



ConexX/ConexX_MiddleWare

Histórico das Revisões:

| Revisão | Modificação | Data | Responsável |
|---------|---|------------|--------------|
| 1.0 | Versão inicial | 12/05/2025 | Gabriel Hama |
| 1.1 | Tabela Firmware compatíveis | 23/05/2025 | Gabriel Hama |
| 1.2 | Adicionado Firmware rep 2k | 28/05/2025 | Gabriel Hama |
| 1.3 | Adicionado Histórico de Revisões | 29/05/2025 | Gabriel Hama |
| 1.4 | Adicionado Envio de Mensagem personalizada para acesso negado | 13/10/2025 | Gabriel Hama |
| 1.5 | Diagrama de Servidores | 16/10/2025 | Gabriel Hama |
| 1.6 | Atualização Configuração Middleware | 21/10/2025 | Gabriel Hama |
| 1.7 | Atualização Diagrama de Servidores, Tópicos "Compatibilidade e Teste de Ambiente" Adicionados | 23/10/2025 | Gabriel Hama |
| 1.8 | Atualizado Diagrama de Servidores e Configuração do ConexX_Driver, Adicionado Tópico "Validação da Licença" | 24/10/2025 | Gabriel Hama |
| 1.9 | Adicionado especificações das configuração do ConexX_Middleware | 27/10/2025 | Gabriel Hama |
| 1.10 | Configuração de Mensagem no Aplicativo | 17/11/2025 | Gabriel Hama |

© Copyright

Este documento é propriedade intelectual da Telemática Sistemas Inteligentes. Pode ser copiado parcial ou integralmente, desde que este Copyright esteja incluído em cada cópia.

® Marcas Registradas

SURICATO, CONEX e CODIN, são marcas registradas da Telemática sistemas Inteligentes. Especificações técnicas e suas disponibilidades estão sujeitas a alterações sem aviso prévio.

Tabela de Conteúdos

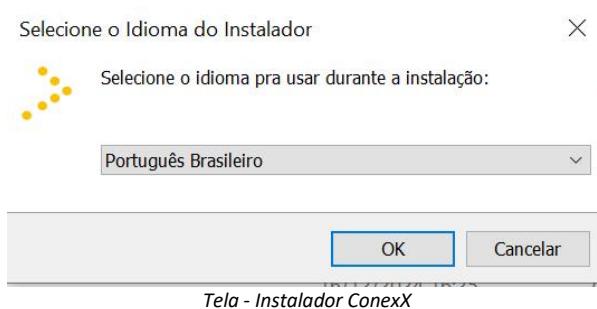
| | |
|--|----|
| 1. Instalações Serviços | 4 |
| 1.1. Instalação ConexX_Driver | 5 |
| 1.1.1. Configuração ConexX_Driver | 8 |
| 1.1.2. Desinstalar ConexX_Driver | 9 |
| 1.2. Instalação ConexX_Middleware | 10 |
| 1.2.1. Configuração ConexX_Middleware | 13 |
| 1.2.2. Desinstalar ConexX Middleware | 15 |
| 1.3. Validação da Licença | 16 |
| 2. Cadastro Equipamento (ConexX_Driver P5) | 17 |
| 2.1. Propriedades Extensíveis | 20 |
| 3. Cadastro Equipamento (ConexT_Driver P8) | 23 |
| 4. Teste de Ambiente | 26 |
| 5. Propriedades Extensíveis para Alarmes | 28 |
| 6. LOG | 30 |
| 7. Firmware | 31 |
| 8. Configuração Aplicativo | 32 |
| 8.1. Mensagem | 34 |
| 9. Diagrama dos Servidores | 35 |
| 10. Compatibilidade | 39 |

1. Instalações Serviços

Nesse Tópico iremos mostrar como instalar e Configurar os Serviços referentes ao ConexX

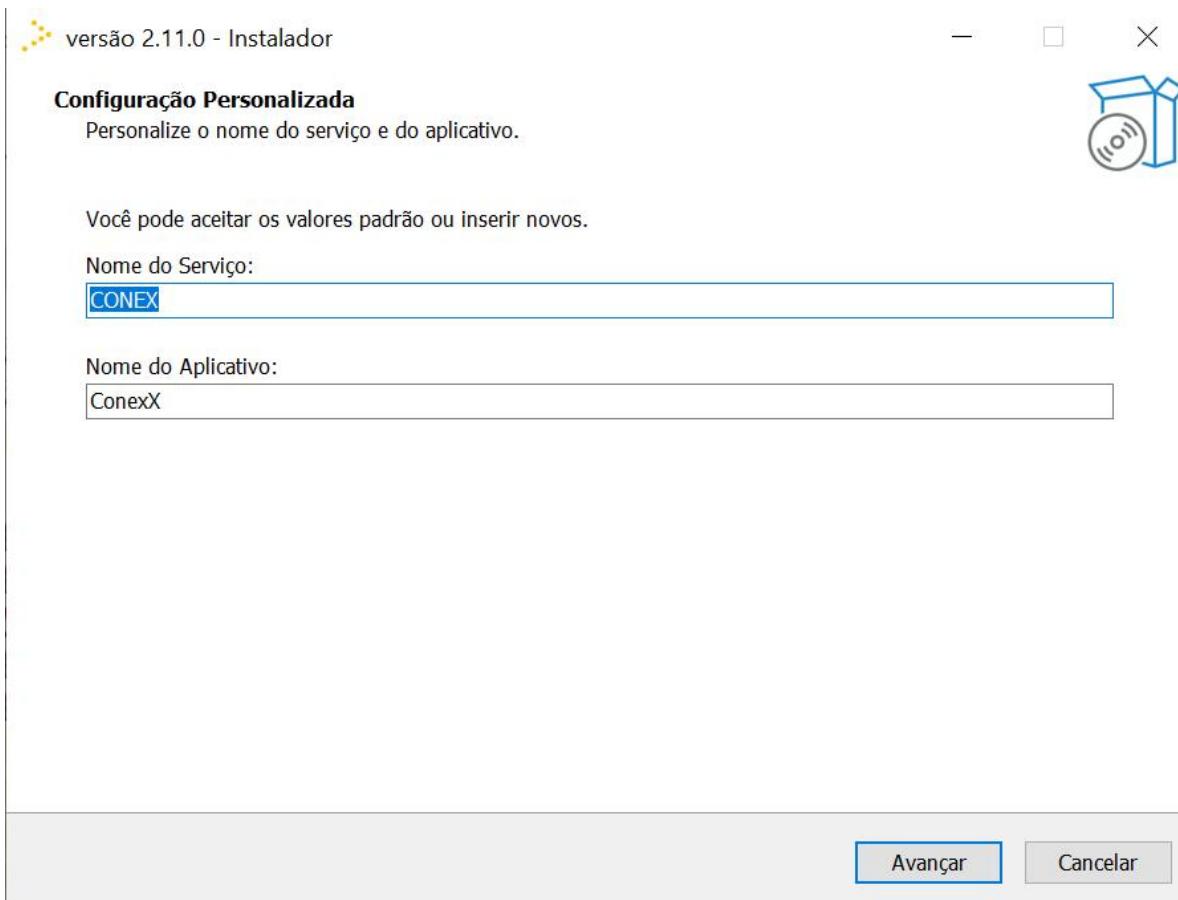
1.1. Instalação ConexX_Driver

1. Execute o Arquivo "ConexService-Installer.Exe", Selecione o Idioma do Instalador e prossiga na instalação



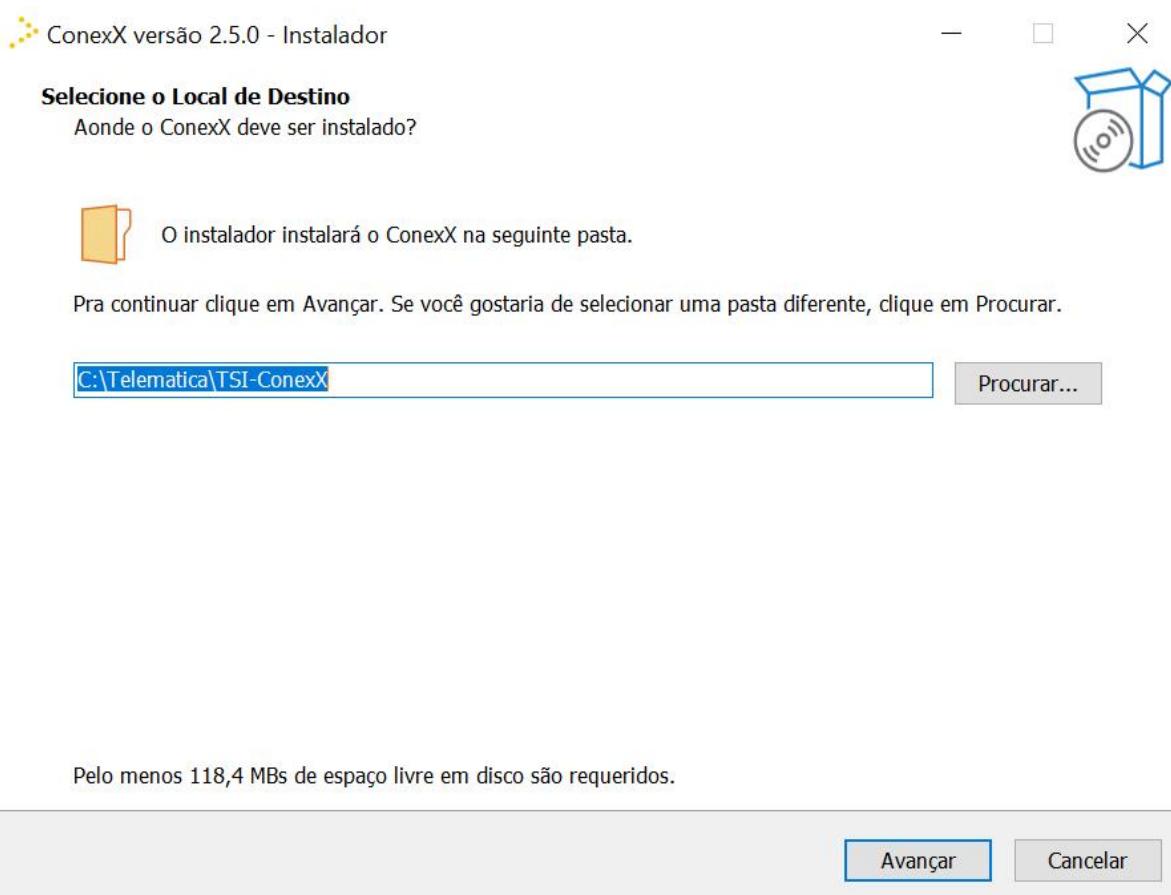
Tela - Instalador ConexX

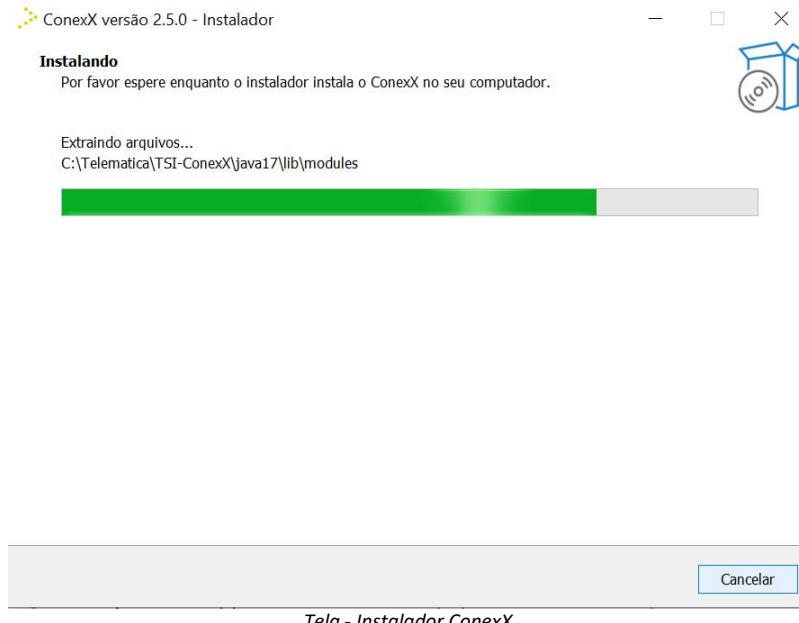
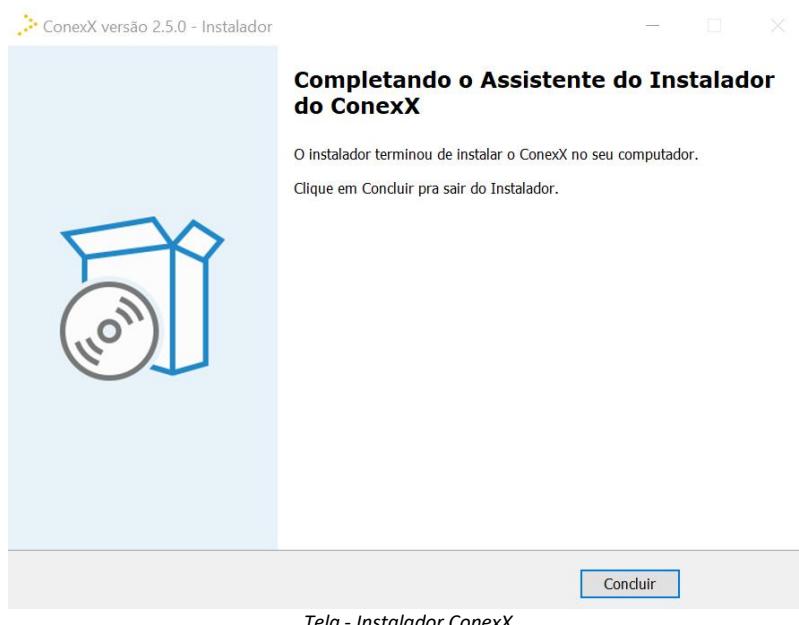
2. Podemos definir se o Nome do Serviço e o Nome do Aplicativo do serviço será CONEX ou podemos modificar para se adequar as necessidades do cliente, podendo possuir mais de um ConexX_Driver por maquina.



Tela - Instalador ConexX

**3. Defina o Local de Instalação do ConeX, por padrão ele é instalado em: C:\Telematica\TSI-ConexX
recomendado que fique na Mesma Pasta do ConexX_MiddleWare**



4. Ao Selecionar o Local da Instalação, será iniciado a Extração dos Arquivos e a instalação do Serviço**5. Finalizando A Extração será Exibido o Término da Instalação****6. Verifique em "Serviços" se o Serviço do ConexX está sendo Executado**

1.1.1. Configuração ConexX_Driver

Para realizar a Configuração do ConeX_Driver, Localize o Arquivo "**conex.properties**" e Altere os Seguintes Parâmetros conforme Necessário

```
# Porta de API do conex, para realizar a comunicação com o ConexX_Middleware, ex: 59974
• api.port=
# Porta para comunicação com equipamentos, ex: 1364
• conex.port=
# CPF padrão utilizado nos comandos em REP INMETRO caso não seja informado nas requisições da api. Ex:
12345678901
• conex.cpf.responsavel=

# Informar se envia ou não auditoria TSI, Ex: false
• tsi.sendWebhookToTSI=false
# Url de auditoria TSI Ex: https://suricato-light.vercel.app/api/webhook
• tsi.webhookUrl=https://suricato-light.vercel.app/api/webhook

# Middleware Uri "http://IPDoMiddleware:8090"
• partner.uri= ex: http://192.168.0.200:8090
# Tempo para reprocessamento dos webhooks Ex: 1
• webhook.reprocessIntervalInMinutes=1

# Lista de equipamentos bloqueados, caso seja preenchido, aceitará os equipamentos que não forem adicionados
• conex.listaEquipamentosBloqueados= ex: 192.168.0.100,192.168.0.101
# Lista de equipamentos, liberados, caso seja preenchido, só aceitará os equipamentos adicionados
• conex.listaEquipamentosLiberados= ex: 192.168.0.100,192.168.0.101
# Tempo em minutos para derrubar o equipamento após a conexão, Ex: 30
• device.telematica.rep.inmetro.disconnect.timeout=

logins=
• dev=true
• dev.checkdApiDevicesTest=

# Emulador
# Des/ativar
• emulator.enabled=
# Tempo de intervalo para subir para cada equipamento em milisegundos(0 se ninformado)
• emulator.device.connetion.delay=3000
# Quantidade de MDs para subir
• emulator.md=4
# MD: Tempo de delay em acessos, 30 + valor informado
• emulator.md.access.random.delay=
```

1.1.2. Desinstalar ConexX_Driver

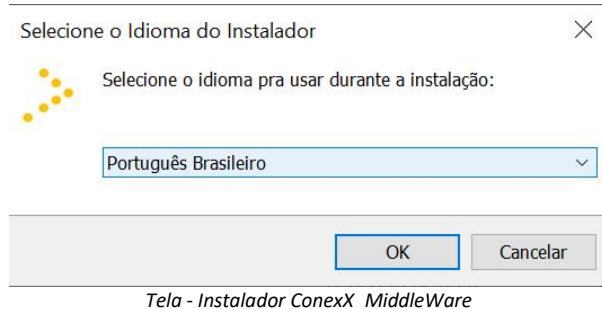
Para Realizar a Desinstalação do serviço do ConexX, Pare o Serviço do Mesmo, e Execute o Arquivo "unins000.exe"

| Disco Local (C:) > Telematica > TSI-ConexX > | | | |
|--|--------------|---------------------|-------------------|
| | Nome | Data de modificação | Tipo |
| | | | Tamanho |
| | ConexService | 18/12/2024 17:18 | Pasta de arquivos |
| | java17 | 18/12/2024 17:09 | Pasta de arquivos |
| | unins000.dat | 18/12/2024 17:09 | Arquivo DAT |
| | unins000.exe | 18/12/2024 17:08 | Aplicativo |
| | | | 93 KB |
| | | | 3.262 KB |

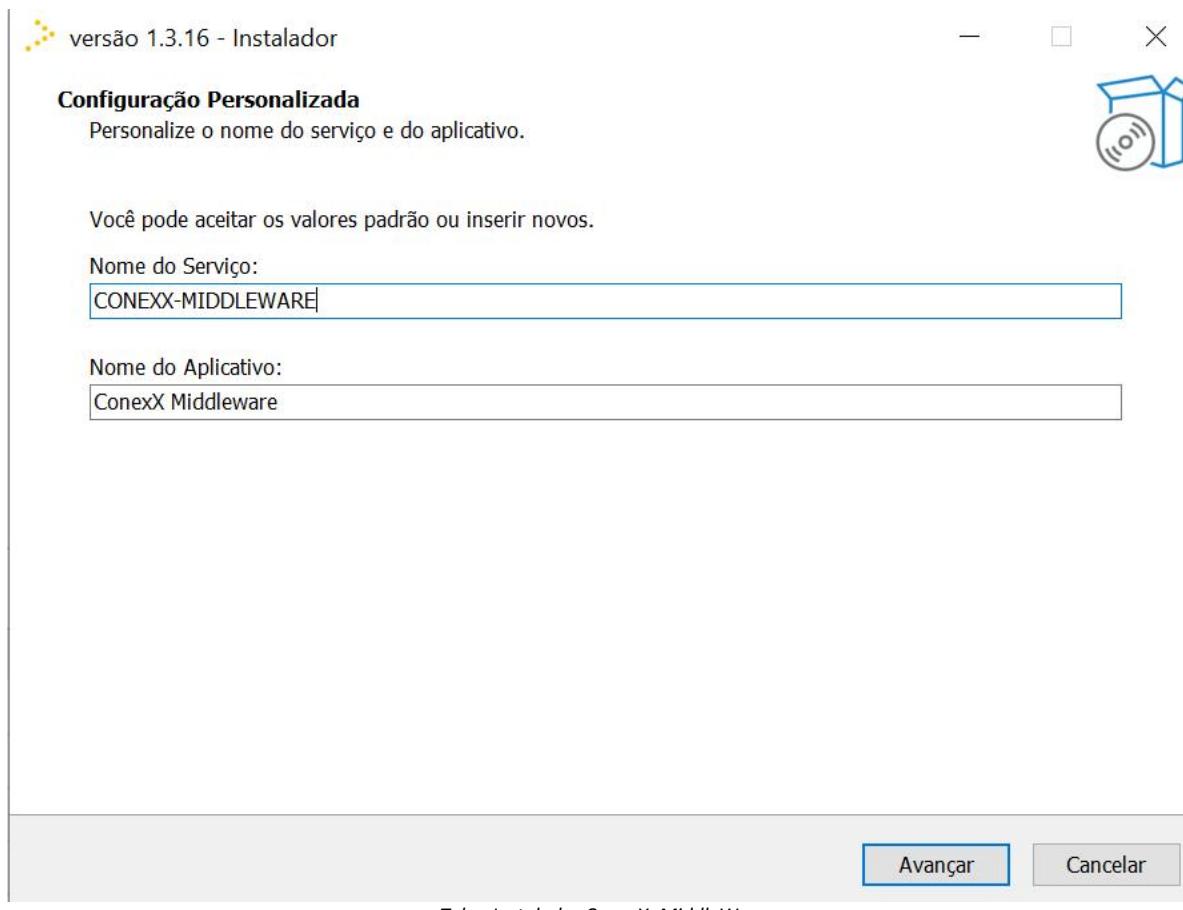
Tela - Diretório ConexX

1.2. Instalação ConexX_Middleware

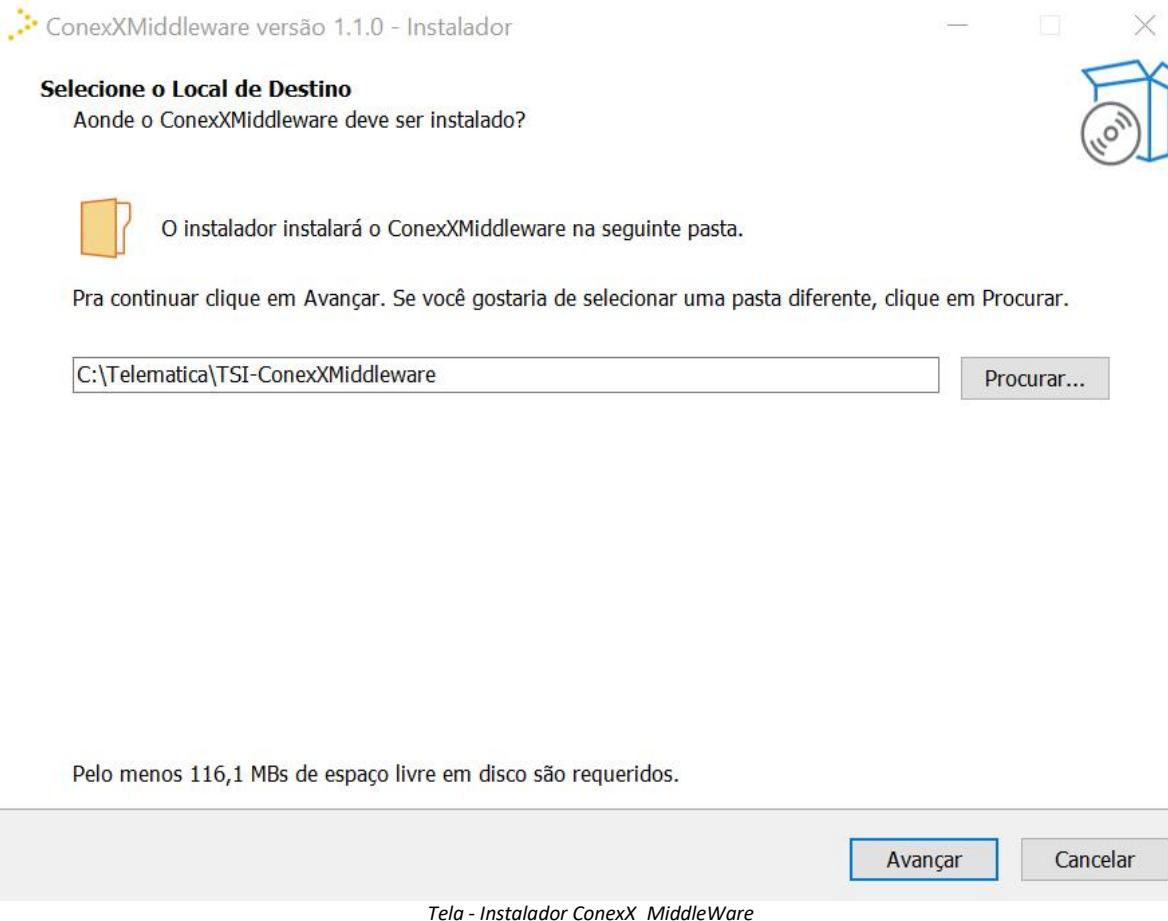
1. Execute o Arquivo "ConexXMiddlewareService-Installer.Exe", Selecione o Idioma do Instalador e prossiga na instalação

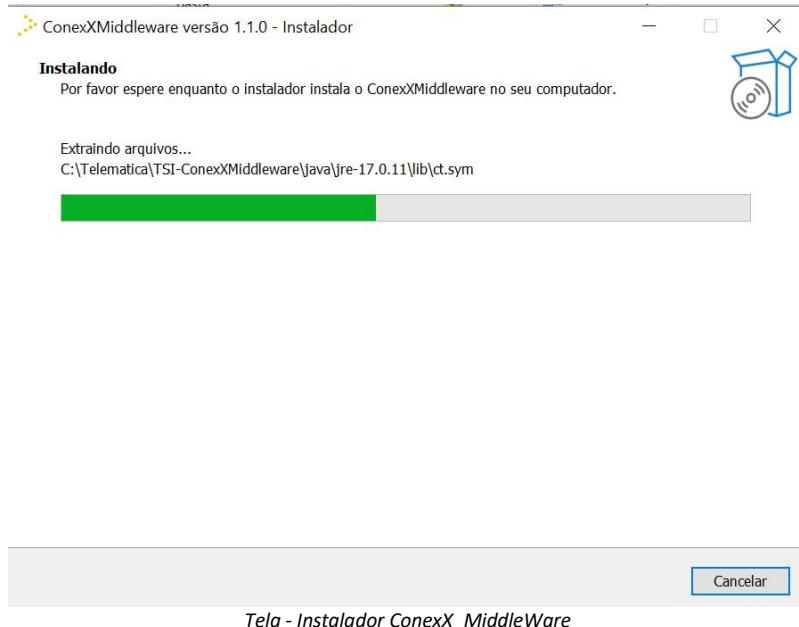


2. Podemos definir se o Nome do Serviço e o Nome do Aplicativo do serviço será CONEXX-MIDDLEWARE ou podemos modificar para se adequar as necessidades do cliente, podendo possuir mais de um ConexX_MIDDLEWARE por maquina.

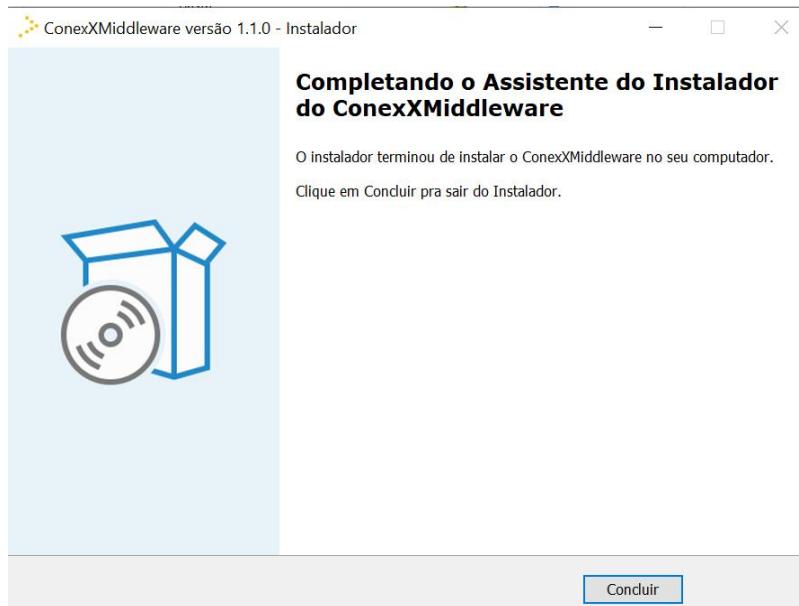


3. Defina o Local de Instalação do ConexXMiddleWare, por padrão ele é instalado em:
C:\Telematica\TSI-ConexXMiddleware
recomendado que fique na Mesma Pasta do ConexX



4. Ao Selecionar o Local da Instalação, será iniciado a Extração dos Arquivos e a instalação do Serviço

Tela - Instalador ConexX_MiddleWare

5. Finalizando A Extração será Exibido o Término da Instalação

Tela - Instalador ConexX_MiddleWare

6. Verifique em "Serviços" se o Serviço do ConexX_Middleware está sendo Executado

Conex X Middleware

Middleware de comunicação do ConexX com SeniorX

Em Execução Automático

Serviço local

Tela - Serviço ConexX_MiddleWare

1.2.1. Configuração ConexX_Middleware

A configuração da Porta do Servidor do ConexX_Middleware é efetuado no Arquivo "**application.properties**" por padrão o serv.port é configurado como 8090

```
#Log
• spring.main.banner-mode=off
#Configuração do Nome que será exibido no Log
• spring.application.name=ConexX-Middleware
#Caminho do schema do Swagger
• springdoc.api-docs.path=/api-docs
#Configuração do Spring
• spring.devtools.restart.enabled=false
#redirect do link [swaggerui/index.html] do swagger Ex:192.168.0.200:8090/swagger-ui/index.html
• springdoc.swagger-ui.path=/

#Driver de comunicação, porta de comunicação com o ConexX_Driver
• server.port=8090

# Configuration ConexX_Middleware
#Configuração do Parceiro (Sara/SeniorX/SeniorXT)
• md.tenant.client=sara
#Ip do middleware
• md.conex.api.url= ex: http://26.224.145.69:59972
#Configuração de requisições dentro do middleware
• md.restTemplateTimeWaitSecond=15
#Configuração de requisições dentro do middleware
• md.restTemplateReadTimeoutSecond=15
#Timezone com padrão Continente/estado
• spring.jackson.time-zone=America/Sao_Paulo
#Formato de comunicação de Json dentro da aplicação e trabalha junto com spring.jackson.time-zone
• spring.jackson.serialization.write-dates-as-timestamps=false

#Campo para definir o Nível do Log do Serviço
• logging.level.org.springframework.web.filter.CommonsRequestLoggingFilter=DEBUG
```

Abaixo teremos as configurações de Acordo com o campo **md.tenant.client**

```
# Sara (Totvs)
#Username da api, Gerado pelo parceiro
• sara.api.authUsername=telematica
#senha da API, Gerado pelo parceiro
• sara.api.authPassword=tele@1234
#Lista de tipo de equipamento
• sara.api.manufacturerList=telematica,dimep,henry
#Tempo do KeepAlive em segundos
• sara.api.keepAliveDeviceSeconds=200
#URL de comunicação
```

- sara.api.baseUrlAccess=https://sandbox.tlra.totvs.com.br
- #URL de comunicação
- sara.api.baseUrlBiometry=https://sandbox.tlra.totvs.com.br

SeniorX (utilizando servidor Cloud da Senior)

#Tempo de consulta de pendencias dentro do Senior

- senior.api.timePoolingPendencySeconds=300

#Tempo do KeepAlive em segundos

- senior.api.timeDriverAliveSeniorSeconds=60

#Tempo do KeepAlive em segundos

- senior.api.timeDeviceAliveSeniorSeconds=30

#CPF do Responsável, valor deve ser igual o configurado no Driver

- senior.api.valuePropertyCpfResponsible=CPF

#Configuração do REP P

- senior.api.valueDeviceFeature=Tipo Gerenciador

#Configuração de Acordo Coletivo

- senior.api.valuePropertyTypeAgreementREP=Tipo de Acordo

#URL de comunicação

- senior.api.sdkSenior=https://sam-api.senior.com.br/sdk/v1

#Driver key gerado pela senior

- senior.api.driverKey=eyJhbGciOiJIUzI1NiJ9eyJkcmI2ZXJJZCI6MSwidGVuYW50IjoidGVsZW1hdGljYS1kZXZjb21iciJ9.jYeF4Pz_4-QzjEKZL1noknlu1NSZJ0B-Du0chzKT08M

SeniorXT (Protocolo 8, instalado no cliente)

#Configuração de Driver do XT, Gerado pela senior

- senior.protocolo.driver=800

#Ip da concentradora/seniorXT

- senior.protocolo.socketHost=100.126.147.29

#Porta da concentradora/seniorXT

- senior.protocolo.socketPort=61000

#Porta do Facial

- senior.protocolo.facialPort=8088

#Certificado para identificação do middleware Telematica com a seniorXT

- senior.protocolo.certificate=TSI.CER

#Tempo do KeepAlive em segundos, Tempo de espera para reconexão do Socket com a SeniorXT

- senior.protocolo.timeOutAwaitSocketSecond=60

#Tempo do KeepAlive em segundos, Tempo de Busca e verificação dos equipamentos

- senior.protocolo.timeOutAwaitSearchDeviceSecond=5

#Tempo do KeepAlive em segundos, Tempo até o equipamento retornar para o middleware

- senior.protocolo.scheduledKeepAliveDeviceSecond=30

obs: não alterar o Campo senior.protocolo.certificate=TSI.CER

1.2.2. Desinstalar ConexX Middleware

Para Realizar a Desinstalação do serviço do ConexX Middleware, Pare o Serviço do Mesmo, e Execute o Arquivo "unins000.exe"

|) > Telematica > TSI-ConexXMiddleware | | | |
|---------------------------------------|---------------------|-------------------|----------|
| Nome | Data de modificação | Tipo | Tamanho |
| ConexXMiddlewareService | 19/12/2024 06:57 | Pasta de arquivos | |
| java | 18/12/2024 17:04 | Pasta de arquivos | |
| unins000.dat | 18/12/2024 17:04 | Arquivo DAT | 116 KB |
| unins000.exe | 18/12/2024 17:03 | Aplicativo | 3.262 KB |

Tela - Diretório ConexX_MiddleWare

1.3. Validação da Licença

Para utilizar os serviço do ConexX_Driver será necessário realizar a validação do arquivo "**Conex.req**" que será gerado ao iniciar o serviço atualmente a validação da licença está sendo realizada de forma manual, portanto será necessário realizar o envio do arquivo para gerarmos o "**Conex.res**" para liberar a utilização do serviço Ambos os arquivos devem estar no mesmo diretório, por padrão o "Conex.req" será gerado na basta "base" do ConexX_Driver

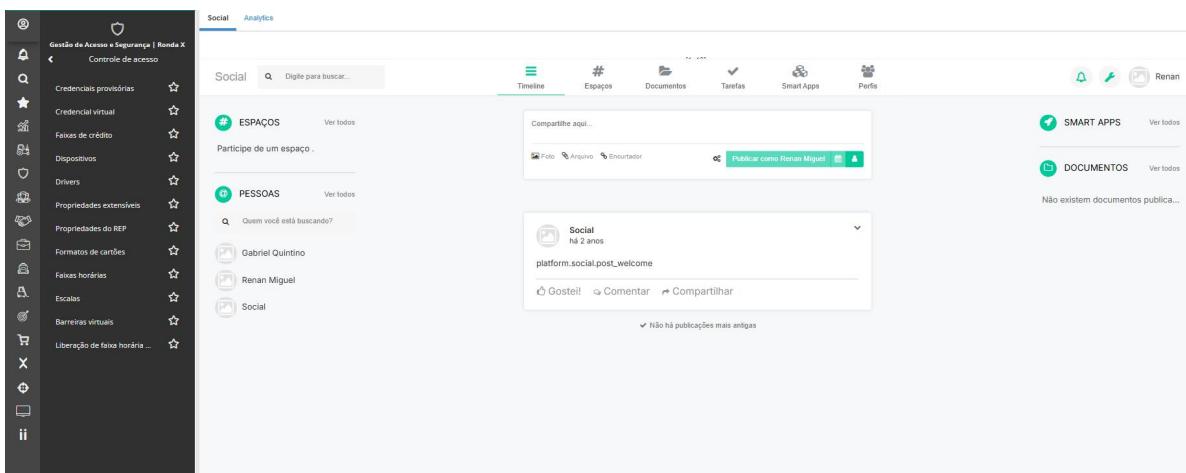
2. Cadastro Equipamento (ConexX_Driver P5)

O cadastro dos Equipamentos será realizado na plataforma da Senior, na qual possui um treinamento próprio para utilizá-la, mas todo o Cadastro, Alteração e configuração do Equipamento será realizado pela própria plataforma

Para evitar problemas de comunicações entre a plataforma e o equipamento realizar a Configuração do Equipamento de acordo com as especificações do Equipamento físico

O cadastro de equipamento será realizado no menu lateral (**Gestão de Acesso e Segurança -> Controle de Acesso -> Dispositivos**)

Clique em Adicionar gerenciador para realizar o cadastro de um novo dispositivo ou selecione um dispositivo para realizar a sua alteração.



Tela - Menu inicial

Preencha os Campos conforme as necessidades da situação e de acordo com as informações do equipamento físico, para adicionar uma leitora no equipamento selecione o tipo de leitora em (**Dispositivos conectados**)
 Para Utilizar o Campo Mensagem de Acesso será necessário possuir uma versão mínima de firmware, na qual podemos conferir no item ([Firmwares Compatíveis com o ConexX](#)), as mensagens inicialmente serão iguais as do Suricato porem são customizaveis

Cadastro de dispositivos
Consulta de dispositivos • MD 400 203

GERENCIADOR

Código
5
Descrição *
MD 400 203
Local físico *
TSI
Identificação da rede *
192.168.0.203
Tipo gerenciador *
Controle de acesso
Usuário

Driver *
Telemática Driver 01
Porta de comunicação *
3001
Mensagem do display
TSI MD400
Propriedades extensíveis
Monitoramento de Alunos
Senha

CONTROLE DE ACESSO

Tempo burla cerraca em segundos *
0

MENSAGENS DE ACESSO

DISPOSITIVOS CONECTADOS

- Biometria 1
- Leitora de QRCode
- Leitora Entrada

Tela - Cadastro Equipamento

Podemos Adicionar ao dispositivo alguns sub-níveis e realizar o envio de Comando ao Equipamento

CONTROLE DE ACESSO

Tempo burla cerraca em segundos *
0

MENSAGENS DE ACESSO

DISPOSITIVOS CONECTADOS

- Biometria 1
- Leitora de QRCode
- Leitora Entrada

EVENTOS MONITORADOS

ACESSOS MONITORADOS

ENVIO DE COMANDO

Lectora
Enviar comando
Entrada
Saída
Teclado
Salvar Excluir Adicionar subnível Cancelar

Tela - Cadastro Equipamento

Preencha os Campos conforme as necessidades da situação e de acordo com as informações do equipamento físico

Tela - Cadastro Equipamento

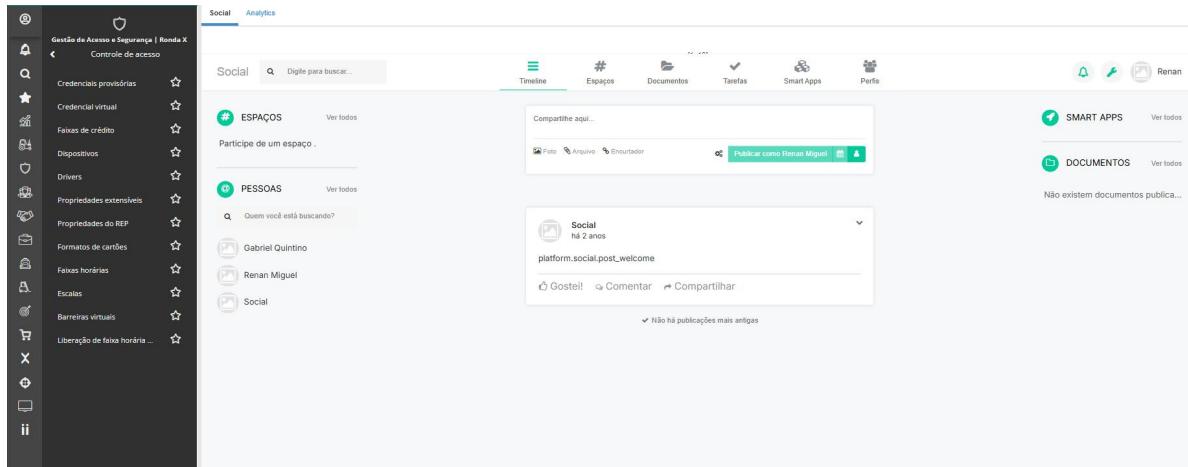
Preencha os Campos conforme as necessidades da situação e de acordo com as informações do equipamento físico, o controle de acesso podemos selecionar a o sentido do acesso, se atentar a não informar erroneamente na plataforma e causar uma falha de comunicação com o equipamento

Tela - Cadastro Equipamento

2.1. Propriedades Extensíveis

No ConexX_Driver (P5) podemos realizar o cadastro de um grupo de Propriedade Extensiva e aplica-la em todos os equipamentos, sem a necessidade de realizar a configuração individual em cada equipamento

Podemos acessar as Propriedades Extensíveis pelo Menu lateral (**Gestão de Acesso e Segurança -> Controle de Acesso -> Propriedades extensíveis**) nesse menu iremos realizar o cadastro e parametrização das propriedades Extensíveis



Tela - Menu inical

Na tela de Grupo de propriedades extensíveis podemos realizar o cadastro, Edição e exclusão dos Grupo de propriedades extensíveis

Tela - Cadastro Propriedades Extensíveis

Ao selecionar um Grupo de Propriedades extensíveis podemos adicionais propriedades Extensíveis a esse grupo, fazendo com o que o Equipamento esteja com todas as propriedades extensíveis registrado no grupo

| | Identificador | Valor |
|--------------------------|-------------------------|-------|
| <input type="checkbox"/> | Alarme de Porta Aberta | 4-101 |
| <input type="checkbox"/> | Alarme de Porta Forçada | 5-102 |
| <input type="checkbox"/> | Alarme de Tampa Aberta | 6-103 |

Quantidade de registros: 3

Tela - Cadastro Propriedades Extensíveis

No cadastro de propriedades extensíveis na plataforma existe o campo Identificador e Valor, o Identificador fica a critério do usuário descrever o que deseja sobre o alarme no campo Valor é Imprescindível o modelo de configuração para que o Middleware possa comunicar de forma adequada com o ConexX.

| Identificador | Valor |
|------------------------|-------|
| Alarme de Porta Aberta | 4-101 |

Created by renan.miguel@telematica-dev.com.br on 19/09/2024 at 10:04. Updated by renan.miguel@telematica-dev.com.br on 20/09/2024 at 15:49.

Tela - Cadastro Propriedades Extensíveis

Na tela de Cadastro de Dispositivos, os Grupo de propriedades extensíveis serão exibidos como uma lista de escolha para selecionar um grupo conforme a necessidade

Tela - Cadastro Equipamento

3. Cadastro Equipamento (ConexT_Driver P8)

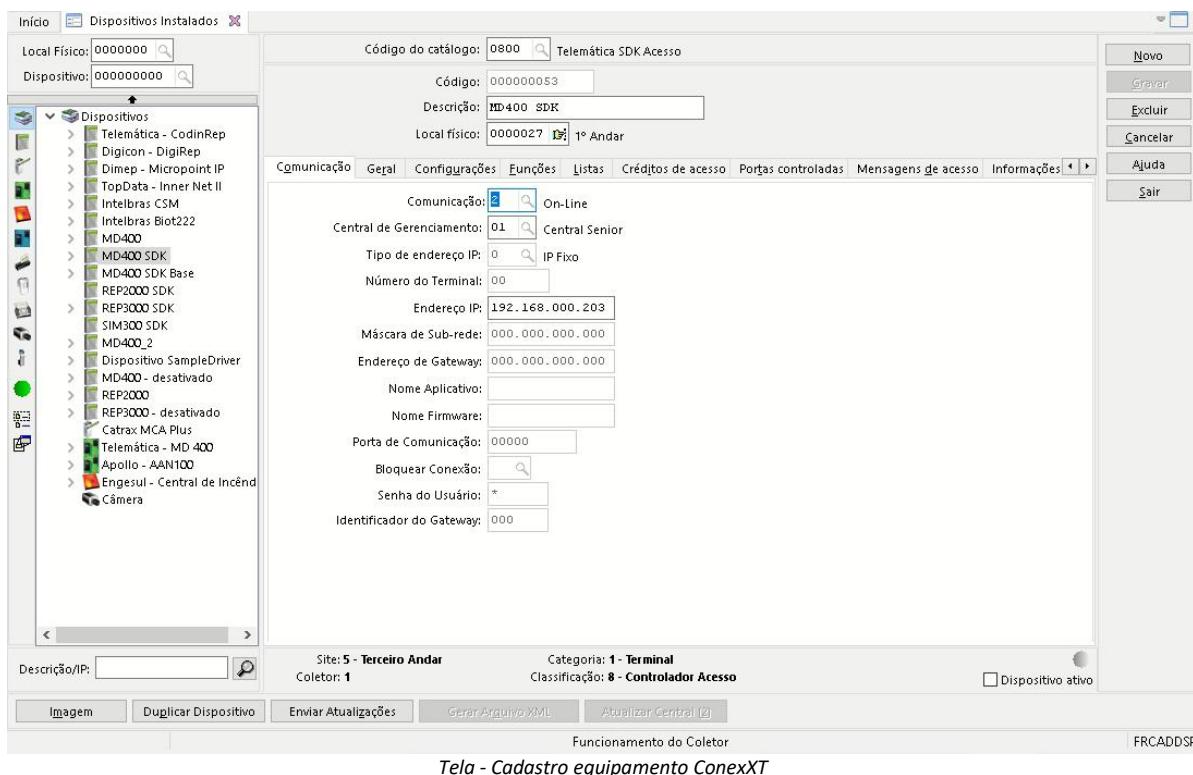
O cadastro dos Equipamentos será realizado na plataforma da Senior, na qual possui um treinamento próprio para utilizá-la, mas todo o Cadastro, Alteração e configuração do Equipamento será realizado pela própria plataforma

Para evitar problemas de comunicações entre a plataforma e o equipamento realizar a Configuração do Equipamento de acordo com as especificações do Equipamento físico

Podemos efetuar o Cadastro dos Equipamentos utilizando o menu lateral em (**Menu -> Dispositivos -> cadastro**), no menu a direita podemos efetuar o cadastro, exclusão e alteração

Para Utilizar o Campo Mensagem de Acesso será necessário possuir uma versão mínima de firmware, na qual podemos conferir no item (Firmwares Compatíveis com o ConexX), as mensagens inicialmente serão iguais as do Suricato porem são customizaveis

Preencha os Campos conforme as necessidades da situação e de acordo com as informações do equipamento físico



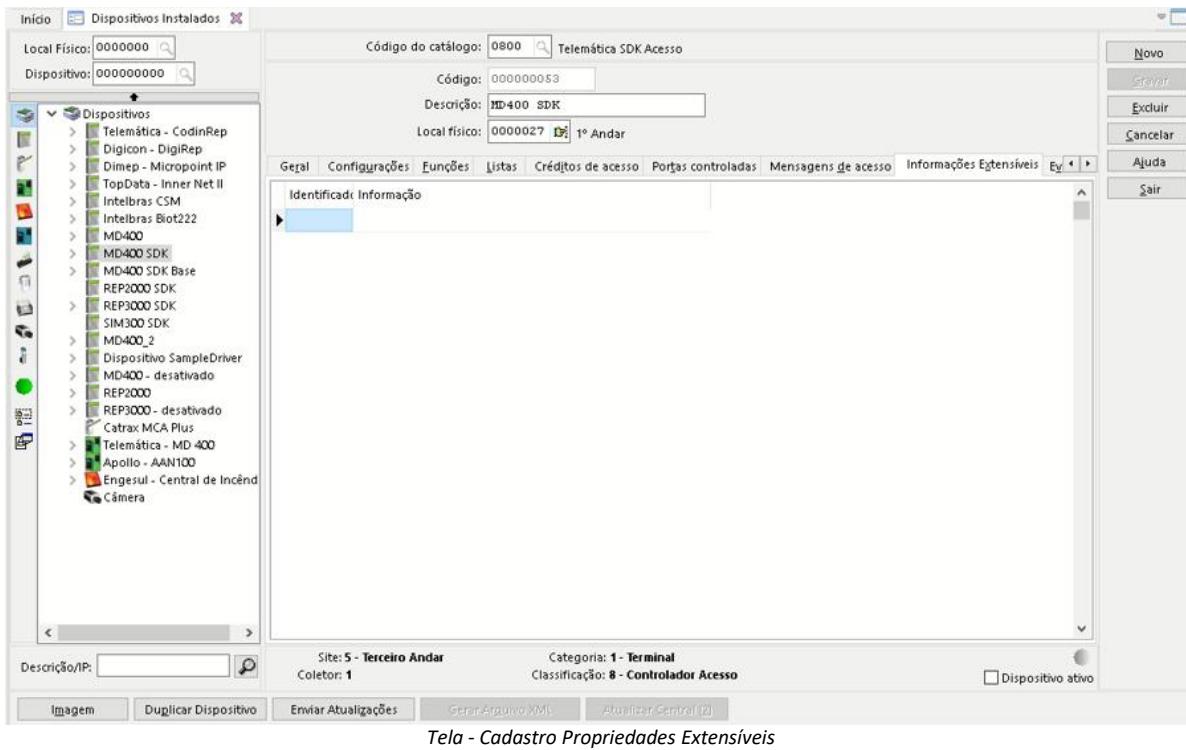
Preencha os Campos conforme as necessidades da situação e de acordo com as informações do equipamento físico

Início Dispositivos Instalados

| | | |
|--|----------|---------------------------------|
| Local Físico: | 00000000 | <input type="button" value=""/> |
| Dispositivo: | 00000000 | <input type="button" value=""/> |
| <ul style="list-style-type: none"> > Dispositivos <ul style="list-style-type: none"> > Telemática - CodinRep > Digicon - DigiRep > Dimep - Micropoint IP > TopData - Inner Net II > Intelbras CSM > Intelbras Biot222 > MD400 > MD400 SDK <ul style="list-style-type: none"> Biometria MD400 Leitora Crachá Acesso E > MD400 SDK Base > REP2000 SDK > REP3000 SDK > SIM300 SDK > MD400_2 > Dispositivo SampleDriver > MD400 - desativado > REP2000 > REP3000 - desativado Catrax MCA Plus Telemática - MD 400 Apollo - AAN100 Engesul - Central de Incêndio Câmera | | |
| Código do catálogo: 1051 <input type="button" value=""/> Leitora Suprema RepBio Código: 000000060 Descrição: Biometria MD400 Local físico: 0000027 <input type="button" value=""/> 1º Andar | | |
| 1 Cadastro 2 Validações 3 Direção e Nível 4 Acionamentos Acesso 5 Formatos de cartões 6 Informações Extensíveis 7 Pessoas Contratadas | | |
| Código da leitora: 023 Localização da leitora: 00 <input type="button" value=""/> Endereço de comunicação: 000 Tipo leitora: 4 <input type="button" value=""/> Biométrico Tipo de validação: 1 <input type="button" value=""/> Remoto Endereço MRA: 0000000000 Endereço TCP/IP Câmera LPR: 000..000.000.000 Leitora embarcada: <input type="button" value=""/> Notificar acesso após cido: <input type="button" value=""/> Tempo para acesso: 00 segundo(s) Alarme porta aberta após: 00 segundo(s) Trabalha em paridade: <input type="button" value=""/> Tipo de validação on-line: 0 <input type="button" value=""/> Bloqueia acesso Tipo de validação off-line: 0 <input type="button" value=""/> Bloqueia acesso Quantidade de pessoas escoltadas: 000 Tecnologia biométrica: 06 <input type="button" value=""/> Suprema Nível de conferência biometria (%): 100 | | |
| Descrição/ID: <input type="text"/> <input type="button" value=""/> Dispositivo: Leitora Categoria: 5 - Leitor Classificação: 99 - Outros <input type="checkbox" value=""/> Dispositivo ativo | | |
| <input type="button" value="Imagem"/> <input type="button" value="Duplicar Dispositivo"/> <input type="button" value="Enviar Atualizações"/> <input type="button" value="Gerar Arquivo XML"/> <input type="button" value="Atualizar Central (2)"/> | | |

Rua Achilles Orlando Curtolo, 597 - Barra-Fundão
CEP 01144-010 | São Paulo/SP
Contato: (11) 3933-6363 / crm@telematica.com.br

No ConexXT_Driver (P8) diferente do ConexX_Driver (P5), não podemos realizar a Configuração das Propriedades Extensíveis de todos os equipamentos em conjunto, a configuração deve ser feito de forma individual



Tela - Cadastro Propriedades Extensíveis

4. Teste de Ambiente

Podemos realizar o Teste utilizando um Swagger desenvolvido para validação dos dados, para acessa-lo será necessário possuir um local com o ConexX_Middleware instalado e configurado

No swagger podemos realizar a Validação de envio e recepção de dados, sendo elas os Post para validação de envio de dados e o Get para validação dos dados recebidos

The screenshot shows the Swagger UI interface for the 'Suricato Sara' driver. At the top, it displays the title 'Suricato Sara' and 'Desconhecido OAS 3.0'. Below this, there's a brief description: 'Driver de comunicação do ConexX com Serviço de Gestão de Acesso SARA. | Usuário: telematica'. There are links to 'Acessar página de logs' and 'Configurações'. The main area is titled 'Servers' and shows 'http://192.168.0.200:8090 - Generated server url'. The API documentation is organized into sections: 'conex-md-controller'. Under this section, there are several API endpoints listed with their methods and URLs:

- POST /api/conex/webhook**
- POST /api/conex/biometricActivation** Ativar leitor biométrico
- POST /api/conex/accessValidation**
- GET /api/conex/getPersonImage**
- GET /api/conex/getDataPerson**
- GET /api/conex/devices** Retrieve all devices
- DELETE /api/conex/clean-pendency**

Para realizar a validação dos dados no swagger selecione o qual a api na qual será utilizada, após a seleção serão exibidos um exemplo de dados da api e qual resposta será devolvido dependendo do resultado, clicando no botão "Try it out" podemos inserir valores e validar os dados

This screenshot provides a detailed view of the 'POST /api/conex/webhook' endpoint. It includes the following sections:

- Parameters**: No parameters.
- Request body (required)**: Shows an example value in JSON format:

```
{
  "deviceIpAddress": "192.168.0.1",
  "error": false,
  "messageId": "12345",
  "message": "Some message",
  "datePost": "2024-07-30T12:34:56",
  "processDate": "2024-07-30T12:34:56",
  "processId": "2024-07-30T12:30:00",
  "data": "Some data",
  "information": "string"
}
```
- Responses**:
 - 200 OK**: Media type: application/json. Example Value: 0
 - 400 Bad request**: Media type: application/json. Example Value: 0

Preenchendo os dados conforme necessário clique em "Execute" para realizar a validação, será exibido o Curl do Post ou Get, dependendo da API selecionado, e o seu resultado

```

POST /api/conex/webhook
Parameters
No parameters
Request body required
application/json
{
    "deviceIpAddress": "192.168.0.1",
    "error": false,
    "dependencyId": "12345",
    "message": "Some message",
    "datePost": "2024-07-30T12:34:56",
    "processStart": "2024-07-30T12:00:00",
    "processEnd": "2024-07-30T12:30:00",
    "dependencyType": "SOME_TYPE",
    "data": {},
    "information": "string"
}
  
```

Responses

```

curl -X 'POST' \
  'http://192.168.0.200:8000/api/conex/webhook' \
  -H 'accept: */*' \
  -H 'Content-Type: application/json' \
  -d '{
    "deviceIpAddress": "192.168.0.1",
    "error": false,
    "dependencyId": "12345",
    "message": "Some message",
    "datePost": "2024-07-30T12:34:56",
    "processStart": "2024-07-30T12:00:00",
    "processEnd": "2024-07-30T12:30:00",
    "dependencyType": "SOME_TYPE",
    "data": {},
    "information": "string"
}'
  
```

Request URL

Server response

Code **Details**

Undocumented Failed to fetch.
Possible Reasons:

- CORS
- Network Failure
- URL scheme must be "http" or "https" for CORS request.

Na seção Schemas podemos visualizar exemplos de estruturas para diversas situações, o tipo do campo e exemplo de dado a ser enviado

```

Webhookmodel ▾ {
  description: Model for generic webhook
  deviceIpAddress*: string
  example: 192.168.0.1
  IP address of the device
  error*: boolean
  example: false
  Indicates if there was an error
  dependencyId: string
  example: 12345
  ID of the dependency
  message*: string
  example: Some message
  Message associated with the webhook
  datePost*: string
  example: 2024-07-30T12:34:56
  Date when the post was made
  processStart*: string
  example: 2024-07-30T12:00:00
  Start time of the process
  processEnd*: string
  example: 2024-07-30T12:30:00
  End time of the process
  dependencyType*: string
  example: SOME_TYPE
  Type of dependency
  data: ▾ {
    description: Additional data
  }
  information: string
}
  
```

5. Propriedades Extensíveis para Alarmes

- Esse Tópico refere-se a como o Middleware, ConexX_Driver irão trabalhar para realizar a comunicação de alarmes com
- Como os equipamentos trabalham com enumeração de portas e elas podem ser configuradas da maneira que o usuário deseja
- Como não existe uma porta específica para cada Alarme, a porta vai depender do equipamento será adotado o seguinte padrão.
- O campo valor é obrigatório deve seguir esse padrão, Numero da Porta (NP) e Código do Alarme (CA) sempre no formato NP - CA Alarme de
- No cadastro de propriedades extensíveis na plataforma existe o campo Identificador e Valor, o Identificador fica a critério do usuário descrever o que deseja sobre o alarme no campo Valor é Imprescindível o modelo de configuração para que o Middleware possa comunicar de forma adequada com o ConexX.

Exemplo de como deve ser preenchido:

- Exemplo - Nome para identificador | Valor: Porta do Equipamento (NP) - Código do Alarme (CA)
- Identificador: Alarme de Porta Aberta | Valor: 1 - 101

Valor de acordo com a portas dos equipamentos da Telemática

- 1 - Porta 1
- 2 - Porta 2
- 3 - Porta 3
- 4 - Porta 4
- 5 - Porta 5
- 6 - Porta 6
- 7 - Porta 7
- 8 - Porta 8
- 9 - Porta 9
- 10 - Porta 10
- 11 - Porta 11
- 12 - Porta 12
- 13 - Porta 13
- 14 - Porta 14
- 15 - Porta 15
- 16 - Porta 16
- 17 - Porta 17
- 18 - Porta 18
- 19 - Porta 19
- 20 - Porta 20
- 21 - Porta 21
- 22 - Porta 22
- 23 - Porta 23
- 24 - Porta 24

Código de Alarmes de acordo com a plataforma

- 101 - "Porta Aberta"
- 102 - "Porta Forçada"
- 103 - "Tampa Aberta"
- 104 - "Dispositivo Iniciado"
- 105 - "Dispositivo Online"
- 106 - "Dispositivo Offline"
- 107 - "Emissão de relatório de ponto e assiduidade"
- 108 - "Iniciado o Uso da porta Fiscal USB"
- 109 - "Bobina de Papel Substituída"
- 110 - "Falta de Papel"
- 111 - "Pouca Memoria"
- 112 - "Sem Memoria"
- 113 - "Bateria Substituída"
- 114 - "Iniciado o Uso da Bateria"
- 115 - "Iniciado o uso de Energia"
- 116 - "Equipamento Travado"
- 117 - "Equipamento destravado"
- 118 - "Alarme não mapeado"
- 119 - "Alarme não mapeado"
- 120 - "Alarme não mapeado"
- 121 - "Alarme não mapeado"
- 122 - "Alarme não mapeado"
- 123 - "Alarme não mapeado"
- 124 - "Alarme não mapeado"

6. LOG

Para verificarmos o Procedimento do Serviço podemos visualizar nos Logs disponibilizados dentro do diretório do serviço ou podemos visualizar o Log pelo Swagger

- Diretório do ConexX "...\\TSI-ConexService\\log"
- Diretório do MiddlewareX "...\\TSI-ConexXMiddleware\\log"

As principais informações do log serão encontradas nos arquivos ".stdout", recomenda-se utilizar o "baretail.exe" na hora de visualizar o log para sempre ter o log atualizado

Os logs sempre seguem esse padrão: **Data | Tipo do Log | Numero do Serviço | Nome da Thread | Classe de Execução**

Log completo:

- 2025-05-05 01:48:34.965 | ERROR | PID[28220] | Th[http-nio-8082-exec-7] - CI[.c.t.c.m.p.c.h.ConexManager] > * Device Status - 192.168.64.186 - DEVICE_STATUS - Device with IP 192.168.64.186 not found.

Data | Tipo do Log | Numero do Serviço | Nome da Thread | Classe de Execução

- 2025-05-05 01:48:34.965 | ERROR | PID[28220] | Th[http-nio-8082-exec-7] - CI[.c.t.c.m.p.c.h.ConexManager]

Aqui tem um padrão mas tem o detalhe do que aconteceu:

Nome da função - IP ou Id do equipamento com o tipo do evento - E o detalhe

- > * Device Status - 192.168.64.186 - DEVICE_STATUS - Device with IP 192.168.64.186 not found.

Abaixo teremos alguns exemplos de log

Erro vindo da requisição feita na api da Senior

- 2025-05-05 02:48:35.951 | ERROR | PID[28220] | Th[http-nio-8082-exec-8] - CI[.c.t.c.m.p.c.h.ConexManager] >

* Device Status - 192.168.64.186 - DEVICE_STATUS - Device with IP 192.168.64.186 not found.

- 2025-05-05 08:59:29.814 | ERROR | PID[28220] | Th[Pooling Pending Senior] -

CI[.c.t.c.m.p.s.h.CodeUtilsSenior] > * Send Request Card List - Error when handling Card List load model: 404 Not Found on GET request for "https://sam-api.senior.com.br/sdk/v1/device/access/225/card": [no body]

Mensagem de erro vinda do conexX quando não possui ação para realizar no Middleware

- 2025-05-05 10:27:33.225 | ERROR | PID[28220] | Th[http-nio-8082-exec-3] -

CI[.c.t.c.m.p.c.w.r.ApiWebhookResource] > * Handler Message Webhook Received - The Webhook was unable to identify a pending execution or the message to be handled: WebhookGenericModel [error=false, datePost=2025-05-05T10:24:13, processStart=2025-05-05T10:23:49, processEnd=2025-05-05T10:24:13, dependencyType=BIOMETRY_REGISTRATION_ERROR, data={registry=999999999999, date=291220560009, error=END_LIST}, deviceIpAddress=192.168.91.104]

7. Firmware

Tabela das Versões dos Firmwares disponíveis para utilização do ConexX_Driver , poderá ser utilizados a partir das versões listadas abaixo:

| Ponto | |
|---------------------------|---|
| CEP 3K LCD | CEP_3k_LCD_01_12_01-0075 |
| CEP 3k TFT | CEP_3k_TFT_01_11_01-0075 CEP_3k_TFT_31_11_01-0018 |
| CEP LT LCD | CEP_LT_LCD_01_14_01-0075 CEP_LT_LCD_31_14_01-0018 |
| CEP LT LCD GPRS | CEP_LT_LCD_GPRS_01_14_01-0075 |
| CEP 4k LCD | CEP_4k_LCD_41_14_01-0018 |
| CEP 4k TFT | CEP_4k_TFT_41_11_01-0018 |
| CEP LT LCD 671 | CEP_LT_LCD_671_32_14_01-0018 |
| CEP MICRO LCD | CEP_MICRO_LCD_01_15_01-0075 |
| CEP MICRO LCD GPRS | CEP_MICRO_LCD_GPRS_01_15_01-0075 |
| REP 3k TFT | REP_3k_TFT_01_01_05A0_D0075 REP_3k_TFT_31_01_01-0018 |
| REP LT LCD | REP_LT_LCD_01_04_01-0075 REP_LT_LCD_31_04_01-0018 |
| REP LT LCD GPRS | REP_LT_LCD_GPRS_01_04_01-0075 |
| REP MICRO LCD | REP_MICRO_LCD_01_05_01-0075 |
| REP MICRO LCD GPRS | REP_MICRO_LCD_GPRS_01_05_01-0075 |
| REP 4k LCD | REP_4k_LCD_41_04_01-0018 |
| REP 4k TFT | REP_4k_TFT_41_01_01-0018 |

Tabela Versão Firmwares

obs: Rep 2000 utilizar sempre a ultima versão do firmware

As mensagens negativas do equipamento de acesso, serão exibidas pelo aplicativo ou pela plataforma da senior dependendo da Versão do Firmware de Acesso. Abaixo podemos visualizar a versão mínima dos firmwares para receberem a mensagem configurada na plataforma da senior. versões aquém dessas devem possuir a mensagem negativa configurada no aplicativo

| Acesso | |
|---------------|-----------|
| MDL | MDL114010 |
| MDF | MDF114010 |
| MD Z80 | MD04A428 |

Tabela Versão Firmwares

obs: A Versão do firmwares dos Equipamentos de Acessos não irá impactar a comunicação com o ConexX_Driver

8. Configuração Aplicativo

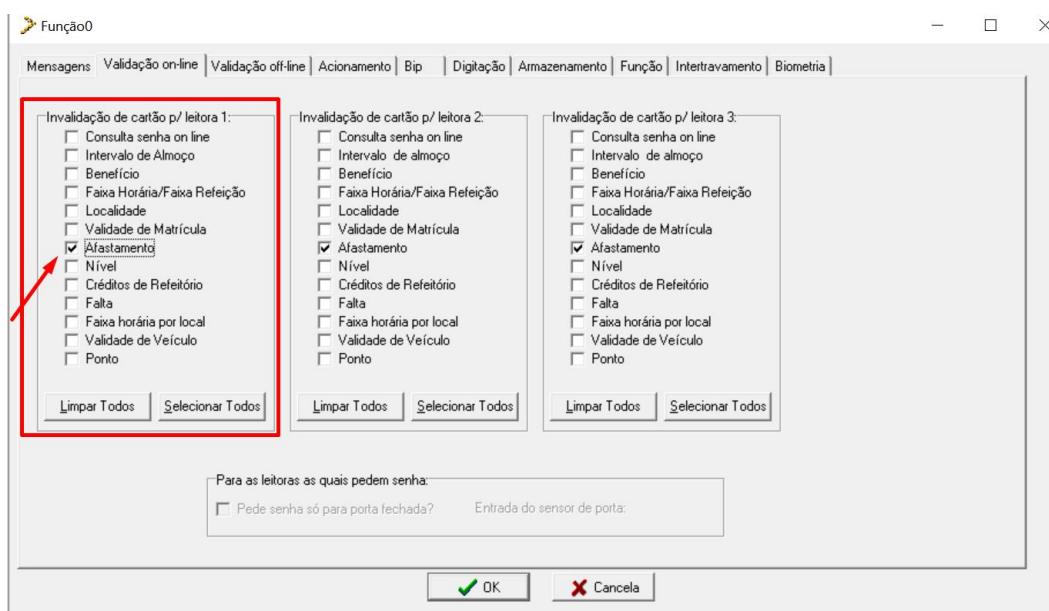
A configuração do Aplicativo é uma etapa importante para o seu funcionamento correto, para utilizarmos com o ConexX_Driver, dependendo a versão do firmware utilizado será necessário uma configuração diferente. Caso esteja utilizando uma versão atual, ou superior dos firmwares demonstrados no tópico [Firmware](#), não será necessário nenhum tipo de validação online



Tela - Validação Aplicativo

Caso esteja utilizando um versão do firmware inferior nas exibidas no tópico [Firmware](#) será necessário realizar uma configuração específica no aplicativo. o equipamento realizará a validação negativa conforme estiver configurado no aplicativo

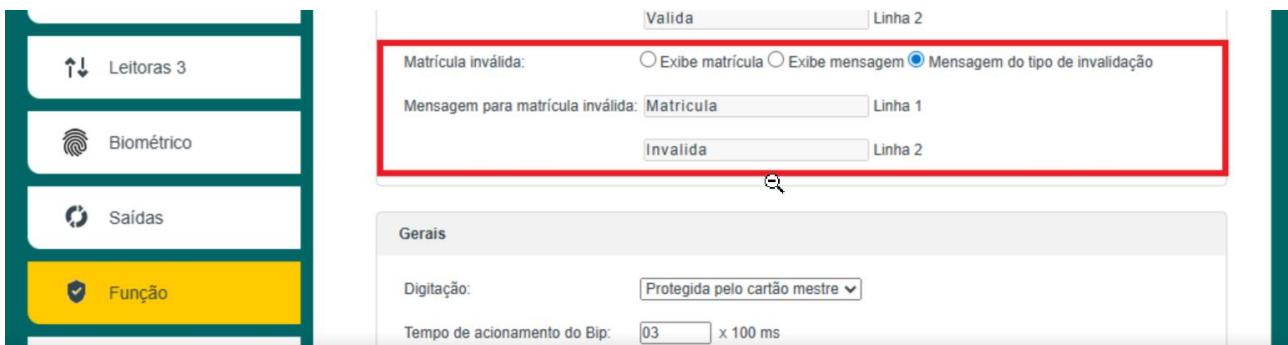
Abaixo temos o exemplo de um validação por afastamento



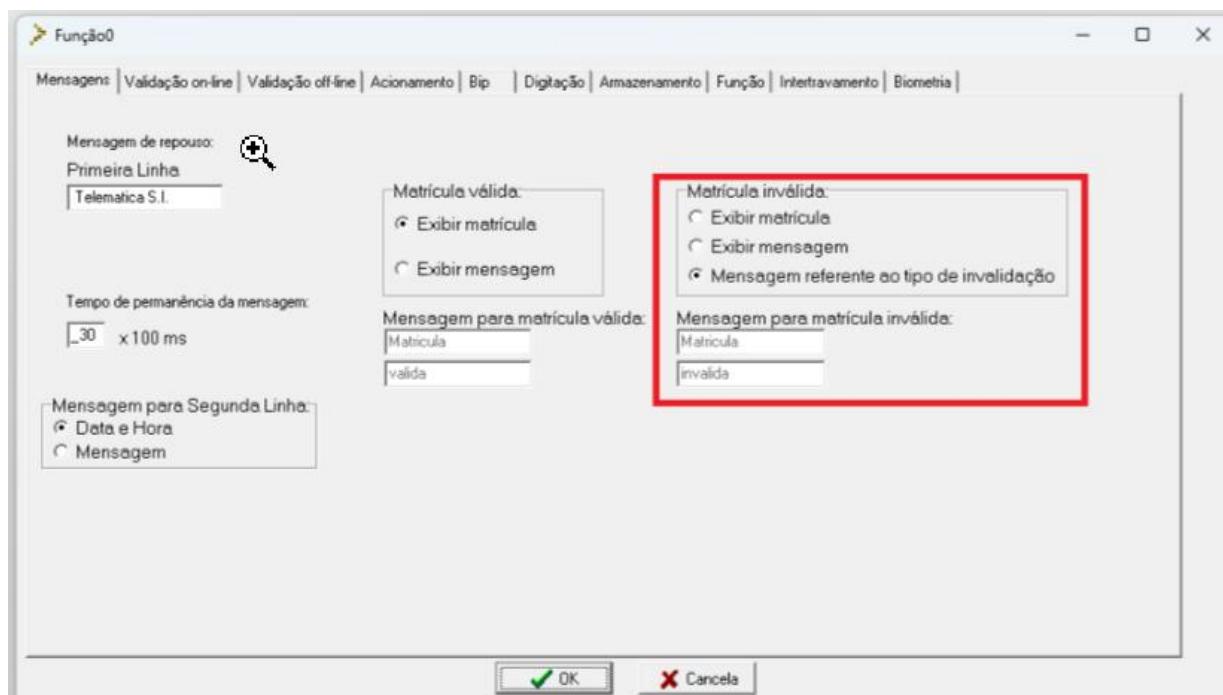
Tela - Validação Aplicativo

Entretanto em ambos os casos será necessário marcar uma opção nas configuração da Função do aplicativo, para ser exibido corretamente a mensagem, a opção "Mensagem do tipo de invalidação" precisa estar selecionado

Configuração no Aplicativo WEB

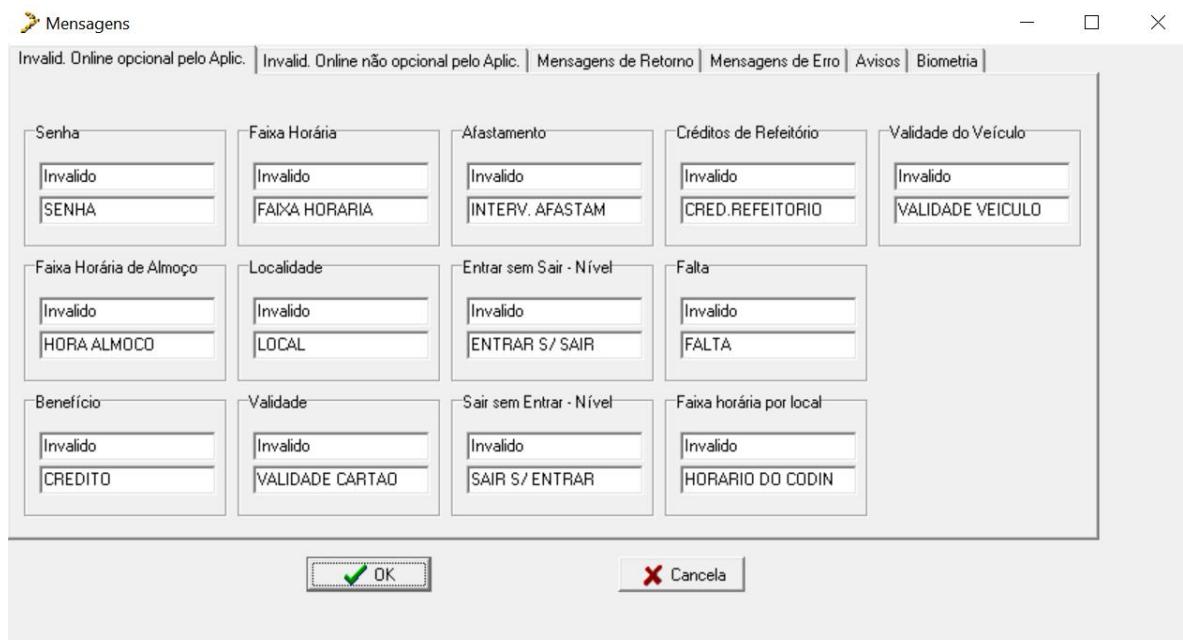


Configuração do Aplicativo pelo Software



8.1. Mensagem

Caso esteja utilizando um versão do firmware inferior nas exibidas no tópico [Firmware](#) a mensagem negativa será configurada no aplicativo, podemos efetuar sua configuração no menu Mensagens



Tela - Configuração Mensagens

9. Diagrama dos Servidores

Os serviços ConexX_Driver, ConexX_Middleware são híbridos podendo estabelecer comunicações contanto que estejam configurados corretamente

O ConexX_Middleware é o serviço responsável por trocar informações através do protocolo de comunicação com os parceiros

O ConexX_Driver é o serviço responsável por trocar informações através do protocolo de comunicação com os Codins

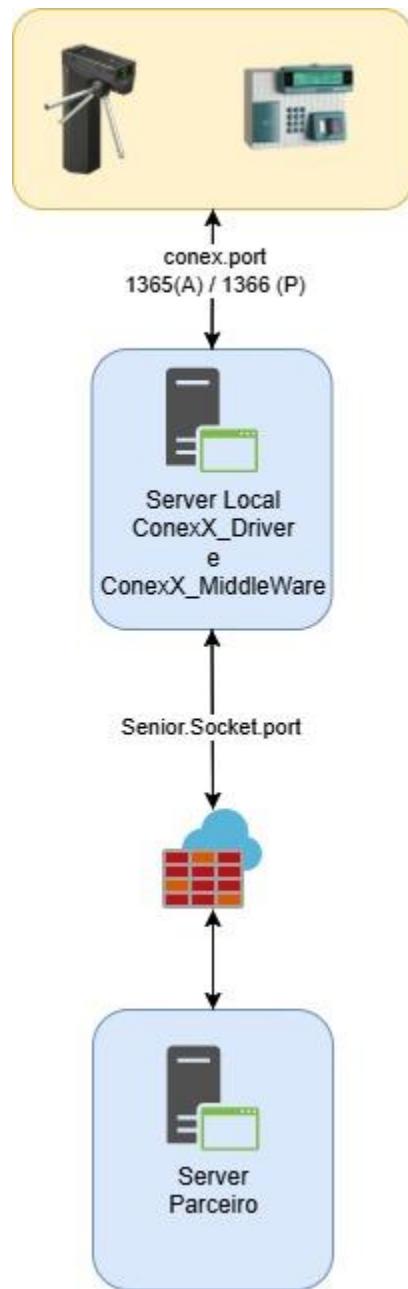
Podemos realizar suas conexões de diferentes formas. sendo elas

- Ambos os serviços locais: ConexX_Driver e ConexX_Middleware na Borda
- Ambos os Serviços na cloud: ConexX_Driver e ConexX_Middleware na Nuvem
- De forma Híbrida: ConexX_Driver na Borda e ConexX_Middleware na Nuvem

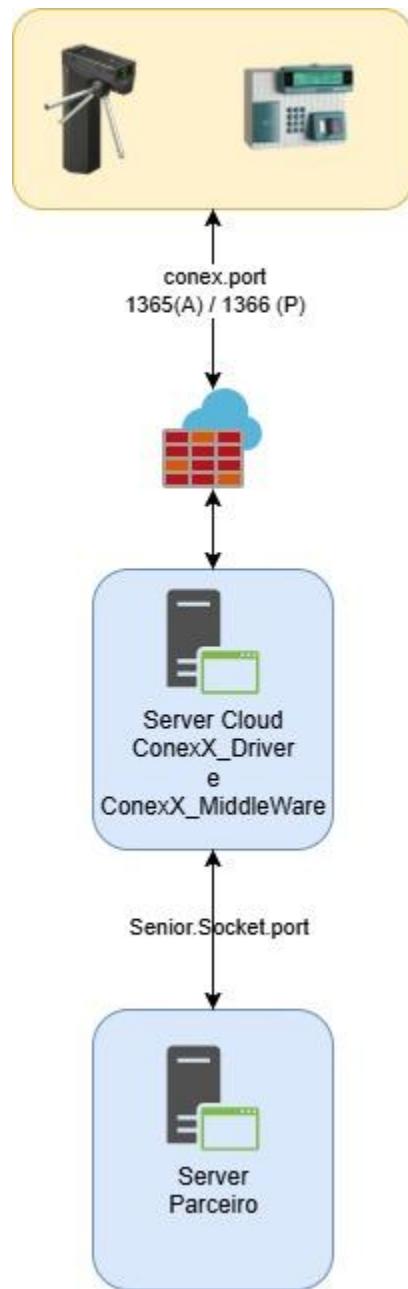
obs: as portas do Conex_port podem ser alteradas

obs²: Parceiros está simbolizando as integrações que o conexX realiza (Sara, SeniorX, SeniorXT)

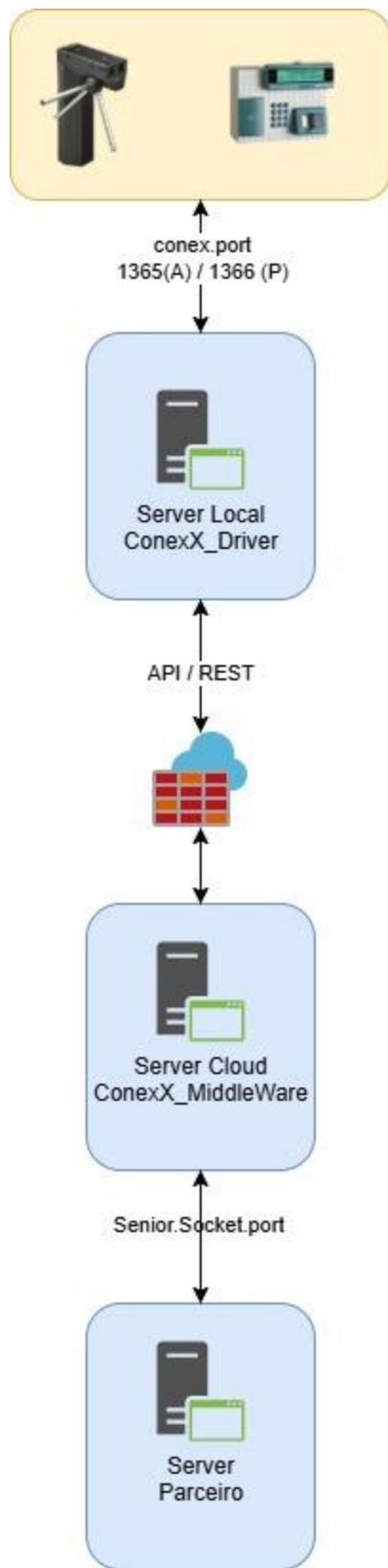
ConexX_Driver e ConexX_Middleware na Borda



ConexX_Driver e ConexX_Middleware na Nuvem



ConexX_Driver na Borda e ConeX_Middleware na Nuvem



10. Compatibilidade

Os serviços de comunicação do ConexX_Driver e ConexX_Middleware possuem versões nas quais são compatíveis e versões nas quais são incompatíveis, a seguir teremos uma tabela com a compatibilidade das versões

| Compatíveis | |
|---------------|-------------------|
| ConexX_Driver | ConexX_Middleware |
| Versão 2.3.0 | Versão 2.15.0 |

Tabela - Compatibilidade