod\_empresa?: int}**

and p.cod\_transportadora = **{cod\_transportadora?: int}**

and p.cod\_localidade in (

select cod\_localidade from localidade

where sgl\_unidade\_federacao in (**{sgl\_unidade\_federacao?: string[]}**)

)

and exists (

select 0

from evento\_erp erp

where erp.cod\_protocolo = p.cod\_protocolo

and erp.COD\_CANAL\_VENDA = **{cod\_canal\_venda?: int}**

)

* GET - /get-ocurrences

Filter:

**{COD\_USUARIO: int}** must be a claim sended by JWT.

Select Query:

SELECT DISTINCT p.COD\_PROTOCOLO

,(

SELECT TOP 1 nom\_cliente

FROM evento\_erp erp

WHERE erp.cod\_protocolo = p.cod\_protocolo

) AS [cliente]

,mop.DSC\_MOTIVO\_OCORRENCIA\_PADRAO

,t.nom\_fantasia

,convert(NVARCHAR, r.dat\_saida, 103) AS [dat\_saida]

,convert(NVARCHAR, p.dat\_limite\_entrega, 103) AS [dat\_limite\_entrega]

,(p.VAL\_FRETE\_PESO + p.VAL\_TAXA\_TOTAL) AS [VALOR]

,STUFF((

SELECT CONCAT (

', '

,convert(INTEGER, erp.num\_nota\_fiscal)

)

FROM evento\_erp erp

WHERE erp.cod\_protocolo = p.cod\_protocolo

FOR XML PATH('')

), 1, 1, '') AS [notas]

,p.VALIDACAO\_ACAO\_TRANSPORTADORA

,u.nom\_usuario

,l.sgl\_unidade\_federacao

FROM protocolo p

INNER JOIN unidade\_empresa ue ON ue.cod\_unidade\_empresa = p.cod\_unidade\_empresa

INNER JOIN ocorrencia\_transporte ot ON ot.cod\_ocorrencia\_transporte = p.cod\_ocorrencia\_transporte\_acao

INNER JOIN aprovacao\_alcada aa ON aa.cod\_protocolo = p.cod\_protocolo

AND aa.cod\_ocorrencia\_transporte = p.COD\_OCORRENCIA\_TRANSPORTE\_ACAO

INNER JOIN romaneio r ON r.cod\_romaneio = p.cod\_romaneio

INNER JOIN localidade l ON l.cod\_localidade = p.cod\_localidade

INNER JOIN motivo\_ocorrencia\_padrao mop ON mop.cod\_motivo\_ocorrencia\_padrao = ot.cod\_motivo\_ocorrencia\_padrao

INNER JOIN transportadora t ON t.cod\_transportadora = r.cod\_transportadora

INNER JOIN usuario u ON u.cod\_usuario = aa.cod\_usuario\_validador

WHERE aa.DAT\_APROVACAO IS NULL

AND aa.cod\_alcada\_aprovacao IN (

SELECT ao.cod\_alcada\_aprovacao

FROM aprovador\_ocorrencia ao

WHERE ao.cod\_usuario = **{COD\_USUARIO: int}**

AND ao.cod\_unidade\_empresa = ue.cod\_unidade\_empresa

)

and p.cod\_unidade\_empresa = **{cod\_unidade\_empresa?: int}**

and ue.cod\_empresa = **{cod\_empresa?: int}**

and p.cod\_transportadora = **{cod\_transportadora?: int}**

and p.cod\_localidade in (

select cod\_localidade from localidade

where sgl\_unidade\_federacao in (**{sgl\_unidade\_federacao?: string[]}**)

)

and exists (

select 0

from evento\_erp erp

where erp.cod\_protocolo = p.cod\_protocolo

and erp.COD\_CANAL\_VENDA = **{cod\_canal\_venda?: int}**

)

* GET - /get-protocols-weight

Select Query:

SELECT p.COD\_PROTOCOLO

,STUFF((

SELECT CONCAT (

','

,convert(INTEGER, replace(erp.num\_nota\_fiscal, 'A', ''))

)

FROM evento\_erp erp

WHERE erp.cod\_protocolo = p.cod\_protocolo

FOR XML PATH('')

), 1, 1, '') AS [notas]

,STUFF((

SELECT CONCAT (

',('

,sum(qtd\_produto)

,') '

,ierp.dsc\_grupo\_produto

,' '

)

FROM evento\_erp erp

INNER JOIN item\_evento\_erp ierp ON ierp.cod\_evento\_erp = erp.cod\_evento\_erp

WHERE erp.cod\_protocolo = p.cod\_protocolo

GROUP BY ierp.dsc\_grupo\_produto

FOR XML PATH('')

), 1, 1, '') AS [produtos]

,p.qtd\_peso\_total

,CONCAT (

'R$ '

,replace(format((p.VAL\_FRETE\_PESO + p.VAL\_TAXA\_TOTAL), '#.00'), '.', ',')

) AS [valor]

FROM protocolo p

INNER JOIN unidade\_empresa ue ON ue.cod\_unidade\_empresa = p.cod\_unidade\_empresa

WHERE EXISTS (

SELECT 0

FROM aprovacao\_alcada ap

INNER JOIN alcada\_aprovacao al ON al.cod\_alcada\_aprovacao = ap.cod\_alcada\_aprovacao

WHERE ap.cod\_protocolo = p.cod\_protocolo

AND al.COD\_TIPO\_ALCADA\_APROVACAO = 4

AND ap.dat\_aprovacao IS NULL

)

and p.cod\_unidade\_empresa = **{cod\_unidade\_empresa?: int}**

and ue.cod\_empresa = **{cod\_empresa?: int}**

and p.cod\_transportadora = **{cod\_transportadora?: int}**

and p.cod\_localidade in (

select cod\_localidade from localidade

where sgl\_unidade\_federacao in (**{sgl\_unidade\_federacao?: string[]}**)

)

and exists (

select 0

from evento\_erp erp

where erp.cod\_protocolo = p.cod\_protocolo

and erp.COD\_CANAL\_VENDA = **{cod\_canal\_venda?: int}**

)

* GET - /get-ocurrences-revalidation

Select Query:

SELECT p.COD\_PROTOCOLO

,(

SELECT TOP 1 nom\_cliente

FROM evento\_erp erp

WHERE erp.cod\_protocolo = p.cod\_protocolo

) AS [cliente]

,mop.DSC\_MOTIVO\_OCORRENCIA\_PADRAO

,t.nom\_fantasia

,convert(NVARCHAR, r.dat\_saida, 103) AS [dat\_saida]

,convert(NVARCHAR, p.dat\_limite\_entrega, 103) AS [dat\_limite\_entrega]

,CONCAT (

'R$ '

,replace(format((p.VAL\_FRETE\_PESO + p.VAL\_TAXA\_TOTAL), '#.00'), '.', ',')

) AS [VALOR]

,STUFF((

SELECT CONCAT (

','

,convert(INTEGER, erp.num\_nota\_fiscal)

,'-'

,erp.num\_serie\_nota\_fiscal

)

FROM evento\_erp erp

WHERE erp.cod\_protocolo = p.cod\_protocolo

FOR XML PATH('')

), 1, 1, '') AS [notas]

,VALIDACAO\_ACAO\_TRANSPORTADORA

,CASE

WHEN VALIDACAO\_ACAO\_TRANSPORTADORA = 'Reentrega'

THEN CONCAT (

'R$ '

,replace(format(((p.VAL\_FRETE\_PESO + p.VAL\_TAXA\_TOTAL) \* 0.5), '#.00'), '.', ',')

)

ELSE CONCAT (

'R$ '

,replace(format((p.VAL\_FRETE\_PESO + p.VAL\_TAXA\_TOTAL), '#.00'), '.', ',')

)

END AS [VALOR\_ACAO]

FROM protocolo p

INNER JOIN localidade l ON l.cod\_localidade = p.cod\_localidade

INNER JOIN unidade\_empresa ue ON ue.cod\_unidade\_empresa = p.cod\_unidade\_empresa

INNER JOIN ocorrencia\_transporte ot ON ot.cod\_ocorrencia\_transporte = p.cod\_ocorrencia\_transporte\_acao

INNER JOIN motivo\_ocorrencia\_padrao mop ON mop.cod\_motivo\_ocorrencia\_padrao = ot.cod\_motivo\_ocorrencia\_padrao

INNER JOIN transportadora t ON t.cod\_transportadora = p.cod\_transportadora

INNER JOIN romaneio r ON r.cod\_romaneio = p.cod\_romaneio

WHERE isNull(p.ind\_acao\_transportadora, 0) = 1

AND p.dat\_acao\_transportadora IS NOT NULL

AND isNull(p.VALIDACAO\_ACAO\_TRANSPORTADORA, '') <> ''

and p.cod\_protocolo = **{cod\_protocolo?: int}**

and exists (

select 0 from evento\_erp erp

where erp.cod\_protocolo = p.cod\_protocolo and erp.num\_nota\_fiscal = **{num\_nota\_fiscal?: int}**

)