
GNOVIT DIALER

1. PROCESSO DE INTEGRAÇÃO ENTRE SISTEMAS

O Gnovit Dialer é uma API compacta que capacita outros sistemas a realizar chamadas VOIP.

O ciclo de vida de um pedido de chamadas ocorre em três etapas, descritas resumidamente a seguir:

1. O sistema cliente faz um request ao web service CallServiceWS do Gnovit Dialer, informando os dados da chamada telefônica. O serviço responde com um OK, inserindo o pedido de ligação na fila de processamento do servidor Asterisk.
2. O Asterisk recebe o pedido, concretizando a ligação telefônica. Se ela é atendida pelo número de destino, a ligação é redirecionada para o ramal do operador de telemarketing correspondente.
3. No momento em que o Asterisk sabe o status da ligação (Linha ocupada, Chamada atendida, Chamada não-atendida, etc.), esse status é informado para um segundo web service, CallStatusServiceWS, que é responsável por comunicar ao sistema cliente a situação final da chamada.

Esse processo é detalhado nas próximas seções deste documento.

2. CALL SERVICE WS

O web service CallServiceWS disponibiliza uma interface para realizar ligações telefônicas via Asterisk e encaminhá-las para os respectivos operadores de telemarketing.

Ele possui duas formas de realizar a chamada: DIRETA ou SPIRE.

2.1. Operação MakeDirectCall

A operação **makeDirectCall** faz chamadas DIRETAS, onde, resumidamente, são informados por parâmetro o número de origem (agentNumber) e o número de destino (toPhone). Quando a chamada é atendida, ela é redirecionada para o ramal do agente.

2.1.1. Request

O request deve ser feito no padrão SOAP/XML.

DEFINIÇÃO DOS PARÂMETROS

PARÂMETRO	TIPO	OBRIGATÓRIO	DESCRIÇÃO
toPhone	xs:string	SIM	Número a ser chamado. Exemplo: 5521999991234
agentNumber	xs:string	SIM	Número/Ramal do Agente que está realizando a chamada. Exemplo: 1000
agentName	xs:string	NÃO	Nome do Agente que está realizando a chamada. Exemplo: Joaquim da Silva DEFAULT VALUE= <i>null</i>
timeout	xs:long	NÃO	Tempo em milisegundos em que o asterisk deve aguardar a chamada ser atendida. MINVALUE=10000 (10 segundos). DEFAULT VALUE=30000 (30 segundos).

EXEMPLO DE REQUEST

```
<?xml version='1.0' encoding='UTF-8'?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:dial="http://dialer.gnovit.com.br">
  <soapenv:Body>
    <dial:makeDirectCall>
      <toPhone>5521999991234</toPhone>
      <agentNumber>1000</agentNumber>
      <!--Optional:-->
      <agentName>Joaquim da Silva</agentName>
      <!--Optional:-->
      <timeout>25000</timeout>
    </dial:makeDirectCall>
  </soapenv:Body>
</soapenv:Envelope>
```

2.1.2. Response

DEFINIÇÃO DOS PARÂMETROS

PARÂMETRO	TIPO	OBRG	DESCRIÇÃO
queuedCall	Objeto queuedCallResponseDTO	SIM	Objeto que encapsula todos os campos de retorno por chamada.
queuedCall. callRequestId	xs:long	SIM	Código único de pedido de chamada gerado pelo Gnovit Dialer, usado para identificar o pedido durante todo o processo.

			* Mais detalhes a seguir.
queuedCall. statusCode	xs:int	SIM	Código numérico de status. 0 = OK 1 = ERROR * Mais detalhes a seguir.
queuedCall. statusDesc	xs:string	SIM	Descrição textual do status.
queuedCall. toPhone	xs:string	SIM	Os dados do pedido são repetidos, para facilitar a associação do pedido original com o callRequestId.
queuedCall. agentNumber	xs:string	SIM	Os dados do pedido são repetidos, para facilitar a associação do pedido original com o callRequestId.
queuedCall. agentName	xs:string	NÃO	Os dados do pedido são repetidos, para facilitar a associação do pedido original com o callRequestId.
queuedCall. timeout	xs:long	NÃO	Os dados do pedido são repetidos, para facilitar a associação do pedido original com o callRequestId.

EXEMPLO DE RESPONSE

```
<?xml version='1.0' encoding='UTF-8'?>
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <ns2:makeDirectCallResponse xmlns:ns2="http://dialer.gnovit.com.br">
      <queuedCall>
        <callRequestId>153472</callRequestId>
        <statusCode>0</statusCode>
        <statusDesc>OK</statusDesc>
        <toPhone>5521999991234</toPhone>
        <agentNumber>1000</agentNumber>
        <agentName>Joaquim da Silva</agentName>
        <timeout>25000</timeout>
      </queuedCall>
    </ns2:makeDirectCallResponse>
  </S:Body>
</S:Envelope>
```

MAIS DETALHES

Cada pedido de chamada gera uma resposta **queuedCall**. Ela contém um identificador único (**callRequestId**), gerado pelo próprio Gnovit Dialer, e que permite que o pedido seja identificado nos diversos serviços oferecidos pelo sistema, dada a natureza assíncrona e paralela de execução de chamadas telefônicas definidas na arquitetura do Gnovit Dialer. É informado também o status da operação.

Caso a chamada seja inserida com sucesso na fila de processamento do Asterisk, será informado o statusCode **0** com statusDesc **OK**. Esse status significa que assim que possível o Asterisk realizará a chamada.

Caso a chamada por ventura não seja inserida na fila de processamento do Asterisk, poderá ser informado o erro genérico de statusCode 1, com statusDesc **ERROR** ou ainda

qualquer statusCode maior que um, indicando algum erro específico e com o motivo descrito na mensagem statusDesc.

É importante que o sistema que solicita as chamadas faça o vínculo do pedido com o **callRequestId**, pois o status da ligação será informado por um segundo web service utilizando este mesmo id. Todo pedido novo gera um novo id. Ele nunca se repete durante a vida do Gnovit Dialer, permitindo uma identificação segura do pedido durante o histórico de uso do sistema.

2.2. Operação MakeSpireCall

A operação **makeSpireCall** faz chamadas SPIRE, onde, resumidamente, são informados por parâmetro o número de origem (agentNumber) e uma lista de números de destino (toPhones). O sistema ligará simultaneamente para todos os números, e quando o primeiro número for atendido, a chamada é redirecionada para o ramal do agente. Na medida em que as ligações são terminadas, os status são encaminhados, permitindo o mapeamento de todos os números discados.

Para exemplificar, caso sejam informados três números de destino, A, B e C, no final do processo se terá os três status das chamadas (número A - OCUPADO, número B ATENDIDO, número C, NÃO ATENDIDO). Neste exemplo, assim que o número B atendeu a ligação, a chamada foi redirecionada para o ramal do agente que solicitou a chamada SPIRE.

2.2.1. Request

O request deve ser feito no padrão SOAP/XML.

DEFINIÇÃO DOS PARÂMETROS

PARÂMETRO	TIPO	OBRIGATÓRIO	DESCRIÇÃO
toPhones	xs:string	SIM	Números a serem chamados, separados por ponto-e-vírgula. Exemplo: 5511999992345;551133347890;5 511988015678
agentNumber	xs:string	SIM	Número/Ramal do Agente que está realizando a chamada. Exemplo: 1008
agentName	xs:string	NÃO	Nome do Agente que está realizando a chamada. Exemplo: Gilberto Pereira DEFAULT VALUE=null
timeout	xs:long	NÃO	Tempo em milissegundos em que o asterisk deve aguardar a chamada ser atendida. MINVALUE=10000 (10 segundos). DEFAULT VALUE=30000 (30 segundos).

EXEMPLO DE REQUEST

```
<?xml version='1.0' encoding='UTF-8'?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:dial="http://dialer.gnovit.com.br">
```

```

<soapenv:Body>
  <dial:makeSpireCall>
    <toPhones>5511999992345;551133347890;5511988015678</toPhones>
    <agentNumber>1008</agentNumber>
    <!--Optional:-->
    <agentName>Gilberto Pereira</agentName>
    <!--Optional:-->
    <timeout>30000</timeout>
  </dial:makeSpireCall>
</soapenv:Body>
</soapenv:Envelope>

```

2.2.2. Response

DEFINIÇÃO DOS PARÂMETROS

PARÂMETRO	TIPO	OBRG	DESCRIÇÃO
Lista de queuedCall	Objeto queuedCallResponseDTO	SIM	Para cada número fornecido em toPhones, é devolvido um objeto queuedCall.
queuedCall.callRequestId	xs:long	SIM	Código único de pedido de chamada gerado pelo Gnovit Dialer, usado para identificar o pedido durante todo o processo. Cada número de toPhones tem seu próprio callRequestId.
queuedCall.statusCode	xs:int	SIM	Código numérico de status. 0 = OK 1 = ERROR
queuedCall.statusDesc	xs:string	SIM	Descrição textual do status.
queuedCall.toPhone	xs:string	SIM	Indica qual dos números de toPhones está sendo informado.
queuedCall.agentNumber	xs:string	SIM	Os dados do pedido são repetidos, para facilitar a associação do pedido original com o callRequestId.
queuedCall.agentName	xs:string	NÃO	Os dados do pedido são repetidos, para facilitar a associação do pedido original com o callRequestId.
queuedCall.timeout	xs:long	NÃO	Os dados do pedido são repetidos, para facilitar a associação do pedido original com o callRequestId.

EXEMPLO DE RESPONSE

```

<?xml version='1.0' encoding='UTF-8'?>
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <ns2:makeSpireCallResponse xmlns:ns2="http://dialer.gnovit.com.br">

```

```

<!--1 or more repetitions:-->
<!--Primeiro Numero:-->
<queuedCall>
  <callRequestId>153473</callRequestId>
  <statusCode>0</statusCode>
  <statusDesc>OK</statusDesc>
  <toPhone>5511999992345</toPhone>
  <agentNumber>1008</agentNumber>
  <agentName>Gilberto Pereira</agentName>
  <timeout>30000</timeout>
</queuedCall>
<!--Segundo Numero:-->
<queuedCall>
  <callRequestId>153474</callRequestId>
  <statusCode>0</statusCode>
  <statusDesc>OK</statusDesc>
  <toPhone>551133347890</toPhone>
  <agentNumber>1008</agentNumber>
  <agentName>Gilberto Pereira</agentName>
  <timeout>30000</timeout>
</queuedCall>
<!--Terceiro Numero:-->
<queuedCall>
  <callRequestId>153475</callRequestId>
  <statusCode>0</statusCode>
  <statusDesc>OK</statusDesc>
  <toPhone>5511988015678</toPhone>
  <agentNumber>1008</agentNumber>
  <agentName>Gilberto Pereira</agentName>
  <timeout>30000</timeout>
</queuedCall>
</ns2:makeSpireCallResponse>
</S:Body>
</S:Envelope>

```

MAIS DETALHES

O pedido SPIRE é feito de uma só vez, unificando os diversos números por ponto-e-vírgula no parâmetro **toPhones**. Porém, a resposta é dada numa lista de **queuedCalls**, um para cada telefone informado.

O **queuedCall** de cada telefone contém um identificador único (**callRequestId**), gerado pelo próprio Gnovit Dialer e exclusivo para cada um dos telefones solicitados e também é informado o status da operação para o respectivo número.

Os demais campos tem o mesmo comportamento já explicado no response do método **makeDirectCall**.

2.3. WSDL e XSD

WSDL

```

<?xml version='1.0' encoding='UTF-8'?>
<definitions
xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.x
sd" xmlns:wsp="http://www.w3.org/ns/ws-policy"
xmlns:wsp1_2="http://schemas.xmlsoap.org/ws/2004/09/policy"
xmlns:wsam="http://www.w3.org/2007/05/addressing/metadata"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" xmlns:tns="http://dialer.gnovit.com.br"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns="http://schemas.xmlsoap.org/wsdl/"

```

```

targetNamespace="http://dialer.gnovit.com.br" name="GnovitDialer">
<types>
<xsd:schema>
<xsd:import namespace="http://dialer.gnovit.com.br"
schemaLocation="http://localhost:8080/GnovitDialer/CallServiceWS?xsd=1"/>
</xsd:schema>
</types>
<message name="makeDirectCall">
<part name="parameters" element="tns:makeDirectCall"/>
</message>
<message name="makeDirectCallResponse">
<part name="parameters" element="tns:makeDirectCallResponse"/>
</message>
<message name="makeSpireCall">
<part name="parameters" element="tns:makeSpireCall"/>
</message>
<message name="makeSpireCallResponse">
<part name="parameters" element="tns:makeSpireCallResponse"/>
</message>
<portType name="CallServiceWS">
<operation name="makeDirectCall">
<input wsam:Action="http://dialer.gnovit.com.br/CallServiceWS/makeDirectCallRequest"
message="tns:makeDirectCall"/>
<output wsam:Action="http://dialer.gnovit.com.br/CallServiceWS/makeDirectCallResponse"
message="tns:makeDirectCallResponse"/>
</operation>
<operation name="makeSpireCall">
<input wsam:Action="http://dialer.gnovit.com.br/CallServiceWS/makeSpireCallRequest"
message="tns:makeSpireCall"/>
<output wsam:Action="http://dialer.gnovit.com.br/CallServiceWS/makeSpireCallResponse"
message="tns:makeSpireCallResponse"/>
</operation>
</portType>
<binding name="CallServiceWSPortBinding" type="tns:CallServiceWS">
<soap:binding transport="http://schemas.xmlsoap.org/soap/http" style="document"/>
<operation name="makeDirectCall">
<soap:operation soapAction=""/>
<input>
<soap:body use="literal"/>
</input>
<output>
<soap:body use="literal"/>
</output>
</operation>
<operation name="makeSpireCall">
<soap:operation soapAction=""/>
<input>
<soap:body use="literal"/>
</input>
<output>
<soap:body use="literal"/>
</output>
</operation>
</binding>
<service name="GnovitDialer">
<port name="CallServiceWSPort" binding="tns:CallServiceWSPortBinding">
<soap:address location="http://localhost:8080/GnovitDialer/CallServiceWS"/>
</port>

```

```
</service>
</definitions>
```

XSD

```
<?xml version='1.0' encoding='UTF-8'?>
<xs:schema xmlns:tns="http://dialer.gnovit.com.br"
xmlns:xs="http://www.w3.org/2001/XMLSchema" version="1.0"
targetNamespace="http://dialer.gnovit.com.br">

<xs:element name="makeDirectCall" type="tns:makeDirectCall"/>

<xs:element name="makeDirectCallResponse" type="tns:makeDirectCallResponse"/>

<xs:element name="makeSpireCall" type="tns:makeSpireCall"/>

<xs:element name="makeSpireCallResponse" type="tns:makeSpireCallResponse"/>

<xs:element name="queuedCallResponse" type="tns:queuedCallResponseDTO"/>

<xs:complexType name="makeSpireCall">
<xs:sequence>
<xs:element name="toPhones" type="xs:string"/>
<xs:element name="agentNumber" type="xs:string"/>
<xs:element name="agentName" type="xs:string" minOccurs="0"/>
<xs:element name="timeout" type="xs:long" minOccurs="0"/>
</xs:sequence>
</xs:complexType>

<xs:complexType name="makeSpireCallResponse">
<xs:sequence>
<xs:element name="queuedCall" type="tns:queuedCallResponseDTO"
maxOccurs="unbounded"/>
</xs:sequence>
</xs:complexType>

<xs:complexType name="queuedCallResponseDTO">
<xs:sequence>
<xs:element name="callRequestId" type="xs:long"/>
<xs:element name="statusCode" type="xs:int"/>
<xs:element name="statusDesc" type="xs:string"/>
<xs:element name="toPhone" type="xs:string"/>
<xs:element name="agentNumber" type="xs:string"/>
<xs:element name="agentName" type="xs:string" minOccurs="0"/>
<xs:element name="timeout" type="xs:long" minOccurs="0"/>
</xs:sequence>
</xs:complexType>

<xs:complexType name="makeDirectCall">
<xs:sequence>
<xs:element name="toPhone" type="xs:string"/>
<xs:element name="agentNumber" type="xs:string"/>
<xs:element name="agentName" type="xs:string" minOccurs="0"/>
<xs:element name="timeout" type="xs:long" minOccurs="0"/>
</xs:sequence>
</xs:complexType>

<xs:complexType name="makeDirectCallResponse">
```



```

<xs:sequence>
<xs:element name="queuedCall" type="tns:queuedCallResponseDTO"/>
</xs:sequence>
</xs:complexType>
</xs:schema>

```

3. CALL STATUS SERVICE WS

Todo sistema que solicita chamadas através do CallServiceWS, deverá implementar um web service que permita que o Gnovit Dialer informe imediatamente o status de cada chamada.

A interface deste web service - CallStatusServiceWS -, está definida nos itens a seguir, enfatizando que o Gnovit Dialer atuará como *client* deste serviço.

3.1. Operação SendCallStatus

A operação **sendCallStatus** é invocada pelo Gnovit Dialer informando ao consumidor do CallServiceWS o status da ligação feita pelo Asterisk, bem como a localização do arquivo de áudio caso esteja disponível.

Note que é possível encaminhar, num mesmo envelope SOAP, diversos callStatus, na medida em que estes estão disponíveis para serem comunicados ao sistema consumidor.

O sistema consumidor, por sua vez, deve informar que o callStatus foi recebido, e assim sendo, o processo de realização de chamadas para o respectivo pedido é concluído.

3.1.1. Request

O request deve ser feito no padrão SOAP/XML.

DEFINIÇÃO DOS PARÂMETROS

PARÂMETRO	TIPO	OBRG	DESCRIÇÃO
Lista de callStatus	Objeto CallStatusDTO	SIM	Lista de status fornecidos por pedido. * Mais detalhes a seguir.
callStatus. callRequestId	xs:long	SIM	Código único de pedido de chamada.
callStatus. toPhone	xs:string	SIM	Os dados do pedido feito em CallServiceWS são repetidos.
callStatus. agentNumber	xs:string	SIM	Os dados do pedido feito em CallServiceWS são repetidos.
callStatus. agentName	xs:string	NÃO	Os dados do pedido feito em CallServiceWS são repetidos.
callStatus. callStatus	DialStatusType	SIM	Status informado pelo Asterisk ao realizar a chamada. * Mais detalhes a seguir.
callStatus. audioFile	xs:string	NÃO	O caminho do arquivo de áudio é fornecido, quando existe.

EXEMPLO DE REQUEST

```

<?xml version='1.0' encoding='UTF-8'?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:dial="http://dialer.gnovit.com.br">
<soapenv:Body>

```

```

<dial:sendCallStatus>
  <!--1 or more repetitions:-->
  <!--Numero DirectCall:-->
  <callStatus>
    <callRequestId>153472</callRequestId>
    <toPhone>5521999991234</toPhone>
    <agentNumber>1000</agentNumber>
    <agentName>Joaquim da Silva</agentName>
    <callStatus>BUSY</callStatus>
  </callStatus>
  <!--Numero SpireCall:-->
  <callStatus>
    <callRequestId>153473</callRequestId>
    <toPhone>5511999992345</toPhone>
    <agentNumber>1008</agentNumber>
    <agentName>Gilberto Pereira</agentName>
    <callStatus>ANSWER</callStatus>
    <!--Optional:-->
    <audioFile>/opt/dialer/audio-153473.mp3</audioFile>
  </callStatus>
</dial:sendCallStatus>
</soapenv:Body>
</soapenv:Envelope>

```

MAIS DETALHES

Uma ligação pode ser concluída pelo Asterisk a qualquer momento. Quando isto ocorre, é necessário que o status de encerramento seja comunicado ao sistema que solicitou a ligação o quanto antes. Deve-se levar em conta também que num volume muito grande de ligações, é possível que haja uma lista considerável de chamadas finalizadas, tornando-se interessante para fins de desempenho que num mesmo pedido, vários status possam ser transmitidos.

Considerando este cenário, o web service deve estar preparado para receber uma lista de callStatus, sendo que os callStatus informados não tem relação nenhuma entre si, ou seja, o primeiro callStatus informado pode ser de um agente fazendo uma ligação direta e o segundo callStatus de outro agente, sendo um número específico dentre os números SPIRE informados.

Os status foram mapeados no tipo DialStatusType, que reflete atualmente os status possíveis do Asterisk. São eles:

- ANSWER: Chamada atendida;
- BUSY: Linha ocupada;
- NOANSWER: Ninguém atendeu;
- CANCEL: Desistência por parte do Asterisk, geralmente é o TIMEOUT atingido;
- CONGESTION: Tráfego intenso na rede;
- CHANUNAVAIL: Sem canal para realizar a chamada;
- DONTCALL: Blacklist;
- TORTURE: Tortura;
- INVALIDARGS: Argumentos inválidos.

3.1.2. Response

DEFINIÇÃO DOS PARÂMETROS

PARÂMETRO	TIPO	OBRG	DESCRIÇÃO
Lista de	Objeto	SIM	Para cada callStatus enviado,

receiveStatus	CallStatusResponseDTO		uma confirmação de recebimento "receiveStatus" é respondida.
receiveStatus.callRequestId	xs:long	SIM	Código único de pedido de chamada.
receiveStatus.statusCode	xs:int	SIM	Código numérico de status. 0 = OK 1 = ERROR
receiveStatus.statusDesc	xs:string	SIM	Descrição textual do status.

EXEMPLO DE RESPONSE

```
<?xml version='1.0' encoding='UTF-8'?>
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <ns2:sendCallStatusResponse xmlns:ns2="http://dialer.gnovit.com.br">
      <!--1 or more repetitions:-->
      <!--Numero DirectCall:-->
      <receiveStatus>
        <callRequestId>153472</callRequestId>
        <statusCode>0</statusCode>
        <statusDesc>OK</statusDesc>
      </receiveStatus>
      <!--Numero SpireCall:-->
      <receiveStatus>
        <callRequestId>153473</callRequestId>
        <statusCode>0</statusCode>
        <statusDesc>OK</statusDesc>
      </receiveStatus>
    </ns2:sendCallStatusResponse>
  </S:Body>
</S:Envelope>
```

MAIS DETALHES

Enquanto o statusCode 0 não for recebido para determinado callRequestId, o Gnovit Dialer continuará enviando os callStatus correspondentes até que o OK seja recebido.

3.2. WSDL e XSD

WSDL

```
<?xml version='1.0' encoding='UTF-8'?>
<definitions
  xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"
  xmlns:wsp="http://www.w3.org/ns/ws-policy"
  xmlns:wsp1_2="http://schemas.xmlsoap.org/ws/2004/09/policy"
  xmlns:wsam="http://www.w3.org/2007/05/addressing/metadata"
  xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
  xmlns:tns="http://dialer.gnovit.com.br"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns="http://schemas.xmlsoap.org/wsdl/"
  targetNamespace="http://dialer.gnovit.com.br"
  name="GnovitDialer">
  <types>
    <xsd:schema>
      <xsd:import namespace="http://dialer.gnovit.com.br"
        schemaLocation="http://localhost:8080/GnovitDialer/CallStatusServiceWS?xsd=1"/>
```

```

</xsd:schema>
</types>
<message name="sendCallStatus">
<part name="parameters" element="tns:sendCallStatus"/>
</message>
<message name="sendCallStatusResponse">
<part name="parameters" element="tns:sendCallStatusResponse"/>
</message>
<portType name="CallStatusServiceWS">
<operation name="sendCallStatus">
<input
wsam:Action="http://dialer.gnovit.com.br/CallStatusServiceWS/sendCallStatusRequest"
message="tns:sendCallStatus"/>
<output
wsam:Action="http://dialer.gnovit.com.br/CallStatusServiceWS/sendCallStatusResponse"
message="tns:sendCallStatusResponse"/>
</operation>
</portType>
<binding name="CallStatusServiceWSPortBinding" type="tns:CallStatusServiceWS">
<soap:binding transport="http://schemas.xmlsoap.org/soap/http" style="document"/>
<operation name="sendCallStatus">
<soap:operation soapAction=""/>
<input>
<soap:body use="literal"/>
</input>
<output>
<soap:body use="literal"/>
</output>
</operation>
</binding>
<service name="GnovitDialer">
<port name="CallStatusServiceWSPort" binding="tns:CallStatusServiceWSPortBinding">
<soap:address location="http://localhost:8080/GnovitDialer/CallStatusServiceWS"/>
</port>
</service>
</definitions>

```

XSD

```

<?xml version='1.0' encoding='UTF-8'?>
<xs:schema xmlns:tns="http://dialer.gnovit.com.br"
xmlns:xs="http://www.w3.org/2001/XMLSchema" version="1.0"
targetNamespace="http://dialer.gnovit.com.br">

<xs:element name="callStatus" type="tns:callStatusDTO"/>

<xs:element name="callStatusResponse" type="tns:callStatusResponseDTO"/>

<xs:element name="sendCallStatus" type="tns:sendCallStatus"/>

<xs:element name="sendCallStatusResponse" type="tns:sendCallStatusResponse"/>

<xs:complexType name="sendCallStatus">
<xs:sequence>
<xs:element name="callStatus" type="tns:callStatusDTO" maxOccurs="unbounded"/>
</xs:sequence>
</xs:complexType>

```

```
<xs:complexType name="callStatusDTO">
  <xs:sequence>
    <xs:element name="callRequestId" type="xs:long"/>
    <xs:element name="toPhone" type="xs:string"/>
    <xs:element name="agentNumber" type="xs:string"/>
    <xs:element name="agentName" type="xs:string" minOccurs="0"/>
    <xs:element name="callStatus" type="tns:dialStatusType"/>
    <xs:element name="audioFile" type="xs:string" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="sendCallStatusResponse">
  <xs:sequence>
    <xs:element name="receiveStatus" type="tns:callStatusResponseDTO"
      maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="callStatusResponseDTO">
  <xs:sequence>
    <xs:element name="callRequestId" type="xs:long"/>
    <xs:element name="statusCode" type="xs:int"/>
    <xs:element name="statusDesc" type="xs:string"/>
  </xs:sequence>
</xs:complexType>

<xs:simpleType name="dialStatusType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="ANSWER"/>
    <xs:enumeration value="BUSY"/>
    <xs:enumeration value="NOANSWER"/>
    <xs:enumeration value="CANCEL"/>
    <xs:enumeration value="CONGESTION"/>
    <xs:enumeration value="CHANUNAVAIL"/>
    <xs:enumeration value="DONTCALL"/>
    <xs:enumeration value="TORTURE"/>
    <xs:enumeration value="INVALIDARGS"/>
  </xs:restriction>
</xs:simpleType>
</xs:schema>
```