



Persona 3

Manual do usuário

digitro.

Manual do desenvolvedor Persona 3

Release: 1.59

© 2024

por

DÍGITRO TECNOLOGIA S.A

Seção de Documentação - Departamento Técnico

Rua Profª Sofia Quint de Souza, 167 - Capoeiras

CEP 88085-040 - Florianópolis - SC

www.digitro.com

Todos os direitos são reservados. É vedada, no todo ou em parte, a sua reprodução por toda a sorte de formas e meios conhecidos. Para tal, é imperativa a autorização, por escrito, da DÍGITRO TECNOLOGIA S.A. Seu conteúdo tem caráter técnico-informativo e os editores se reservam ao direito de revisar as versões, de modo a aproveitar a totalidade ou parte deste trabalho, sem necessidade de qualquer forma de aviso prévio.

Florianópolis, abril de 2024.

SUMÁRIO

Sobre este documento.....	10
Bem-vindo.....	10
Observações Importantes.....	10
Convenções	12
Introdução	13
Acesso ao Sistema	14
Alarmes.....	16
Ocupação de HD.....	18
Alterar dados do usuário e Sair.....	19
Cadastros	22
Calendários	23
Contatos.....	29
Robôs.....	39
Rotas	43
Serviços	57
Submídias.....	65
Pessoas	83
Interface de Edição de Serviços	85
Barra de Botões.....	87
Árvore de Diretórios e Arquivos	93
Editor.....	94
Comparar revisões	97

Debug.....	98
Console.....	107
Supervisão	108
Supervisão de Chamadas.....	109
Supervisão de Estatísticas.....	115
Gráfico de histórico de ocupação.....	116
Barra de ocupação atual.....	116
Pico de ocupação:.....	116
Estatísticas de chamadas.....	117
Gráfico de estatísticas.....	119
Supervisão de Grupos.....	120
Supervisão de Rotas.....	121
Supervisão de Serviços.....	122
Configurações	124
Configurações Gerais.....	124
Relatórios	125
Log de Operações.....	126
Outros	128
Licenças.....	128
Sobre.....	129
Testar chamada de <i>chat</i>	130
Linguagem de Programação de URA.....	132
Linguagem de LUA.....	133
Programação de menus.....	134
Biblioteca de domínio.....	134

Menu.....	134
API	135
Métodos diretos	135
Message	136
hsm.getFidelization	141
hsm.getMessages	142
hsm.sendMessage	143
Play.....	145
Prompt.....	148
Typing.....	150
File.receive	151
File.send	151
Xfer	152
Drop.....	153
Dialer.getInfo.....	154
Dialer.setContactState.....	155
Sleep	156
Trace	157
Sms.....	157
Answer	158
Auxiliar de navegação	158
Menu.....	158
Utilidades gerais.....	160
Métodos especiais.....	160
Callback.....	160

Report.....	162
Checkpoint.....	163
sendAlarm.....	164
nlp.dialogflow.....	164
nlp.watson.....	165
record.start.....	166
record.stop.....	166
contact.insert.....	167
contact.get.....	168
contact.remove.....	168
Variáveis de ambiente.....	169
Funções úteis.....	170
Calendar.....	170
Store.....	171
Recover.....	171
Format.....	172
Funções do Lua exportadas para a LPU.....	172
Utils.....	173
SQL.....	177
Socket.....	181
valida_cpf.....	182
valida_cnpj.....	182
readfile.....	182
file2string.....	183
damm_generate.....	183

dammm_validate.....	183
luhn_generate.....	184
luhn_validate.....	184
utf2iso.....	184
iso2utf.....	185
string2table.....	185
filter.....	185
sttparse.....	186
Exemplo de <i>script</i> de URA.....	186
Anexo A – Emissão de Relatórios Persona 3	204
Aba Parâmetros.....	205
Agrupar.....	205
Atributo.....	205
Classificação.....	206
Classificado Como.....	206
Colunas de resultado de envio de mensagens HSM – Exportação.....	207
Contabilizar apenas último ponto de verificação.....	208
Detalhes da exportação CSV.....	208
Detalhes da exportação XLS.....	208
Exibição do relatório.....	209
Exibir Somente.....	209
Gerar totalizador de chamadas por ponto de verificação.....	210
Grupos.....	210
Intervalo.....	210
Informações adicionais.....	211

Origem.....	212
Resultado do Atendimento no Grupo.....	212
Robôs.....	212
Separador de Campos para Arquivo CSV	213
Serviço de URA.....	213
Submídia.....	214
Submídia WhatsApp.....	214
Telefone.....	214
Template HSM.....	215
Tipo de Mídia	215
Tipo de Chamada de Chat.....	216
Tipo de Submídia	216
Módulo Persona 3	217
Modelo 7000 - Integração Mídias Sociais - Estatísticas de Mensagens.....	218
Modelo 30601 – URA – Chamadas, Totalizador de Acessos por Ponto de Verificação.....	222
Modelo 30602 – URA – Chamadas, Exportação da Listagem dos Pontos de Verificação.....	225
Modelo 30603 – URA – Chamadas, Totalizador de Chamadas no Grupo por Serviço	226
Modelo 30604 – URA – Chamadas, Informações da Chamada no Serviço, Ponto de Verificação	229
Modelo 30605 – URA – Chamadas, Gráfico das Chamadas por Hora.....	232
Modelo 30606 – URA – Chamadas, Gráfico das Chamadas por Dia	235
Modelo 30607 – URA – Agentes, Relatório de Pesquisa de Satisfação.....	237

Modelo 30608 – URA – Chamadas, Totalizadores por Classificação da Chamada	240
Modelo 30609 – URA – Chamadas, Totalizadas por atributos valores do serviço	242
Modelo 30610 – URA – Lista Chamadas com Resultado	244
Glossário.....	247

1

SOBRE ESTE DOCUMENTO

BEM-VINDO

A interface **Persona 3** permite a realização de todas as configurações de URA, simulação de chamadas, cadastro de rotas e calendários, geração de relatórios, supervisão de chamadas, monitoração de ocupação e estatísticas da plataforma e estatísticas dos serviços, rotas e grupos.

OBSERVAÇÕES IMPORTANTES

1. Para as interfaces gráficas dos programas detalhados desse manual, sugere-se a utilização de uma resolução gráfica de vídeo de 1280X800 ou superior, nos formatos 16:9 ou 16:10.
2. Ficará a critério da Dígito disponibilizar, através de proposta de fornecimento ou contrato de suporte específico, facilidades adicionais que venham a ser criadas.

Sobre este Documento

CAPÍTULO 1

3. Serviços solicitados pelo cliente, que impliquem em alterações em características específicas, funções adicionais ou outros itens não especificados, serão considerados como adicionais, e serão efetuados conforme cronograma de execução e alocação de recursos, elaborados pela Dígitro e aprovados pelo cliente, através de proposta comercial.
4. Toda funcionalidade identificada com a palavra **Opcional**, não faz parte da solução. Seu fornecimento depende de proposta específica.
5. A Dígitro, como qualquer empresa desenvolvedora, não pode garantir que *softwares* não contenham erros ou que o cliente seja capaz de operá-los sem problemas ou interrupções, por isso, não assume eventuais prejuízos financeiros decorrentes dessas falhas ou de problemas de responsabilidade de terceiros.
6. Devido ao desenvolvimento contínuo de técnicas de invasão e ataques à rede, não é possível garantir que o equipamento (*hardware* e *software*) esteja livre da vulnerabilidade da invasão/ação externa.
7. Após o aceite ou a entrada em operação do sistema, se ocorrer erros ou falhas, estes somente serão avaliados e/ou corrigidas mediante contrato de suporte ou autorização para pagamento de suporte avulso, conforme a tabela de preços vigente na data da solicitação.
8. A Dígitro não atualizará este produto em função de novas versões, sendo necessário, para isso, negociação comercial.
9. A Dígitro não se responsabiliza por perdas de informações, devido a não observação, por parte do cliente, de procedimentos de *backup*, orientando para que, regularmente, armazene os dados também em mídia eletrônica (CD, DVD, etc.), de forma a possuir contingência externa.
10. A Dígitro mantém um processo de ciclo de vida de seus produtos devido a inovações tecnológicas, necessidade de mercado ou outro motivo. Para mais informações, acesse o ambiente exclusivo para clientes no site www.digitro.com.br.
11. Chamadas transferidas entre URAs irão consumir, por um curto período de tempo, duas licenças. Este tempo pode aumentar, caso a primeira URA possua ações dentro do menu finish.

CONVENÇÕES

JANELAS E MENUS	Os nomes das janelas e dos menus, quando aparecerem no meio do texto, serão grafados em CAIXA ALTA.
Campos	As iniciais maiúsculas identificam nome de Campos no meio do texto.
<i>Palavras de Origem Estrangeira</i>	As palavras de origem estrangeira estarão grafadas em <i>itálico</i> .
<u>Palavras de destaque</u>	As palavras que necessitarem de destaque em um determinado contexto, serão <u>sublinhadas</u> .

2

INTRODUÇÃO

O **Persona 3** fornece duas formas diferentes de supervisão: a supervisão de chamadas ativas e a supervisão das estatísticas das chamadas recebidas durante o dia. Essas opções podem ser acessadas através do menu Supervisão.

Para acessar a supervisão, o usuário deve possuir privilégios de supervisão.

Por padrão, os perfis **Administrador Persona 3** e **Supervisor Persona 3** já possuem este privilégio.

ACESSO AO SISTEMA

O sistema é acessado através de um navegador Internet (*browser*).

PROCEDIMENTO

Para acessar o sistema

1. Abra o Mozilla Firefox e, na barra de endereços, digite o endereço do servidor de acesso acrescentado no final “*persona*”.
2. É apresentada a janela de login, conforme mostra a figura a seguir.

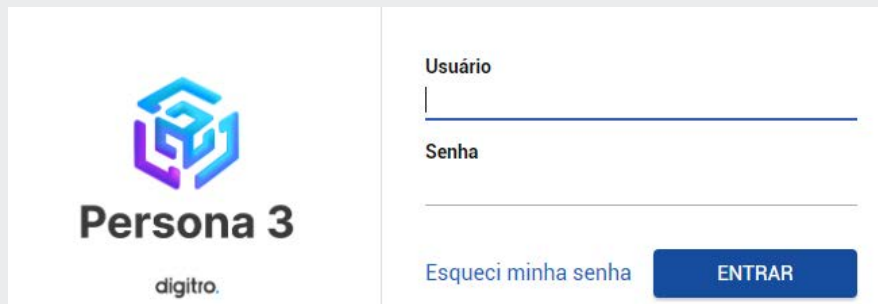


Figura 1. Login

3. Digite o usuário e a senha de acesso ao sistema.
4. Em seguida é apresentada a janela principal do **Persona 3**, conforme mostra a figura a seguir.



Figura 2. Tela Principal da Interface

O usuário pode redefinir sua senha de acesso à interface selecionando a opção **Esqueci minha senha** (Figura 1) na página de login do **Persona 3**.

ATENÇÃO

*A opção **Esqueci minha senha** não estará disponível quando o Cadastro de Pessoas estiver integrado ao LDAP.*

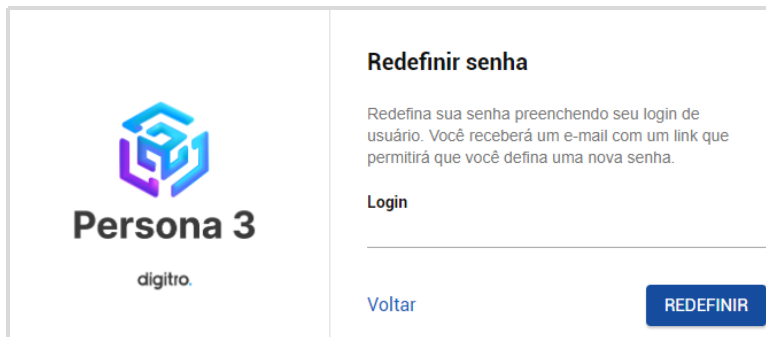


Figura 3. Redefinir senha

Informe o login do usuário e clique no botão **Redefinir**. Será exibida uma mensagem confirmando o envio de um e-mail para o endereço cadastrado com o link para a redefinição da senha.


NOTA

*A interface apresenta, no canto superior direito, o horário atual do servidor onde rodam os processos do **Persona3**.*

ALARMES

O alarme indica que um problema ocorreu (ou está ocorrendo) em determinado site, para que o usuário logado (operador ou administrador) possa tomar as devidas providências.



O botão Alarme  está localizado no canto superior direito da interface e muda de cor conforme a prioridade do alarme, como apresentado na figura a seguir:

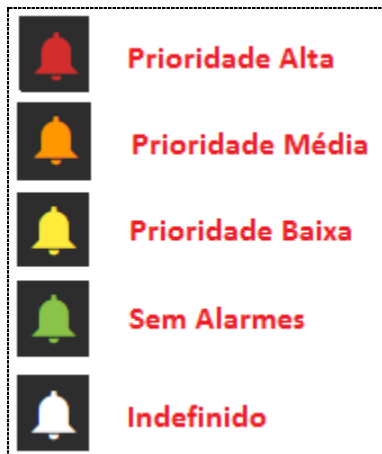
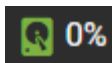


Figura 4. Alarmes

Ao clicar no ícone Alarmes, é aberta uma nova janela que apresenta a aplicação de **Supervisão de Alarmes** filtrada pelos alarmes do **Persona 3**.
Para mais informações, consulte o manual **Supervisão de Alarmes**.

OCUPAÇÃO DE HD



O ícone ocupação de HD, localizado à direita do ícone Alarmes, na parte superior da interface, permite visualizar o percentual de ocupação do HD.

As cores dos ícones indicam o nível de ocupação:

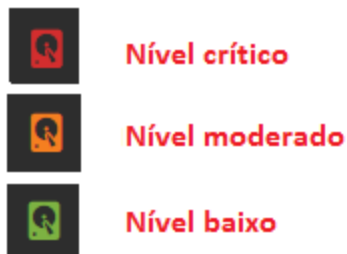


Figura 5. Níveis de ocupação do HD

A cor verde indica um nível baixo de ocupação, entre 0 e 69%; a cor laranja indica um nível moderado de ocupação, entre 70 a 89%; e a cor vermelha indica um nível crítico de ocupação, a partir de 90%.

Ao posicionar o cursor do mouse sobre o ícone são apresentadas informações mais detalhadas da ocupação, tais como o nível, o espaço total ocupado pelos serviços salvos no servidor e o total de espaço máximo disponível configurado para ser utilizado pelos serviços.

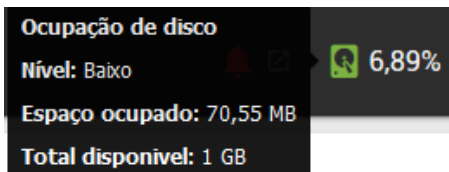


Figura 6. Detalhes da ocupação de disco

Ao atingir o nível crítico, o sistema poderá impossibilitar a criação, cópia, importação e edição de serviços. Nesse caso, será exibida a mensagem de espaço insuficiente em disco.

NOTA

O valor padrão de espaço em disco é de 1GB. Somente o SAC da Dígitro poderá visualizar e alterar esta configuração.

ALTERAR DADOS DO USUÁRIO E SAIR

As opções **Alterar dados do usuário** e **Sair** são exibidas ao clicar no botão usuário



, localizado no canto superior direito da interface, e permitem, respectivamente, alterar os dados do usuário e encerrar a sessão ativa do usuário.

PROCEDIMENTO

Alterar a senha do usuário

1. Clique em menu *Alterar dados do usuário* > **Senha**. Será apresentada a Figura 7.

Alterar dados do usuário

Alteração de dados

▶ **Senha** E-mail

Ao definir uma nova senha, certifique-se que ela deve conter o mínimo de 8 caracteres e seguir pelo menos 3 regras abaixo:

- Um caractere do tipo maiúsculo;
- Um caractere do tipo minúsculo;
- Um caractere do tipo especial;
- Um caractere do tipo numérico.

*Senha atual:

*Nova senha:

*Confirme a nova senha:

OK Cancelar

Figura 7. Alteração de Senha


2. Informe a *Senha Atual*, a *Nova Senha* e confirme a nova senha.
3. Clique em **OK** para confirmar a alteração.

PROCEDIMENTO

Alterar e-mail

1. Clique em menu *Alterar dados do usuário* > **E-mail**. Será apresentada a Figura 8.

Alterar dados do usuário



Alteração de dados

[Senha](#) ▶ [E-mail](#)

Ao alterar o e-mail principal, deve-se confirmar a mudança mediante senha atual do usuário

*Senha atual:

*E-mail atual:

*Novo e-mail:

*Confirme o novo e-mail:

Figura 8. Alteração de E-mail

2. Informe a Senha Atual, o Novo e-mail e confirme-o.
3. Clique em **OK** para confirmar a alteração.

3

CADASTROS

O menu **Cadastro** disponibiliza ao usuário o acesso aos cadastros de calendários, contatos, robôs, rotas, serviços, submídias e pessoas. Esse menu pode ser acessado por usuários que possuem privilégio de cadastro supervisão ou cadastro edição.

Por padrão, os perfis **Administrador Persona 3** e **Supervisor Persona 3** já possuem esses privilégios.

O usuário que possuir cadastro supervisão somente poderá visualizar as informações cadastradas, não sendo permitido realizar alterações.

O menu CADASTROS apresenta os submenus mostrados na figura a seguir.

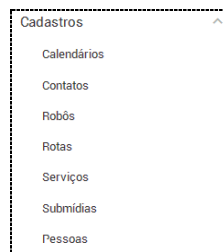


Figura 9. Menu Cadastros

CALENDÁRIOS

O cadastro de calendários, além de exibir a listagem com todos os calendários configurados, permite adicionar, remover, editar e copiar um calendário.

Existem dois tipos de calendários:

- **Ativos:** utilizados para determinar a data e hora de disparo de uma chamada ou notificação de *chat*.
- **Receptivos:** utilizados para informar os horários de atendimento ou início de um determinado serviço.

Podem ser utilizados de duas formas: dentro da *script* de URA ou no Cadastro de rota.

Em uma rota, é possível configurar vários calendários diferentes.

Ao receber uma chamada em uma rota receptiva, o **Persona 3** verifica todos os calendários e, caso algum retorne que o horário é válido, o atendimento é realizado. Caso esteja fora das faixas configuradas nos calendários, a rota não é considerada válida e a próxima, caso exista, será verificada. Em rotas ativas, utilizando mais de um calendário, estes não devem conter horários iguais.

Em calendários receptivos, caso o usuário não queira bloquear o atendimento da chamada em uma rota, mas sim realizar um tratamento específico dentro de um horário pré-determinado, existe a possibilidade de utilizar a primitiva *calendar*, passando como parâmetro o nome do calendário desejado. Caso a chamada tenha sido realizada dentro do horário aceito, será retornado o valor *true*.

É possível configurar um calendário receptivo sem faixa de horário. Neste caso, as datas configuradas serão consideradas como datas sem atendimento (exemplo: feriados).



The screenshot shows a web application window titled 'Cadastro de calendários'. It contains a table with the following data:

Ações	Tipo	Nome	Turnos	Dias	Horas
<input type="checkbox"/>	Ativo	tempo	-	SEG,QUA,SEX,QUI,TER	10:13,10:59
<input type="checkbox"/>	Ativo	diasComFes	-	26/04	08:00
<input type="checkbox"/>	Receptivo	CalendarioNoturno	18:00-23:59	SEG,TER,QUA,QUI,SEX,SAB,DOM	-

Figura 10. Calendários

PROCEDIMENTO

Novo Calendário


1. Para criar um novo calendário, pressione o botão  (Figura 10).
2. É apresentada a figura a seguir.

Figura 11. Novo Calendário

3. No campo **Tipo**, selecione a opção **Ativo** que indica se o calendário será utilizado em rotas ativas (disparo de chamadas ou notificações de chat) ou **Receptivo** para atendimento de chamadas de chat ou voz.
4. No campo **Nome**, preencha o nome que será utilizado como identificador do calendário. Não são permitidos dois calendários com o mesmo nome.

5. No campo **Dias**, selecione qual(is) o(s) dia(s) da semana em que o serviço será utilizado.
6. No campo **Dias especiais**, selecione a opção **Dia**, **Dia/Mês** ou **Dia/Mês/Ano** e insira a data em que o calendário terá efeito sobre o serviço.
7. Clique em **Incluir** para validar.
8. No campo **Horário** (receptivo), preencha a faixa de horário em que ele será válido.
9. Clique em **Incluir** para validar.
10. No campo **Hora** (ativo), preencha a hora em que será iniciado o disparo de chamadas ou notificações de chat.
11. Clique em **Incluir** para validar.
12. Clique no botão **Salvar** para validar o cadastro.

NOTA



Os campos **Nome**, **Dias** da semana ou **Dias Especiais** são obrigatórios. Ao informar um valor no campo **Nome**, a aplicação já verifica a existência de outro registro idêntico.

O campo **Nome** aceita somente caracteres alfanuméricos, '_' e '-', com no mínimo 3 e no máximo 50 caracteres.

No cadastro de um horário, o final de turno deve ser maior que o início.

PROCEDIMENTO

Editar Calendário

1. Para editar um calendário, clique no ícone  e, em seguida, no ícone  do registro que deseja modificar.
2. É apresentada a janela do cadastro selecionado, conforme mostra a figura a seguir.

Editar calendário

Tipo Ativo Receptivo

* Nome

Dias
Em quais dias da semana se usa URA?
 DOM SEG TER QUA QUI SEX SAB

Dias especiais
Em quais dias do ano se usa URA?

Hora
Em qual hora do dia se usa URA?

10:13 ✕ 10:59 ✕ 11:33 ✕ 11:36 ✕
 11:41 ✕ 11:52 ✕ 13:46 ✕ 13:54 ✕



* Campos obrigatórios

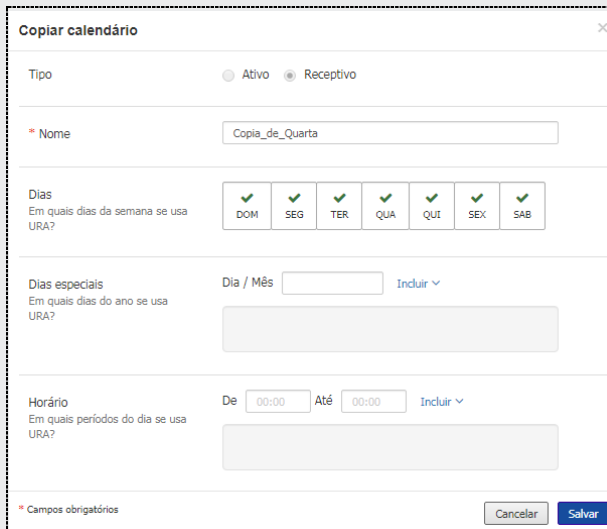
Figura 12. Editar calendário

3. *Edite os campos dessa janela conforme desejar, porém os campos **Tipo** e **Nome** não são alteráveis. Os demais campos são editáveis, seguindo as mesmas regras do cadastro de um novo calendário.*
4. *Clique sobre o botão **Salvar** para validar as alterações.*

PROCEDIMENTO

Copiar Calendário

1. A cópia de um calendário permite a criação de um novo registro baseado no selecionado, adicionando o prefixo, *Copia_de_* no nome do calendário. O campo *Tipo* fica desabilitado e os demais campos são editáveis, seguindo as mesmas regras do cadastro de um novo calendário.
2. Para copiar um calendário, clique no ícone  e, em seguida, no ícone  do registro que deseja copiar.
3. É apresentada a janela do cadastro selecionado, conforme mostra a figura a seguir.



Copiar calendário

Tipo Ativo Receptivo

* Nome

Dias
Em quais dias da semana se usa URA?

<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
DOM	SEG	TER	QUA	QUI	SEX	SAB

Dias especiais
Em quais dias do ano se usa URA?

Dia / Mês Incluir ▾

Horário
Em quais períodos do dia se usa URA?

De Até Incluir ▾

* Campos obrigatórios


Cancelar Salvar

Figura 13. Copiar calendário

4. Clique sobre o botão **Salvar** para realizar a cópia.

PROCEDIMENTO

Remover Calendário

1. Para excluir um ou mais calendários, selecione o(s) registro(s) desejado(s) e pressione o botão  **Remover** (Figura 10).
2. É apresentada a mensagem.

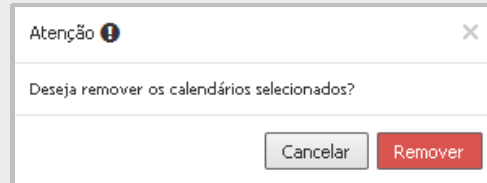


Figura 14. Mensagem remover calendário(s)

3. Clique no botão **Remover** para validar a exclusão.

CONTATOS

Os contatos são utilizados nas chamadas de chat ativas. Há três formas de cadastrar um contato:

1. Através de uma chamada receptiva de *chat*, onde na *script* de URA, através das primitivas do **Persona**, é possível adicionar o contato ao banco de dados;
2. Cadastro manual;
3. Importação de arquivo txt ou csv.

Os contatos adicionados ao banco de dados estarão sempre vinculados a um robô através do seu id.

NOTA

Ao criar um serviço personalizado para chat ativo, é possível utilizar as informações cadastradas no ambiente ao enviar a mensagem ativa.

Para a submídia WhatsApp, existem algumas particularidades que é preciso levar em conta:

- 1. O envio de mensagem para um contato de WhatsApp que nunca tenha entrado em contato previamente é feito via HSM, que deverá ser enviado a partir do fluxo de serviço.*
- 2. Só é possível enviar mensagem de texto, áudio ou imagem para contatos do WhatsApp com sessão aberta, isto é, para contatos que tenham tido uma sessão aberta nas últimas 24 horas.*
- 3. Para enviar uma mensagem HSM, é preciso informar o token Dígitro da submídia e o template no fluxo do serviço.*
- 4. O template, parâmetro, fidelização e demais dados deverão ser tratados no fluxo da URA com dados particulares de cada um dos contatos informados através do campo Complemento;*
- 5. Caso o número para o qual está sendo enviado o HSM tenha uma conversa em andamento através da mesma submídia, o envio deverá ser negado.*
- 6. A exclusão de contatos adicionados à lista deverá ser feita manualmente via interface ou no fluxo da URA/chatbot conforme a necessidade.*

O cadastro de contatos, além de exibir a listagem com todos os contatos cadastrados, permite editar e remover um contato.

Na Lista de contatos são exibidas as colunas **Ações**, **Nome do Contato**, **Submídia**, **Origem**, **Tipo**, **Destino** e **Complemento** do contato.

Cadastro de contatos

Lista de contatos

+ Novo contato Remover Importação

Pesquisar

<input type="checkbox"/>	Ações	Nome do contato	Submídia	Tipo	Destino	Complemento
<input type="checkbox"/>		Teste1	Submídia Teste	📞	221333333	101010192
<input type="checkbox"/>		Cliente5	Submídia Teste	📞	222333333	101010192
<input type="checkbox"/>		...	Submídia Teste	📞	223333333	101010192
<input type="checkbox"/>		Cliente4	Submídia Teste	📞	515489444444	09550404.3030330
<input type="checkbox"/>		Contato Telegram	Teste telegram	📧	55444444444444	
<input type="checkbox"/>		Cliente1	Submídia Teste	📞	55479111111111	0951150404.3030330
<input type="checkbox"/>		C.	Submídia Teste	📞	554894444444	09550404.3030330
<input type="checkbox"/>		Cliente2	Submídia Teste	📞	554899998884	00010110101010
<input type="checkbox"/>		Cliente importante 1	Submídia Teste	📞	554899999999	rua martin pescador

Exibindo 1 até 9 de 9 linhas

Figura 15. Cadastro de Contatos

No cadastro ou edição de um contato são exibidos os campos:

- **Ações:** permite editar o contato selecionado;
- **Nome do contato:** o nome do contato cadastrado.
- **Submídia:** tipo da submídia vinculada ao robô;
- **Origem:** apresenta o identificador, que varia conforme a submídia:
 - **WhatsApp:** número da submídia.
 - **Telegram:** token.
 - **Facebook:** id da página.

- **Instagram:** id da página
- **Tipo:** apresenta o ícone da submídia associada ao robô.
- **Destino:** número ou id do cliente, que varia conforme a submídia:
 - **WhatsApp:** número do cliente.
 - **Telegram:** id de identificação (essa informação pode ser obtida através de uma URA receptiva).
 - **Facebook:** id do usuário do Facebook (essa informação pode ser obtida através de uma URA receptiva).
 - **Instagram:** id do usuário do Instagram (essa informação pode ser obtida através de uma URA receptiva)

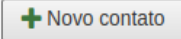
NOTAS

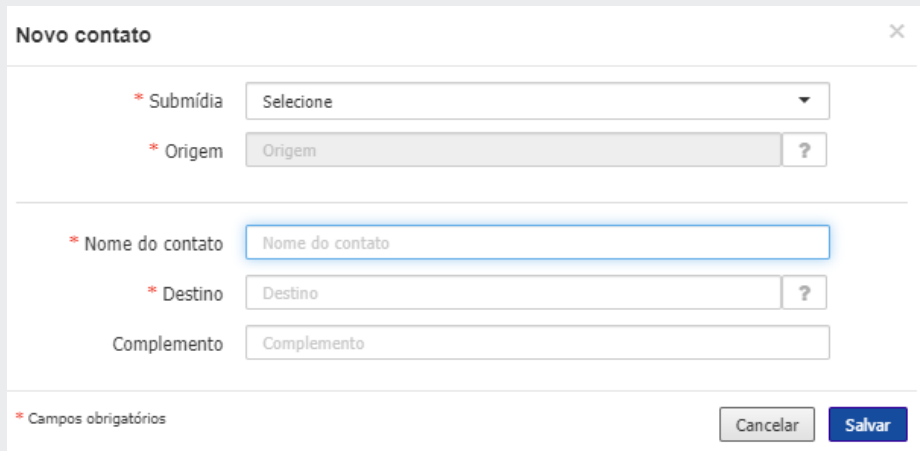
1. *O cadastro manual na interface de importação se aplica somente para as submídias WhatsApp e Telegram.*
2. *As submídias do Instagram e Facebook mantêm o funcionamento do cadastro de contatos somente via URA receptiva utilizando as primitivas do Persona.*

- **Complemento:** informações complementares adicionadas ao cadastro do contato.

PROCEDIMENTO

Cadastrar um Contato manualmente

1. Para cadastrar um contato, acesse o menu CADASTRO > CONTATOS, será aberta a tela CADASTRO DE CONTATOS.
2. Clique no botão .
3. Será aberta a janela Novo Contato, como mostra a figura a seguir.



A janela "Novo contato" apresenta os seguintes campos:

- * Submídia:** Menu suspenso com o texto "Selecione".
- * Origem:** Campo de texto com o texto "Origem" e um ícone de interrogação.
- * Nome do contato:** Campo de texto com o texto "Nome do contato".
- * Destino:** Campo de texto com o texto "Destino" e um ícone de interrogação.
- Complemento:** Campo de texto com o texto "Complemento".

Na base da janela, há o texto "* Campos obrigatórios" e dois botões: "Cancelar" e "Salvar".

Figura 16. Novo contato

4. No campo **Submídia**, selecione o tipo da submídia cadastrada.
5. O campo **Origem** será preenchido automaticamente após selecionar a Submídia.
6. No campo **Nome do contato**, insira o nome do contato que deseja cadastrar.
7. No campo **Destino**, insira o número ou id do cliente, conforme a Submídia.

NOTAS

Para a submídia Telegram, é necessário ter trocado pelo menos uma vez mensagem com o robô do Telegram.

8. No campo **Complemento**, adicione qualquer informação adicional que desejar armazenar.
9. Uma vez preenchidos todos os dados, clique no botão **Salvar** para armazenar o contato, ou em **Cancelar** para ignorar a operação.

PROCEDIMENTO

Importar um arquivo .txt ou .csv

1. Para cadastrar uma lista de contatos em arquivo .txt ou .csv, acesse o menu CADASTRO > CONTATOS, será aberta a tela CADASTRO DE CONTATOS.



2. Clique no botão
3. Será aberta a tela que mostra a última importação realizada.

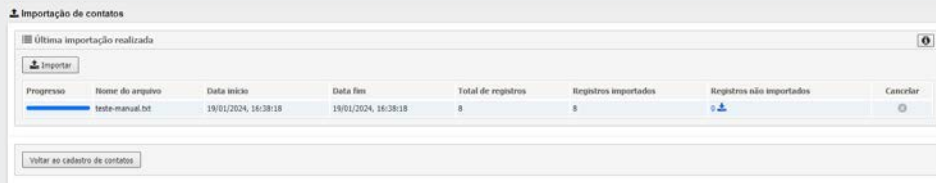



Figura 17. Importação de contatos

4. Clique no botão .
5. Será aberta a janela *Importar*, como mostra a Figura 18.

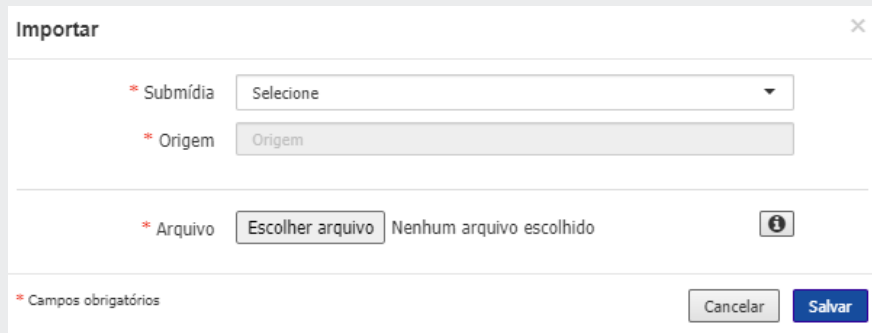


Figura 18. Importar lista de contatos

6. No campo **Submídia**, selecione a submídia na qual serão cadastrados os contatos.
7. O campo **Origem** será preenchido automaticamente após selecionar a Submídia.
8. Clique no botão **Escolher arquivo** e escolha o arquivo .csv ou .txt e aguarde o pré-carregamento da sua lista de contatos.
9. Será apresentada um preview das 10 primeiras linhas da sua lista como mostra a figura seguinte:

Importar
✕

* Submídia ✔

* Origem

* Arquivo teste-manual.txt i

* Separador

Preview (10 primeiras linhas da tabela)

Nome	Telefone	CPF	Endereço	Fidelizado
Cliente1	55479111111	09511150404.3030330	rua das flores	sim
Cliente2	554899998884	00010110101010	rua celeste	sim
Teste1	221333333	101010192	RUA DO ENGENHO	NAO
Cliente4	515489444444	09550404.3030330	avenida kobrasol	sim
CLiente5	222333333	101010192	RUA MARIA DO SOCORRO	NAO
Cliente importante 1	554899999999	rua martin pescador	talvez	
C.	55489444444	09550404.3030330	TRAVESSA campos	sim
~.	223333333	101010192	RUA rosa e carvo	NAO

Excluir contatos já cadastrados para esta submídia.

Primeira linha é cabeçalho.

* Nome do contato

* Destino do contato

Complemento do contato

* Campos obrigatórios

Figura 19. Preview da lista de contatos



10. Ao marcar a caixa de seleção **“Excluir contatos cadastrados desta submídia”**, serão apagados todos os contatos cadastrados daquela submídia antes de inserir os novos valores.
11. A caixa de seleção **“Primeira linha é cabeçalho”** permite identificar os campos da tabela.
12. Caso a primeira linha não seja um cabeçalho, desmarque esta opção e insira o número da linha correspondente ao cabeçalho no campo **Linha do cabeçalho da tabela**.
13. As opções de seleção serão atualizadas para os campos correspondentes.
14. Selecione a coluna correspondente ao nome do contato, a coluna do destino do contato, que pode ser um número de WhatsApp ou um id de conta do Telegram, e o complemento do contato, isto é, uma informação adicional que pode ser utilizada na URA.
15. Uma vez preenchidos todos os dados, clique no botão **Salvar** para armazenar o contato, ou em **Cancelar** para ignorar a operação.

NOTAS

Só é possível efetuar uma importação por vez.

PROCEDIMENTO

Editar Contato

1. Para editar um contato, clique no ícone  e, em seguida, no ícone  do registro que deseja modificar.
2. É apresentada a janela do contato selecionado, conforme mostra a figura a seguir.

Editar contato

* Submídia

* Origem

* Nome do contato

* Destino

Complemento

* Campos obrigatórios


Cancelar Salvar

Figura 20. Editar contato

3. Edite os campos conforme desejar.
4. Clique no botão **Salvar** para validar o cadastro.

PROCEDIMENTO

Remover Contato

1. Para remover um contato, clique na caixa de seleção do contato que deseja excluir e, em seguida, no botão  **Remover** na parte superior da lista.
2. Será apresentada uma mensagem de confirmação:

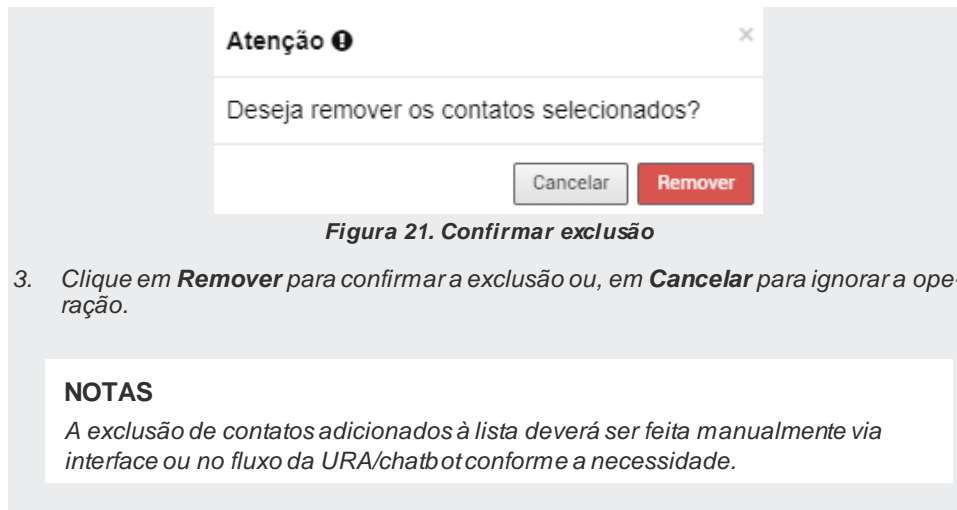


Figura 21. Confirmar exclusão

3. Clique em **Remover** para confirmar a exclusão ou, em **Cancelar** para ignorar a operação.

NOTAS

A exclusão de contatos adicionados à lista deverá ser feita manualmente via interface ou no fluxo da URA/chatbot conforme a necessidade.

ROBÔS

Os Robôs são utilizados para identificar o ponto de entrada das chamadas de *chat* em uma rota.

As informações cadastradas nesta tela serão utilizadas no cadastro de Rotas e Submídias.

O cadastro de robôs, além de exibir a listagem com todos os robôs configurados, permite adicionar, remover e editar um robô.

Na Lista de robôs são exibidas as colunas **Nome** e **Descrição** do robô.




Ações	Nome :	Descrição :	Contatos ativos :	Contatos inativos :
	tempo	123	1	0
	RoboAtivo	aaaa	2	0
	roboAndré	Meu(Nome)	0	0
	testeForanario	testeForanario	0	0
	transfeste	infedeadad	0	0
	eca	eca	1	0

Figura 22. Robôs

Nesta janela também são apresentados os robôs cadastrados no sistema.

No campo **Pesquisar** é possível pesquisar uma palavra ou número e o sistema retornará a busca com os resultados encontrados.

O botão  permite ao usuário selecionar as colunas que serão apresentadas na janela **Robôs**, que são: Ações, Nome, Descrição, Contatos Ativos e Contatos Inativos.

PROCEDIMENTO

Novo Robô

1. Para criar um novo robô, pressione o botão  (Figura 22).
2. É apresentada a figura a seguir.

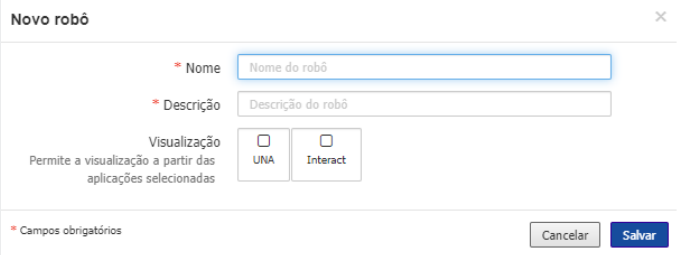


Figura 23. Novo robô

3. No campo **Nome** preencha o nome do robô que será utilizado. Não é permitido dois robôs com o mesmo nome.
4. No campo **Descrição**, informe uma breve descrição do robô a fim de facilitar sua identificação. Este campo aceita caracteres alfanuméricos exceto <, >, e **com quantidade de 1 a 150 caracteres**.
5. Na opção **Visualização**, marque a aplicação ou as aplicações em que deseja visualizar o robô.
6. Clique no botão **Salvar** para validar o cadastro.

PROCEDIMENTO

Editar Robô





1. Para editar um robô clique no ícone  e, em seguida, no ícone  do registro que deseja modificar.
2. É apresentada a janela do robô selecionado, conforme mostra a figura a seguir.

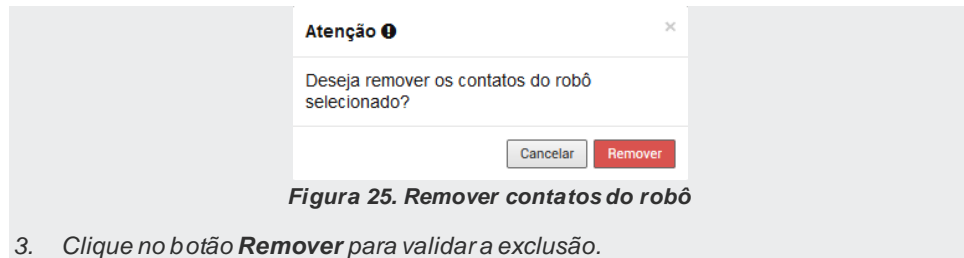
Figura 24. Editar robô

3. Edite o campo **Descrição** dessa janela conforme desejar, exceto o campo **Nome**, que não permite edição.
4. Na opção **Visualização**, marque a aplicação ou as aplicações em que deseja visualizar o robô. Ou desmarque a aplicação na qual deseja desativar a visualização do robô.
5. Clique no botão **Salvar** para validar o cadastro.

PROCEDIMENTO

Remover contatos Robô

1. Para remover os contatos de um robô clique no ícone  e, em seguida, no ícone  do registro que deseja remover.
2. É apresentada a janela do robô selecionado, conforme mostra a figura a seguir.



ROTAS

São as configurações para realizar o atendimento de chamada para determinado Serviço de URA/Chat.

O cadastro de rotas permite adicionar, remover, editar, copiar, bloquear/desbloquear e iniciar/pausar uma rota.

As rotas podem ser de **Chat (Ativa** ou **Receptiva)** ou **Voz**.

As rotas de *chat* receptivas são utilizadas para o atendimento de chamadas de *chat*, já as rotas ativas são utilizadas para geração de chamadas de *chat*. Uma rota de *chat* ativa pode ser do modo **Chamada** ou **Notificação**.

Uma rota pode possuir cinco estados:

- **Ativa:** rota válida e liberada para receber chamadas.
- **Inativa:** rota criada em um grupo inválido do sistema (somente voz), por exemplo, grupo não existente.
- **Bloqueada:** rota bloqueada para o recebimento de chamadas.
- **Executando:** rota ativa executando, isto é, realizando chamadas.
- **Pausada:** rota ativa em pausa para o disparo de chamadas.

Ao clicar no submenu ROTAS, é apresentada a figura a seguir.

Cadastro de rotas

Lista de rotas

+ Nova rota Remover


Pesquisar

Ações	Mídia	Tipo	Modo	Nome	Serviço	Revisão	Arquivo	Grupo/Robô	Destino	Origem	Tráfego	Calendário	Sleep	Estado
		Receptiva	-	RotaVoz	ServicoTesteGravaURA	workdir	index.lua	2020	-	-	-	Receptivo	-	✓
		Receptiva	-	RotaTransfere	Transfere	workdir	index.lua	2021	-	-	-	Receptivo	-	✓
		Receptiva	-	RotaTelegram	BotRegistraAtivo	workdir	index.lua	robo_telegram	-	-	-	Receptivo	-	✓
		Receptiva	-	RotaFacebook	TesteMenu	workdir	index.lua	robo_facebook	-	-	-	-	-	✓
		Receptiva	-	RotaChatclient	ServSemMenu	workdir	index.lua	robo_chatclient	-	-	-	Receptivo	-	✓
		Ativa	Chamada	RotaFacebookA	TesteMenu	workdir	index.lua	robo_facebook	-	-	-	Ativo	10	✓
		Ativa	Chamada	RotaTelegramA	TesteMenu	workdir	index.lua	robo_telegram	-	-	-	Ativo	20	▶
		Ativa	Chamada	RotaChatclientA	TesteMenu	workdir	index.lua	robo_chatclient	-	-	-	Ativo	-	⏸
				RotaTransfereA	Transfere	workdir	index.lua	robo_transfer	-	-	-	Receptivo	-	⊘
				Inativa	Transfere	workdir	index.lua	3333	-	-	-	-	-	⊘

Editar rota
 Copiar rota
 Bloquear rota
 Iniciar rota

Figura 26. Rotas

Nessa janela são apresentadas as rotas cadastradas no sistema.

No lado superior direito da janela há o ícone , que apresenta as informações a seguir.

Exibe as rotas e seus perfis de atendimento das chamadas

Pausar rota

Quando uma rota é colocada em pausa, os agendamentos são desprezados e somente serão executados os posteriores à retirada da pausa.

Caso uma rota for pausada durante a execução de um agendamento, ao ser retirada de pausa, os contatos remanescentes receberão as mensagens programadas neste serviço.

Estado das rotas

- ✓ Ativa
- ⊘ Bloqueada
- ⊖ Inativa
- || Pausada*
- ▶ Executando*

* Somente para rotas ativas





Figura 27. Ícone Informação

No campo **Pesquisar** é possível pesquisar uma palavra ou número e o sistema retornará a busca com os resultados encontrados.



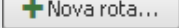
O botão  permite ao usuário selecionar as colunas que serão apresentadas na janela **Rotas**, que são: Ações, Mídia, Tipo, Modo, Nome, Serviço, Revisão, Arquivo, Grupo/Robô, Destino, Origem, Tráfego, Calendário, Sleep e Estado.



O botão , na coluna Ações, permite editar, copiar, bloquear/desbloquear ou iniciar/pausar a rota em questão.

PROCEDIMENTO

Nova Rota - Voz

1. Para criar uma nova rota, pressione o botão  (Figura 26).
2. É apresentada a figura a seguir.

A janela "Nova rota" contém os seguintes elementos:

- Abas: Mídia, Chat, **Voz** (selecionada).
- * Nome: Campo de texto com o valor "Nome da rota".
- * Serviço: Menu suspenso com o valor "Nada selecionado".
- * Revisão: Menu suspenso com o valor "Nada selecionado".
- * Arquivo: Menu suspenso com o valor "Nada selecionado".
- * Grupo: Campo de texto com o valor "Número grupo".
- Calendário: Menu suspenso com o valor "Nada selecionado" e ícone de ajuda (?).
- Filtro de origem: Campo de texto com o valor "Filtro de origem" e ícone de ajuda (?).
- Filtro de destino: Campo de texto com o valor "Filtro de destino" e ícone de ajuda (?).
- Restrição de tráfego: Campo de texto com o valor "Restrição de tráfego" e ícone de ajuda (?).
- Legenda: * Campos obrigatórios.
- Botões: Cancelar e Salvar.

Figura 28. Nova Rota - Voz

Os campos disponíveis para criar uma nova rota de mídia voz são:

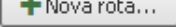
3. **Mídia:** selecione o tipo de mídia que a rota irá atender.
4. **Nome:** é utilizado como identificador da rota e deve ser único. Ao preencher este campo é realizada a validação, identificando a existência de outra rota com nome idêntico. Este campo aceita somente caracteres alfanuméricos, '_' e '-' com no mínimo três e no máximo 50 caracteres.
5. **Serviço:** selecione o serviço de URA que realizará o atendimento da chamada.
6. **Revisão:** selecione a versão do serviço que será utilizada pela rota.
7. **Arquivo:** selecione o nome do arquivo que contém a função que inicia o serviço.
8. **Grupo:** insira o número do ponto de roteamento que receberá a chamada. Este campo aceita somente caracteres numéricos, com no mínimo três e no máximo 10 caracteres. Para cadastro de Rotas de chat, deverá ser selecionado neste campo um Robô, previamente informado no cadastro de robôs.
9. **Calendário:** permite a seleção de um ou mais calendários receptivos de atendimento do serviço. Estando fora das faixas de horário configuradas nos calendários, a rota não é considerada válida para atendimento. Caso um calendário seja configurado sem faixa, é considerado fora do horário e a rota é inválida (mais informações consulte o item Calendários).
10. **Filtro de destino** (somente em PABX 3): critério para escolher uma rota baseado no número de B (número destino da chamada). Neste caso deve-se utilizar a tabela DDR do PABX, pois a verificação é realizada no número discado original. É possível utilizar uma regra para indicar uma faixa (? para números obrigatórios, * para qualquer quantidade de números. Exemplos: 3000 (somente para número 3000), 30?? (qualquer número que começa com 30 e que possua mais 2 dígitos, 3*, qualquer número que começa com 3, independente da quantidade de dígitos que forem recebidos).
11. **Filtro de destino/Siga-me** (somente em PABX 2): critério para escolher uma rota baseado no número de destino ou no último siga-me realizado pela chamada. O número do ramal do siga-me será predominante em relação ao do DDR. Caso não exista siga-me configurado, ou a chamada não passe por nenhum ramal com siga-me, será utilizado o valor do DDR. É possível utilizar uma regra para indicar uma faixa (? para números obrigatórios, * para qualquer quantidade de números. Exemplos:

3000 (somente para número 3000), 30?? (qualquer número que começa com 30 e que possua mais 2 dígitos, 3*, qualquer número que começa com 3, independente da quantidade de dígitos que forem recebidos).

12. **Restrição de tráfego:** critério baseado na quantidade de chamadas simultâneas que a rota pode atender. Caso exceda o limite de chamadas, esta rota não será considerada como válida para as chamadas que entrarem, sendo retirada da lista de rotas válidas para atendimento.

PROCEDIMENTO

Nova Rota - Chat

1. Para criar uma nova rota, pressione o botão  (Figura 26).
2. É apresentada a figura a seguir.

Nova rota [X]

Mídia Chat Voz

Tipo Ativa Receptiva

Modo Chamada Notificação

* Nome

* Serviço

* Revisão

* Arquivo

* Robô

Calendário ?

Sleep ?

* Campos obrigatórios

Cancelar Salvar

Figura 29. Nova Rota - Chat

Os campos disponíveis para criar uma nova rota de mídia chat são:

3. **Mídia:** selecione o tipo de mídia que a rota irá atender.
4. **Tipo:** indica se a rota é Ativa ou Receptiva.
 - **Ativa:** rota que gera a chamada de chat para a lista de contatos associada ao robô.
 - **Receptiva:** rota que atende as chamadas de chat.
5. **Modo:**
 - **Chamada:** chamada ativa que consome licença. Neste modo o usuário poderá interagir com o robô e a primitiva prompt será considerada na script de URA.



- **Notificação:** chamada ativa que não consome licença e permite apenas o envio de mensagens do robô para o usuário. Neste modo, se o usuário interagir com o Robô, uma chamada é gerada e esta consumirá licença.
6. **Nome:** é utilizado como identificador da rota e deve ser único. Ao preencher este campo é realizada a validação, identificando a existência de outra rota com nome idêntico. Este campo aceita somente caracteres alfanuméricos, '_' e '-' com no mínimo três e no máximo 50 caracteres.
 7. **Serviço:** selecione o serviço de URA que realizará o atendimento da chamada.
 8. **Revisão:** selecione a versão do serviço que será utilizada pela rota.
 9. **Arquivo:** selecione o nome do arquivo que contém a função que inicia o serviço.
 10. **Robô:** insira o número do ponto de roteamento que receberá a chamada. Este campo aceita somente caracteres numéricos, com no mínimo três e no máximo 10 caracteres. Para cadastro de Rotas de chat, deverá ser selecionado neste campo um Robô, previamente cadastrado no cadastro de robôs.
 11. **Calendário:** permite a seleção de um ou mais calendários receptivos de atendimento do serviço. Estando fora das faixas de horário configuradas nos calendários, a rota não é considerada válida para atendimento. Caso um calendário seja configurado sem faixa, é considerado fora do horário e a rota é inválida (mais informações consulte o item Calendários).
 12. **Sleep:** critério baseado na quantidade de chamadas simultâneas que a rota pode atender. Caso exceda o limite de chamadas, esta rota não será considerada como válida para as chamadas que entrarem, sendo retirada da lista de rotas válidas para atendimento.

NOTAS

- 1) *As rotas que possuem algum tipo de filtro ou restrição de tráfego configurada possuirão prioridade na análise, ou seja, serão verificadas primeiramente pelo processo. Por exemplo, caso existam duas rotas idênticas cadastradas, porém, uma com filtro de origem e a outra não, a que possui o filtro será priorizada na escolha.*
A sequência de análise, por ordem de importância é:
 - Bloqueio
 - Tráfego
 - Horário de atendimento
 - Destino
 - Origem
 - Erro de script
- 2) *Em casos onde existam duas rotas que possam efetuar o atendimento de alguma chamada e nenhuma delas possua nenhum tipo de filtro ou restrição de tráfego, a primeira rota será selecionada para efetuar o atendimento.*
- 3) *Um mesmo robô não pode ser cadastrado em duas rotas de chat Ativas. Quando um robô é cadastrado em uma rota ativa, este não fica mais disponível para seleção em outro cadastro de rota ativa.*

PROCEDIMENTO

Editar Rota

1. Para editar uma rota clique no ícone  e, em seguida, no ícone  do registro que deseja modificar.
2. É apresentada a janela do cadastro selecionado, conforme mostra a figura a seguir.

Editar rota

Mídia Chat Voz

Tipo Ativa Receptiva

* Nome

* Serviço

* Revisão

* Arquivo

* Robô

Calendário ?

Restrição de tráfego ?

* Campos obrigatórios

Cancelar Salvar

Figura 30. Editar rota



3. Ao editar uma rota não é permitida a alteração dos campos **Mídia**, **Tipo** e **Nome**.
4. Clique sobre o botão **Salvar** para validar as alterações.

PROCEDIMENTO

Copiar Rota

1. A cópia de uma rota permite a criação de um novo registro baseado no selecionado, adicionando o prefixo, *Copia_de_* no nome da rota. Os campos **Mídia**, **Tipo** e **Modo**

ficam desabilitados, não permitindo sua edição. Os demais campos são editáveis inclusive o nome, seguindo as mesmas regras do cadastro de uma nova rota.

2. Para copiar uma rota clique no ícone  e, em seguida, no ícone  do registro que deseja copiar.
3. É apresentada a janela do cadastro selecionado, conforme mostra a figura a seguir.

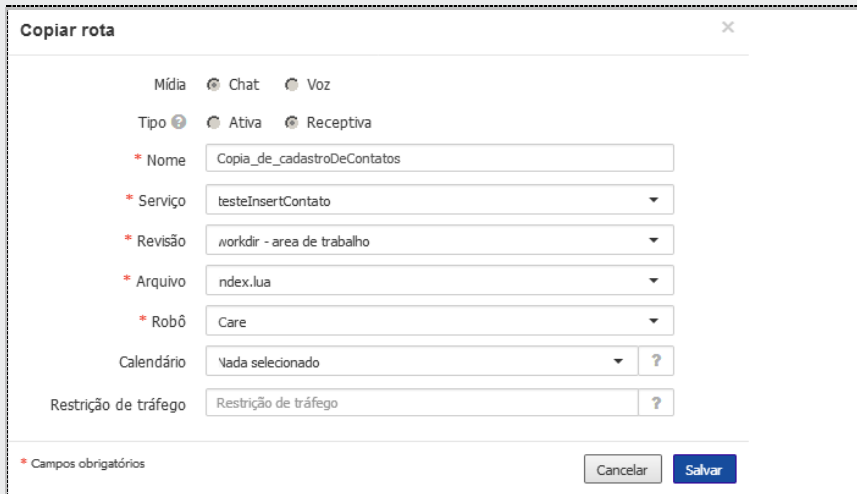





Figura 31. Copiar rota

4. Clique sobre o botão **Salvar** para realizar a cópia.


PROCEDIMENTO

Bloquear/Desbloquear uma rota

1. Para bloquear uma rota clique no ícone  e, em seguida, no ícone  (Figura 26). Em seguida esse mesmo ícone mudará para , o qual tem a função de desbloquear a rota.

PROCEDIMENTO

Remover uma rota

1. Para excluir uma ou mais rotas, selecione o(s) registro(s) desejado(s) e pressione o botão  (Figura 26).
2. É apresentada a mensagem.

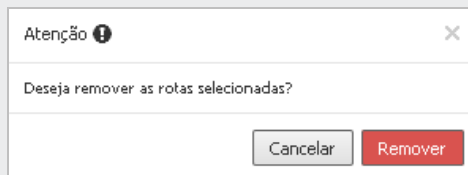








Figura 32. Mensagem remover rota(s)



3. Clique no botão **Remover** para validar a exclusão.

PROCEDIMENTO

Iniciar/pausar uma rota ativa

1. Para iniciar ou pausar uma rota ativa, clique no ícone  e, em seguida, nos ícones  ou , respectivamente (Figura 26).
2. Se a rota estiver pausada, o menu Ações exibirá a opção  **Iniciar rota**, do contrário, se estiver executando, exibirá a opção **Pausar rota**.
3. A coluna Estado exibirá o ícone  somente no momento em que a rota ativa estiver enviando mensagens.
4. Nas demais situações, a coluna Estado exibirá o ícone , que indica que a rota está ativa.

NOTAS

- 1) Se uma rota ativa for pausada enquanto estiver em execução (enviando mensagens), ela ficará pausada até que o usuário clique no botão .
- 2) As chamadas de um agendamento que ainda não tenham sido iniciadas e, cujo horário de início coincida com a rota no estado pausada, não serão geradas até que o usuário clique no botão .

PROCEDIMENTO

Editar Serviço a partir da listagem de rotas

1. Para editar o serviço de uma rota a partir da listagem de rotas, vá à coluna Serviços e clique no link do serviço que deseja editar, como mostra a figura a seguir.

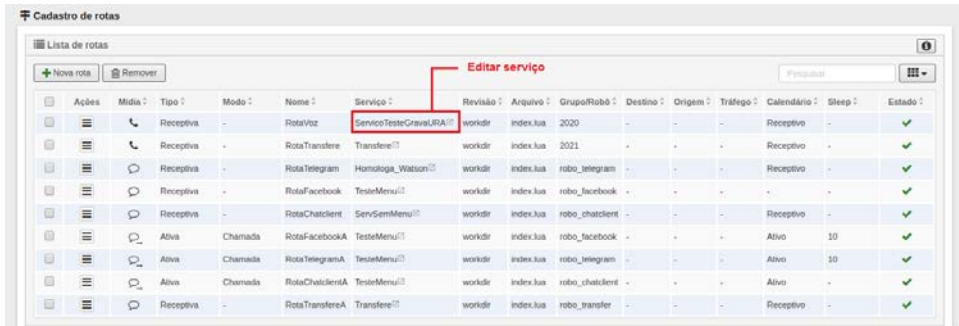


Figura 33. Editar serviço a partir da listagem de rotas

2. Será aberta a interface de Edição de Serviços, mostrada a seguir.

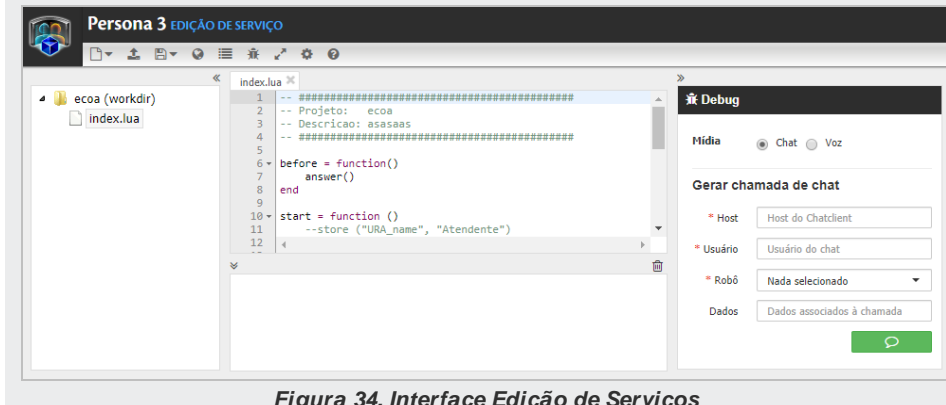


Figura 34. Interface Edição de Serviços

NOTAS

- 1) *A interface de edição de serviços requer licença específica.*
- 2) *O link de acesso à interface de edição de serviço só estará disponível para usuários com privilégio de edição de serviços.*
- 3) *Para mais detalhes sobre esta funcionalidade, acesse o capítulo Interface de Edição de Serviços.*

SERVIÇOS

No **Persona 3**, um serviço pode ser considerado como a URA, pois nesse serviço ficarão armazenadas as *scripts* lua contendo o código da URA, bem como, caso necessário, os arquivos de áudio.

É possível cadastrar, remover, editar, copiar e importar um serviço.

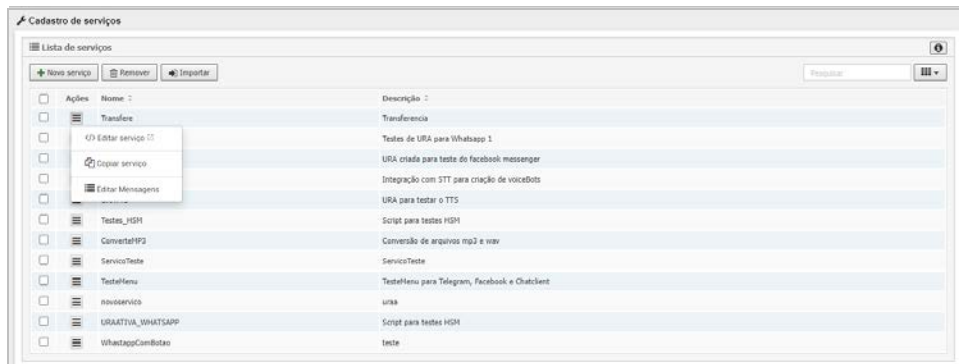



Figura 35. Cadastro de Serviços


No lado superior direito da janela há o ícone , que apresenta as informações a seguir.


Exibe os serviços de URA cadastrados, disponíveis para utilização nas rotas



Figura 36. Ícone Informação

No campo **Pesquisar** é possível pesquisar uma palavra ou número e o sistema retornará a busca com os resultados encontrados.

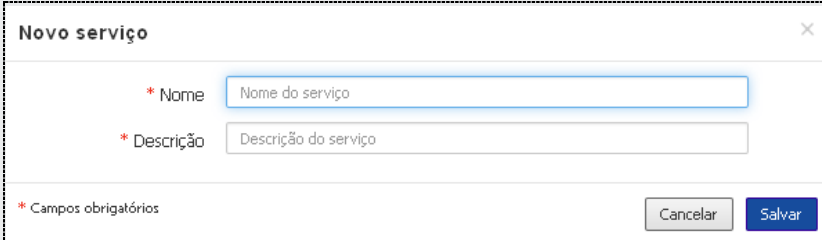
O botão  permite ao usuário selecionar as colunas que serão apresentadas na janela **Serviços**, que são: Ações, Nome e Descrição.

O botão  permite ao usuário, com os devidos privilégios, editar um serviço, copiar um serviço e editar mensagens de um serviço.

PROCEDIMENTO

Novo Serviço

1. Para criar um novo serviço pressione o botão  (Figura 35),
2. É apresentada a figura a seguir.



A captura de tela mostra uma janela de diálogo intitulada "Novo serviço" com um ícone de fechamento (X) no canto superior direito. O formulário contém dois campos de entrada de texto:

- Um campo rotulado "* Nome" com o placeholder "Nome do serviço".
- Um campo rotulado "* Descrição" com o placeholder "Descrição do serviço".

Na base da janela, há uma linha de texto que diz "* Campos obrigatórios" e dois botões: "Cancelar" (botão cinza) e "Salvar" (botão azul).



Figura 37. Novo Serviço

Os campos disponíveis para criar um novo serviço são:

3. **Nome:** é utilizado como o identificador do serviço. Ao preencher este campo é realizada a validação, identificando a existência de outro serviço com nome idêntico, pois não é permitido existir mais de um serviço com mesmo nome. Este campo aceita somente caracteres alfanuméricos, '_' e '-' com no mínimo três e no máximo 20 caracteres.
4. **Descrição:** é utilizado para realizar uma breve descrição do serviço a fim de facilitar sua identificação. Este campo aceita caracteres alfanuméricos exceto '<', '>' e "'" com quantidade de 1 a 150 caracteres.
5. Ao cadastrar um novo serviço é criada uma estrutura de diretórios que armazenará os arquivos que farão parte deste serviço. O nome do diretório raiz é o mesmo nome do serviço. Além disso, é gerada a revisão inicial (workdir) para desenvolvimento da URA.

PROCEDIMENTO

Editar Serviço

1. Para editar um serviço clique no ícone  e, em seguida, no ícone  do registro que deseja modificar.
2. É apresentada a interface de Edição de Serviços, conforme mostra a figura a seguir.

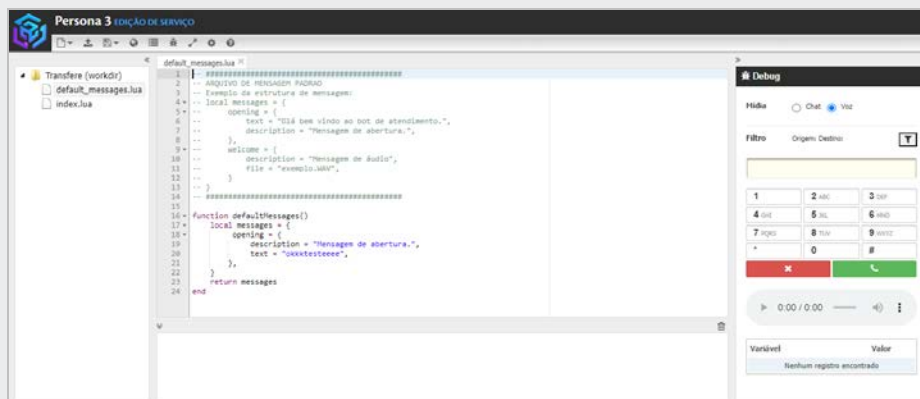


Figura 38. Interface Edição de Serviços

NOTAS

Para ter acesso a edição de serviços, é necessário adquirir licença específica, além da licença de acesso a interface do **Persona 3**. Para mais informações sobre esta funcionalidade, acesse o capítulo Interface de Edição de Serviços.

PROCEDIMENTO

Copiar Serviço

1. A cópia de um serviço permite a criação de um novo registro baseado no selecionado, adicionando o prefixo, *Copia_de_* no nome do serviço e a descrição e revisões são editáveis. Somente o campo pacote não permite sua edição.

Copiar serviço

* Serviço: SRVteste

* Nome: Copia_de_SRVteste

* Descrição: Desc SRVteste

* Revisões: 1 - teste, workdir - area de trabalho

* Campos obrigatórios

Cancelar Salvar

Figura 39. Copiar Serviço

2. Ao realizar a cópia de um serviço, uma nova estrutura de diretórios é criada baseada no novo nome do serviço.
3. O campo **Revisões** é de preenchimento obrigatório e permite a seleção de quais revisões serão copiadas. Os demais campos seguem as mesmas regras do cadastro de um novo serviço.

PROCEDIMENTO

Editar Mensagens





1. A edição de mensagens permite ao usuário, com privilégio, fazer alterações nos conteúdos das mensagens divulgadas pelas URAs ou chatbots sem necessidade de alterações nas scripts.
2. Na Lista de Serviços (Figura 35), clique no ícone  do serviço cuja mensagem deseja editar, e, em seguida, no ícone  Editar Mensagens. Será aberta a Lista de Mensagens do serviço:



Figura 40. Lista de mensagens do serviço

3. Clique no ícone  da mensagem que deseja editar e, em seguida, no ícone  Editar Mensagens.
4. Nas mensagens de texto, é possível alterar todo o conteúdo, com um limite de 1.000 caracteres.

Editar Mensagens ✕

* Descrição

* Conteúdo

* Campos obrigatórios

Cancelar Salvar

Figura 41. Editar mensagem de texto

5. Nas mensagens de áudio, somente é possível escutar a mensagem atual em uso e fazer o upload de um novo arquivo.

Editar Mensagens X

Descrição Mensagem 3 de voz

Áudio atual 00:00 00:10

* Novo áudio

* Campos obrigatórios

Cancelar Salvar

Figura 42. Editar mensagem de áudio


6. Se o formato de áudio do novo arquivo for suportado pelo Persona, será apresentado o player para executá-lo.
7. Clique em **Salvar** para substituir o áudio atual pelo novo áudio ou, em cancelar, para ignorar a operação.

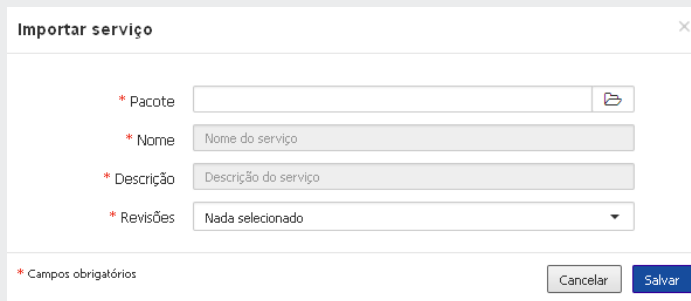
NOTA

Para ter acesso à edição de mensagens, é preciso ter privilégio de edição de serviços ou privilégio de alteração de mensagens.

PROCEDIMENTO

Importação de Serviços

1. Para importar um serviço pressione o botão  (Figura 35).
2. É apresentada a figura a seguir.



O formulário "Importar serviço" contém os seguintes campos:

- * Pacote: Campo de texto com ícone de pasta.
- * Nome: Campo de texto com o valor "Nome do serviço".
- * Descrição: Campo de texto com o valor "Descrição do serviço".
- * Revisões: Menu suspenso com o valor "Nada selecionado".

Na base do formulário, há o texto "* Campos obrigatórios" e dois botões: "Cancelar" e "Salvar".

Figura 43. Importar Serviço

3. Permite a criação de um novo registro baseado em um pacote exportado através da Interface de Edição de Serviços.
4. Ao selecionar um arquivo válido, os campos **Nome**, **Descrição** e **Revisões** são preenchidos com os valores já existentes no arquivo exportado e é permitida a edição de todos os campos, inclusive selecionar somente as revisões necessárias.
5. O preenchimento dos campos seguem as mesmas regras do cadastro de um novo serviço e cópia de serviço.

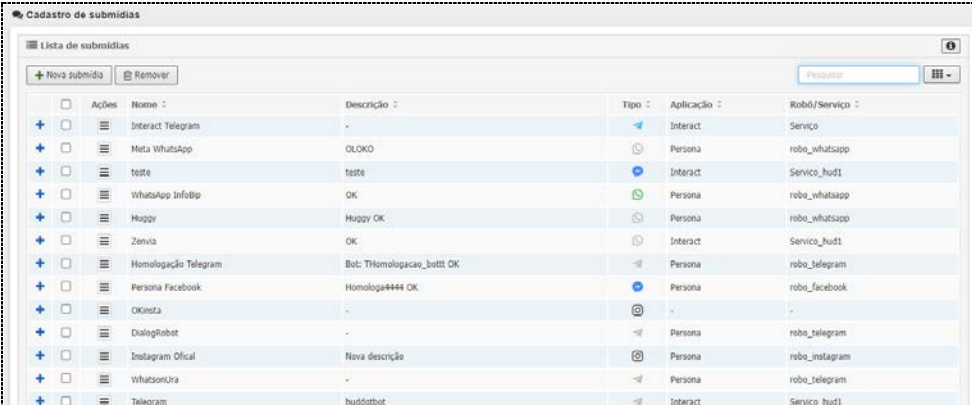
SUBMÍDIAS

As **Submídias** são utilizadas para identificar o ponto de entrada das chamadas de *chat*. Elas são cadastradas com um *token* do *Telegram*, *Facebook Messenger*, *Instagram Messenger* e *WhatsApp*. As submídias podem receber chamadas diretamente de um usuário do respectivo aplicativo, sem a necessidade de a chamada passar por outro processo de atendimento, co-

mo o Interact.

O cadastro de submídias, além de exibir a listagem com todas as submídias configuradas, permite adicionar, remover e editar uma submídia.


Na Lista de submídias são exibidas as colunas **Ações**, **ID**, **Nome**, **Descrição**, **Tipo**, **Aplicação** e **Robô/Serviço** da submídia. A coluna tipo exibe o ícone da submídia, quando este for *Facebook Messenger*, *Instagram Messenger*, *Telegram* e/ou *WhatsApp*. Quando a submídia estiver desativada, seu ícone aparecerá na cor cinza.



The screenshot shows a web interface titled "Cadastro de submídias" with a sub-header "Lista de submídias". It includes buttons for "+ Nova submídia" and "Remover", and a search bar. Below is a table with the following columns: Ações, Nome, Descrição, Tipo, Aplicação, and Robô/Serviço. The table contains 14 rows of data, each with a blue plus icon in the "Ações" column.

Ações	Nome	Descrição	Tipo	Aplicação	Robô/Serviço
+ [icon]	Interact Telegram	-	[Telegram icon]	Interact	Serviço
+ [icon]	Meta WhatsApp	OLOKO	[WhatsApp icon]	Persona	robo_whatsapp
+ [icon]	teste	teste	[Interact icon]	Interact	Servico_hud1
+ [icon]	WhatsApp InfoBip	OK	[WhatsApp icon]	Persona	robo_whatsapp
+ [icon]	Huggy	Huggy OK	[Persona icon]	Persona	robo_whatsapp
OK [icon]	Zenvia	OK	[Interact icon]	Interact	Servico_hud1
+ [icon]	Homologação Telegram	Bot: THomologacao_bottt OK	[Disabled Telegram icon]	Persona	robo_telegram
+ [icon]	Persona Facebook	Homologa4444 OK	[Facebook icon]	Persona	robo_facebook
+ [icon]	OKinsta	-	[Instagram icon]	-	-
+ [icon]	DialogRobot	-	[Disabled Telegram icon]	Persona	robo_telegram
+ [icon]	Instagram Oficial	Nova descrição	[Instagram icon]	Persona	robo_instagram
+ [icon]	WhatssonOra	-	[Disabled Telegram icon]	Persona	robo_telegram
+ [icon]	Telegram	huddotbot	[Disabled Telegram icon]	Interact	Servico_hud1

Figura 44. Cadastro de Submídias

Na listagem de submídias é possível visualizar os detalhes de uma submídia clicando no ícone , na primeira coluna da tabela.

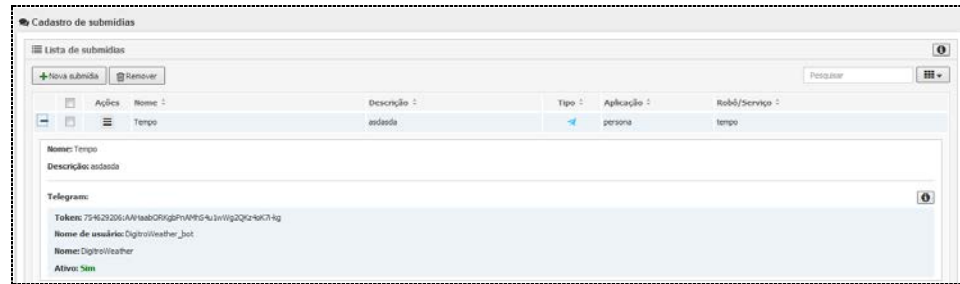


Figura 45. Detalhe da Submídia

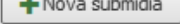
As informações apresentadas nos detalhes de uma submídia são:

- **Telegram**
 - **Token:** chave gerada no Telegram no momento da criação da submídia.
 - **Nome de usuário:** nome que deverá ser utilizado para pesquisar a submídia no aplicativo do Telegram.
 - **Nome:** nome que será exibido no aplicativo do Telegram.
- **Facebook/Instagram Messenger**
 - **Chave desafio:** valor que deve ser informado no cadastro da página do Facebook/Instagram.
 - **Nome da página:** nome da página que é atendida pela submídia.
 - **ID da página:** código fornecido pelo Facebook/Instagram no momento do cadastro da página.
 - **Token:** código fornecido pelo Facebook/Instagram no momento do cadastro da página.
 - **Chave do aplicativo:** código fornecido pelo Facebook/Instagram no momento do cadastro da página.

- **WhatsApp**
 - **Broker:** provedor para integração com a API do WhatsApp.
 - **Interlocutor:** *Nome, Telefone, Nome e telefone ou Telefone e nome do usuário na conversa.*
 - **URL Webhook:** URL disponibilizada pelo cliente para o Broker se comunicar com a plataforma encaminhando as mensagens recebidas.
 - **Token Dígitro:** token utilizado para integrações com esta submídia. Por exemplo: para utilizar a API de envio de mensagem HSM para o WhatsApp é necessário informar esse token.
 - **Telefone:** número de telefone utilizado no WhatsApp da conta criada na Zenvia. Este número deverá estar no seguinte padrão: DDI + DDD + número. Ex.: 5548999999999.
 - **Conta:** chave informada quando o usuário acessa a sua conta na Infobip.
 - **Usuário:** usuário utilizado para logar na Infobip.
 - **Senha:** senha utilizada para logar na Infobip.
 - **URL Base:** campo da URL específico para cada usuário. Essa informação pode ser acessada através da URL.

PROCEDIMENTO

Nova Submídia

1. Para criar uma nova submídia pressione o botão  (Figura 45),
2. É apresentada a figura a seguir.

Nova submídia

Ativar submídia

* Nome

Descrição

* Tipo

Chave desafio

* ID da página

* Token do Messenger

* Chave do aplicativo

Aplicação

* Campos obrigatórios

Figura 46. Nova submídia

Os campos disponíveis para criar uma nova submídia são:

3. **Ativar submídia:** utilizado para ativar ou desativar a submídia.
4. **Nome:** utilizado como o identificador da submídia não é permitido duas submídias com o mesmo nome.
5. **Descrição:** utilizado para realizar uma breve descrição da submídia a fim de facilitar sua identificação. Este campo aceita caracteres alfanuméricos exceto '<', '>' e 'admitte de 1 a 150 caracteres.
6. **Tipo:** utilizado para selecionar o tipo da submídia que está sendo cadastrada. Os tipos de submídia podem ser **Facebook Messenger**, **Instagram Messenger**, **Telegram** ou **WhatsApp**.
7. Ao selecionar o tipo **Facebook Messenger** são exibidos os campos **Chave desafio**, **ID da página**, **Token do Messenger**, **Chave do aplicativo** e o botão **Assistente de configuração**.

8. **Chave desafio:** nas configurações do Facebook Messenger, ao criar um WebHook, é solicitado ao usuário uma chave de validação (no campo **Verificar token**). Deve-se copiar o valor fornecido pelo **Persona 3** no campo **Chave desafio** para utilizá-lo no cadastro do token do Facebook Messenger.
9. **ID da página:** nas configurações do Facebook Messenger, o usuário seleciona a página que será associada ao Messenger. Após a concluir as configurações, o administrador da conta deverá acessar a página do Facebook e no item **Sobre**, copiar o identificador da página e colá-lo neste campo de cadastro do **Persona 3**.
10. **Token do Messenger:** nas configurações do Facebook Messenger, ao selecionar a página que será utilizada com o Messenger, o Facebook fornece um token, que deve ser copiado e colado neste campo de cadastro do **Persona 3**.
11. **Chave do aplicativo:** após finalizar as configurações do Facebook Messenger, deve-se copiar o campo **Segredo da aplicação** do Facebook e colá-lo neste campo de cadastro do **Persona 3**.
12. **Assistente de configuração** - o **Assistente de configuração** (página 74) do Facebook Messenger é um facilitador na configuração e integração de um robô do Messenger. Ele fornece um passo a passo para o preenchimento de todas as informações necessárias no cadastro de um robô do Facebook.
13. Ao selecionar o tipo **Instagram Messenger** são exibidos os campos **Chave desafio**, **ID da página**, **Webhook do Instagram Messenger**, **Token do Messenger**, **Chave do aplicativo** e o botão **Assistente de configuração**.
14. **Chave desafio:** nas configurações do Instagram Messenger, ao criar um WebHook, é solicitado ao usuário uma chave de validação (no campo **Verificar token**). Deve-se copiar o valor fornecido pelo **Persona 3** no campo **Chave desafio** para utilizá-lo no cadastro do token do Instagram Messenger.
15. **ID da página:** nas configurações do Instagram Messenger, o usuário seleciona a página que será associada ao Messenger. Após concluir as configurações, o administrador da conta deverá copiar o identificador da página e colá-lo no campo **ID da página**.
16. **Webhook do Instagram:** informa à plataforma sobre novas chamadas de chat ou mensagens enviadas.

17. **Token do Messenger:** nas configurações do Instagram Messenger, ao selecionar a página que será utilizada com o Messenger, o Instagram fornece um token, que deve ser copiado e colado neste campo de cadastro do **Persona 3**.
18. **Chave do aplicativo:** após finalizar as configurações do Instagram Messenger, deve-se copiar o campo **Segredo da aplicação** do Instagram e colá-lo neste campo de cadastro do **Persona 3**.
19. **Assistente de configuração** - o **Assistente de configuração** (página 74) do Facebook Messenger é um facilitador na configuração e integração de um robô do Messenger. Ele fornece um passo a passo para o preenchimento de todas as informações necessárias no cadastro de um robô do Instagram.
20. Ao selecionar o tipo **Telegram** é exibido o campo **Token do Telegram**:
 - **Token do Telegram:** utilizado para inserir o código gerado pelo Telegram ao criar um novo Bot. Para mais informações de como criar um novo Bot, acesse <https://core.telegram.org/bots>.
21. Ao selecionar o tipo **WhatsApp**, por default, é exibido os campos **Broker**, **Interlocutor**, **URL Webhook**, **Token Dígito**, **Telefone**, **Conta**, **Usuário**, **Senha** e **URL Base**.
 - **Broker:** selecione uma das opções: **Infobip**, **Zenvia**, **Cloud API** ou **Huggy**.

NOTAS

O cliente precisa contratar o serviço de integração com o WhatsApp com uma dessas empresas: **Infobip**, **Zenvia**, **Cloud API** ou **Huggy**.

22. Ao selecionar **Infobip** será exibido os campos:
 - **Broker:** provedor para integração com a API do WhatsApp.
 - **Interlocutor:** Nome, Telefone, Nome e telefone ou Telefone e nome do usuário na conversa.
 - **URL Webhook:** URL disponibilizada pelo cliente para o Broker se comunicar com a plataforma encaminhando as mensagens recebidas.

- **Token Dígitro:** chave gerada para integrações com esta submídia. Por exemplo: para utilizar a API de envio de mensagem HSM para o WhatsApp é necessário informar esse token
 - **Telefone:** número do telefone utilizado no WhatsApp da conta criada na Infobip. Este número deverá estar no seguinte padrão: DDI + DDD + Número. Ex.: 5548999999999.
 - **Conta:** chave informada quando o usuário acessa a sua conta Infobip.
 - **Usuário:** usuário utilizado para logar na Infobip.
 - **Senha:** senha utilizada para logar na Infobip.
 - **URL Base:** campo da URL específico para cada usuário. Essa informação pode ser acessada através da URL <https://dev.infobip.com/getting-started/base-url>.
23. Ao selecionar **Zenvia** será exibido os campos:
- **Interlocutor:** Nome, Telefone, Nome e telefone ou Telefone e nome do usuário da conversa.
 - **URL Webhook:** URL disponibilizada pelo cliente para o Broker se comunicar com a plataforma encaminhando as mensagens recebidas.
 - **Token Dígitro:** token utilizado para integrações com esta submídia. Por exemplo: para utilizar a API de envio de mensagem HSM para o WhatsApp é necessário informar esse token.
 - **Telefone:** número de telefone utilizado no WhatsApp da conta criada na Zenvia. Este número deverá estar no seguinte padrão: DDI + DDD + Número. Ex.: 5548999999999.
 - **Token Zenvia:** token utilizado para enviar as mensagens ao Broker. Este token é criado nas configurações da conta da Zenvia para utilização da API.
24. Ao selecionar **Cloud API** será exibido os campos:
- **Broker:** provedor para integração com a API do WhatsApp.
 - **Interlocutor:** Nome, Telefone, Nome e telefone ou Telefone e nome do usuário da conversa.

- **URL Webhook:** URL disponibilizada pelo cliente para o Broker se comunicar com a plataforma encaminhando as mensagens recebidas.
- **Token Dígito:** chave gerada para integrações com esta submídia. Por exemplo: para utilizar a API de envio de mensagem HSM para o WhatsApp é necessário informar esse token.
- **Telefone:** número do associado a conta da META. Este número deverá estar no seguinte padrão: DDI + DDD + Número. Ex.: 5548999999999.
- **Id. Telefone:** identificação gerado no site da META para utilização da API.
- **Token Permanente:** token gerado no site da META para utilização da API.
- **Aplicação:** utilizado para a aplicação em que a submídia será utilizada. As opções são Interact e Persona 3. Ao selecionar a aplicação Interact é exibido o campo Serviços. Se a aplicação selecionada foi Persona 3, serão exibidos os robôs cadastrados.

25. Ao selecionar **Huggy** será exibido os campos:

- **Broker:** provedor para integração com a API do WhatsApp.
- **URL Webhook:** URL disponibilizada pelo cliente para o Broker se comunicar com a plataforma encaminhando as mensagens recebidas.
- **Telefone:** número de telefone utilizado no WhatsApp da conta criada na Huggy.
- **Código da empresa:** número da empresa presente no sistema da Huggy.
- **Token – Webhook:** informação utilizada para validar o Webhook que a Huggy efetua em nosso sistema.
- **Token – Via API:** token utilizado para enviar as mensagens ao Broker. Este token é criado nas configurações da conta da Huggy.

26. **Aplicação:** utilizado para selecionar a aplicação em que a submídia será utilizada. As opções são Interact e **Persona 3**. Ao selecionar a aplicação Interact é exibido o campo Serviços. Se a aplicação selecionada for **Persona 3**, serão exibidos os robôs cadastrados.

- **Serviços:** utilizado para selecionar o serviço do Interact que utilizará a submídia

configurada.

- **Robô:** utilizado para selecionar o serviço do Interact que utilizará a submídia configurada.

PROCEDIMENTO

Assistente de configuração


1. O botão  *Assistente de configuração* *exibe a interface (passo 1 de 3) contendo os pré-requisitos para sua configuração.*



Figura 47. Passo 1 de 3

2. Ao clicar em **Avançar** será exibido o passo 2 de 3, onde serão realizadas as configurações gerais e o preenchimento dos campos necessários para o cadastro de uma submídia no **Persona 3**.



Figura 48. Passo 2 de 3

3. Ao preencher este campo, ao final do passo 3, este valor será preenchido automaticamente no campo **Chave do aplicativo** na interface do **Persona 3**.

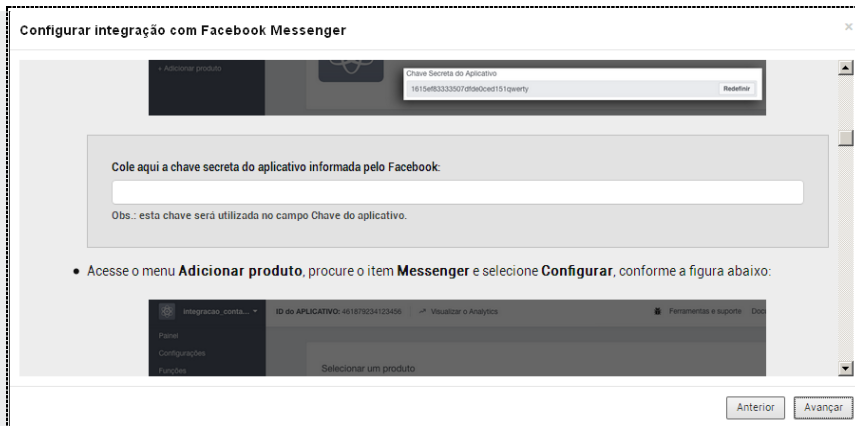


Figura 49. Passo 3 de 3

- Ao preencher este campo, ao final do passo 3, este valor será preenchido automaticamente no campo **Token do Messenger** na interface de cadastro de submídias do **Persona 3**.

Configurar integração com Facebook Messenger

Cole aqui o token gerado pelo Facebook:

Obs.: este token será utilizado no campo "Token do Messenger".

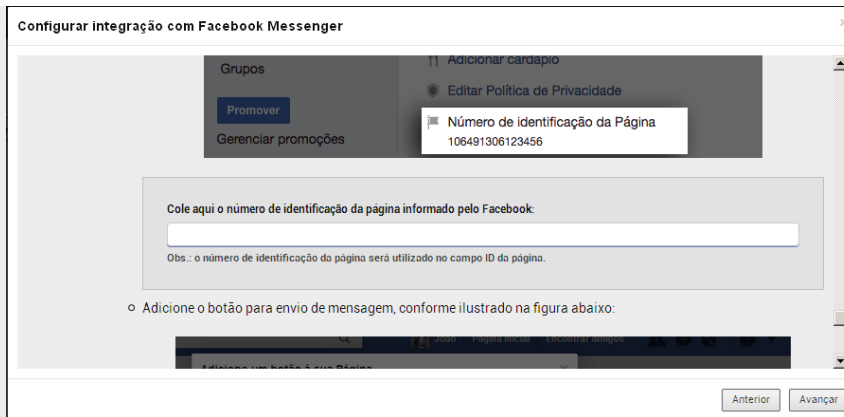
- Webhooks: esta configuração permite que o Facebook informe à plataforma sobre novas chamadas de chat ou mensagens enviadas. Abra a configuração do webhook do Facebook, conforme ilustrado na figura abaixo:

Anterior Avançar

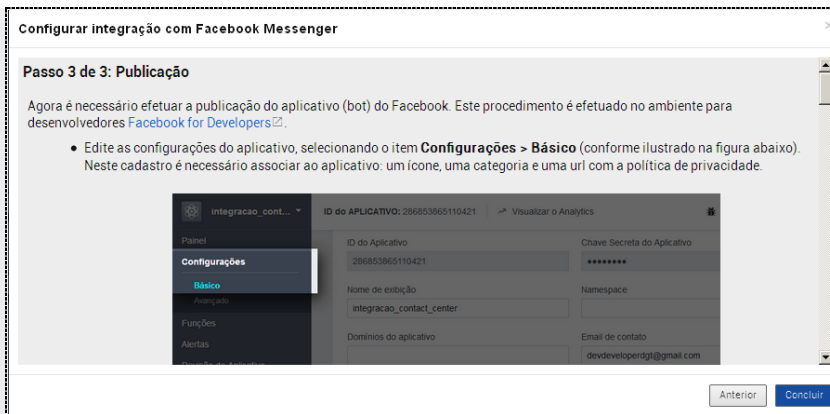
5. *Este código é o mesmo que aparece no campo **Chave desafio** na interface de cadastro de submídias do **Persona 3**.*



6. Ao preencher este campo, ao final do passo 3, este valor será preenchido automaticamente no campo **ID da página** na interface de cadastro de submídias do **Persona 3**.



7. Ao final do passo 2, após clicar em **Avançar**, será exibido o passo 3 de 3 onde será realizada a publicação do aplicativo no ambiente para desenvolvedores do Facebook.



8. No passo 3, após clicar em **Concluir**, os campos **ID da página**, **Token do Messenger** e **Chave do aplicativo** serão preenchidos automaticamente possibilitando o cadastro da submídia do Facebook Messenger no **Persona 3**, como mostra a figura a seguir.

Nova submídia

Ativar submídia

* Nome

Descrição

* Tipo

Chave desafio

* ID da página

* Token do Messenger

* Chave do aplicativo

Aplicação



* Campos obrigatórios

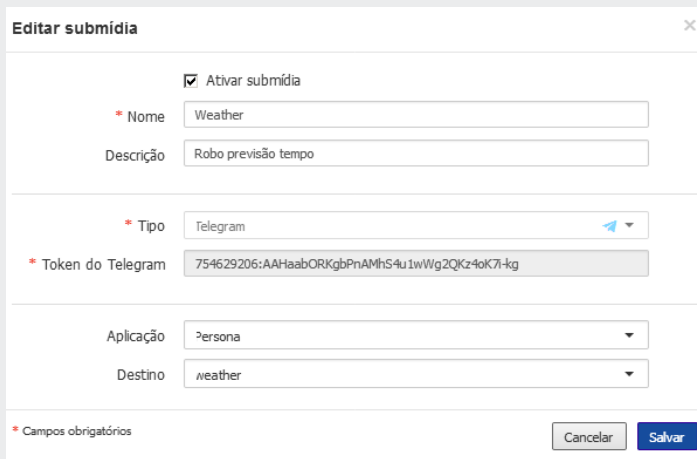
ATENÇÃO

Ao utilizar o Telegram® ou Facebook Messenger® deve ser respeitada a característica de cada mídia social conforme descrito em seu respectivo manual de utilização.

PROCEDIMENTO

Editar Submídia

1. Para editar uma submídia clique no ícone  e, em seguida, no ícone  da submídia que deseja modificar.
2. É apresentada a figura a seguir.



A janela de diálogo "Editar submídia" contém os seguintes campos e controles:

- Uma caixa de seleção marcada "Ativar submídia".
- Um campo de texto obrigatório "Nome" com o valor "Weather".
- Um campo de texto "Descrição" com o valor "Robo previsão tempo".
- Um campo de seleção obrigatório "Tipo" com o valor "Telegram".
- Um campo de texto obrigatório "Token do Telegram" com o valor "754629206:AAHaabORKgbPhAMhS4u1wWg2QKz4oK7l-kg".
- Um campo de seleção "Aplicação" com o valor "Persona".
- Um campo de seleção "Destino" com o valor "weather".
- Um ícone de fechamento "X" no canto superior direito.
- Um ícone de ajuda (?) no canto inferior direito.
- Botões "Cancelar" e "Salvar" no canto inferior direito.
- Uma legenda "* Campos obrigatórios" no canto inferior esquerdo.

Figura 50. Editar submídia

3. Edite os campos dessa janela conforme desejar, salvo o campo **Tipo** que não permite edição.
4. Clique sobre o botão **Salvar** para validar as alterações.

ATENÇÃO

- *Na edição de uma submídia o campo Tipo não pode ser editado.*
- *Os campos **ID da página** (Facebook Messenger), o campo **Token do Telegram** (Telegram) e o campo **Broker** (WhastApp) ficam desabilitados, não sendo permitida sua edição.*
- *Os demais campos seguem as mesmas regras do cadastro de uma nova submídia.*

PESSOAS

O cadastro de Pessoas permite consultar, cadastrar e editar os usuários ou suas informações de acordo com o privilégio de acesso do usuário logado.

É apresentada a janela a seguir.

Pessoas

- ▶ **Pessoas**
- Estrutura Organizacional
- Perfil de Acesso
- Exportação de Contatos
- Importação de Contatos
- Token/Autenticação

Pessoas

[-] Filtros

Tipo de Pessoa

Usuários Operadores de Interfaces
 Usuários Internos
 Contatos

Pesquisa nomes com pronúncia semelhante ?

Nome: ?

Login:

Telefones:

Perfis: ▼

Estrutura Organizacional

Filtrar itens 11 a 14 de 14 Itens por Página:

Lista de Pessoas
 Novo Usuário Novo Contato

<input type="checkbox"/>	Nome	Login	Perfis	Telefones	Email
<input type="checkbox"/>	teste03	teste03	TarIFone Web Administrador; TarIFone Web Consulta; Administrador Voice Manager; Supervisor Voice Manager; Supervisão de Alarmes Consulta; TarIFone Web Relatórios Administrativos; TarIFone Web Relatórios para a Gerência; TarIFone Web Relatórios para o Usuário		
<input type="checkbox"/>	teste04			8111	
<input type="checkbox"/>	TST sem permissão	tst			
<input type="checkbox"/>	xcontato02			00022; 0002	


Figura 51. Pessoas

NOTA

Para mais informações, consulte o manual Cadastro de Pessoas.

4

INTERFACE DE EDIÇÃO DE SERVIÇOS

Essa interface é acessada por meio do botão  **Editar Serviço**, ao selecionar o serviço desejado, no menu **Serviços**. Para esta opção estar disponível é necessário aquisição de licença adicional.

A interface de edição de serviços possui cinco partes principais: barra de botões, painel da árvore de diretórios e arquivos, editor de desenvolvimento, painel de debug e painel console de eventos.

Interface de Edição de
Serviços
CAPÍTULO 4

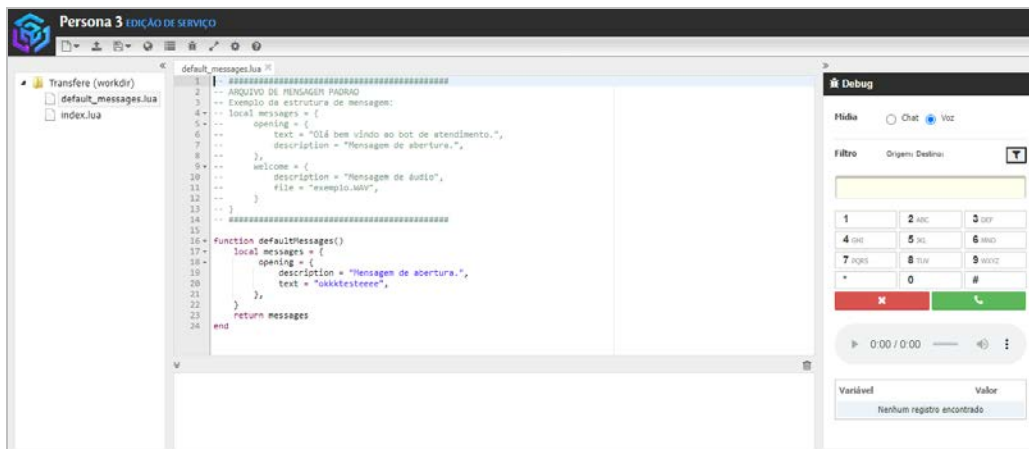


Figura 52. Interface de Edição de Serviço

É possível Expandir/Recolher o painel a fim de dar maior visibilidade à área de desenvolvimento, através dos ícones **Expandir/Recolher o painel à esquerda**, **Expandir/Recolher o painel à direita** e **Expandir/Recolhe o painel inferior**.

O recolhimento dos painéis melhora a visualização do código do serviço de URA, conforme mostra a figura a seguir.

Interface de Edição de
Serviços
CAPÍTULO 4



```
index.lua
79 end
80 end
81
82 - function menu3 ()
83   checkpoint("Menu3")
84   message (text="Opções de menu com 100, 500 e 1500 caracteres. Teste as 3 opções.")
85   teste = prompt (chat_timeout=60)
86   if teste=="100" then
87     message (text="100 - Este Módulo é um DTD XML criado pela Dígítro para permitir a elaboração de diálogos de voz, .")
88     menu.previous(1)
89   else if teste=="500" then
90     message (text="500 - Este Módulo é um DTD XML criado pela Dígítro para permitir a elaboração de diálogos de voz, .100 - Este Módulo é um DTD XML criado pela Dígítro para permitir a
91     menu.previous(1)
92   else
93     message (text="1.500 - Este Módulo é um DTD XML criado pela Dígítro para permitir a elaboração de diálogos de voz, .100 - Este Módulo é um DTD XML criado pela Dígítro para permitir
94     menu.previous(1)
95   end
96 end
97 end
98
99 - function menu4 ()
100   checkpoint("Menu4")
101   typing.start(1)
102   -- sleep(20)
103   message (text="1 - Negrito \n 2 - Itálico \n 3 - sublinhado \n 4 - 🍌 emoji \n 5 - Quebra \n de linha")
104
```

Figura 53. Visualização de código do serviço URA

BARRA DE BOTÕES

A barra de botões é composta por nove botões, conforme mostra a figura a seguir.



Figura 54. Barra de Botões

Novo arquivo 

Possibilita, através da lista de seleção, a criação de um novo arquivo dos tipos JSON, TXT e LUA ou ainda a criação de um diretório. Existem algumas restrições que devem ser levadas em consideração do momento da criação do arquivo. São elas:

- Um diretório pode ser criado somente dentro de outro diretório.
- É possível criar somente um subnível de diretórios.
- Um arquivo pode ser criado somente dentro de um diretório.
- Não são permitidos dois diretórios com o mesmo nome.
- Não são permitidos dois arquivos com mesmo nome mesmo que em diretórios diferentes. O nome não depende da extensão do arquivo.
- Um arquivo obrigatoriamente precisa possuir uma extensão e as extensões permitidas são: JSON, TXT e LUA.

Enviar arquivo

Possibilita o envio de arquivos para o projeto.

- As extensões permitidas são mp3, WAV, JSON, TXT e LUA.
- O nome dos arquivos de áudio deverão estar no formato M_0xxx.ext, onde xxx são dígitos e ext a extensão em maiúsculo. Ex.: M_01234.MP3.
- Ao realizar o envio de um áudio, este será automaticamente convertido para o formato aceito na reprodução, sendo mantida a extensão original. A conversão poderá alterar a qualidade do áudio.
- Formatos aceitos pela plataforma:

	WAV	MP3
Taxa de bits	32kbps	8kbps
Taxa de amostragem	8kHz	8kHz
Quantidade de Canais	1 (mono)	1 (mono)
Formato de áudio	IMA ADPCM	

Salvar

Possibilita, através da lista de seleção, salvar um ou todos os arquivos abertos no editor.

- Atalho no editor: Salvar (Ctrl + s).
- Atalho no editor: Salvar todos (Ctrl + Shift + s).

Publicar revisão

Possibilita a publicação de uma nova revisão do serviço contendo as alterações realizadas e seus arquivos.

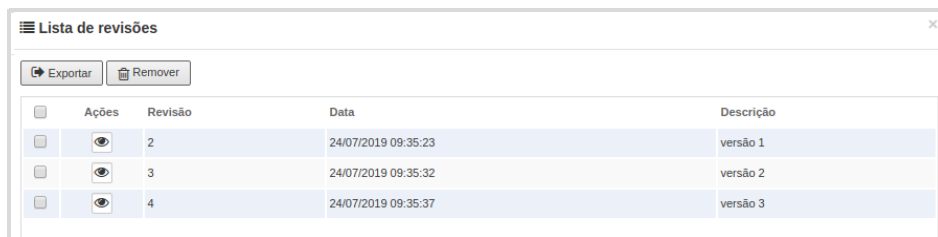
Após a edição de um serviço é possível 'fechar' uma versão com as alterações realizadas no projeto. Após a publicação, não será permitida a alteração desta revisão. Caso o desenvolvedor queira fazer alguma alteração no projeto, deverá realizar no workdir e publicar uma nova revisão.

O campo de descrição é de preenchimento obrigatório e aceita caracteres alfanuméricos exceto '<', '>' e ' ' com quantidade de 1 a 150 caracteres.

Listar revisões

Apresenta a listagem de revisões publicadas e possibilita a exportação de uma ou mais revisões.

Na listagem de revisões é possível exportar ou remover as revisões selecionadas. Ao exportar uma revisão é gerado um arquivo com extensão TGZ contendo todos os arquivos de scripts, áudios, etc, existentes no projeto. Não é possível realizar a exportação do workdir. Ao clicar nesse botão é apresentada a figura a seguir.



A interface 'Lista de revisões' apresenta uma barra de ferramentas com os botões 'Exportar' e 'Remover'. Abaixo, há uma tabela com as seguintes colunas: 'Ações', 'Revisão', 'Data' e 'Descrição'. Cada linha da tabela possui um ícone de olho na coluna 'Ações'.

Ações	Revisão	Data	Descrição
<input type="checkbox"/>	2	24/07/2019 09:35:23	versão 1
<input type="checkbox"/>	3	24/07/2019 09:35:32	versão 2
<input type="checkbox"/>	4	24/07/2019 09:35:37	versão 3

Figura 55. Lista de Revisões

Debug

Apresenta o item **Mídia** com as opções **Chat** e **Voz**, que permitem, respectivamente, testar as chamadas de *chat* e testar as scripts de URA e TTS das chamadas de voz.

Tela cheia

Expande a área do editor para melhor visualização do conteúdo.

Configurações do editor

Possibilita alterar algumas configurações do editor como o tema, tamanho da fonte e quebra de linha, conforme mostra a figura a seguir.



Figura 56. Configurações do editor

Ajuda

Apresenta um help online contendo exemplo de utilização das funções da LPU, conforme mostra a figura a seguir.



Figura 57. Ajuda

ÁRVORE DE DIRETÓRIOS E ARQUIVOS

A árvore mantém todos os arquivos e diretórios do serviço de URA.

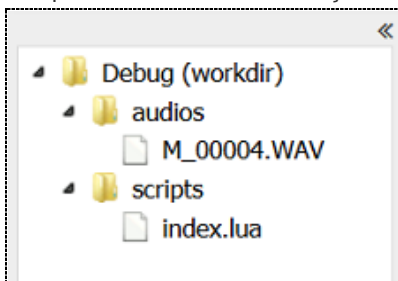


Figura 58. Árvore de diretórios e arquivos

Através do menu de contexto da árvore é possível criar, editar, renomear e remover arquivos e diretórios, comparar arquivos LUA, TXT e JSON do *workdir* com outras revisões e, além disso, testar os áudios existentes no serviço.

Os arquivos podem ser abertos pelo menu de contexto ou com um duplo clique. Para os arquivos de áudio, ao abri-los serão reproduzidos automaticamente no player existente no painel de debug.

ATENÇÃO

No navegador Internet Explorer não é possível a reprodução de arquivos WAV devido a sua limitação.

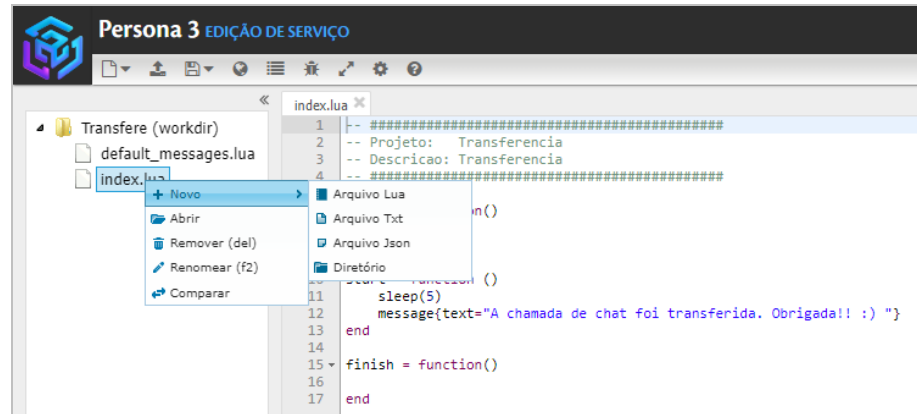


Figura 59. Menu

EDITOR

É no editor que será desenvolvido todo fluxo do serviço de URA. Para facilitar esse desenvolvimento existem algumas facilidades que auxiliam no trabalho como, por exemplo, menu autocompletar, que é apresentado automaticamente ao digitar algo ou através das teclas de atalho Ctrl + space. Além disso, o editor também acusa erros de sintaxe informando a linha com problema e a sua descrição.

Outra funcionalidade disponível é a reprodução de áudios em TTS. Para executar esta função basta selecionar o parâmetro tts junto com o texto e clicar no botão de Play do player de áudio, localizado no painel de debug. Caso o conteúdo do tts seja uma função,

será divulgado o nome da função com os parâmetros, ou seja, a função não será executada e o mesmo se aplica a uma variável, o áudio divulgado será o nome da variável.

ATENÇÃO

Para que esta ação esteja disponível, é necessário que o usuário tenha adquirido licenças de uso do servidor TTS da Dígitro.

Interface de Edição de
Serviços
CAPÍTULO 4

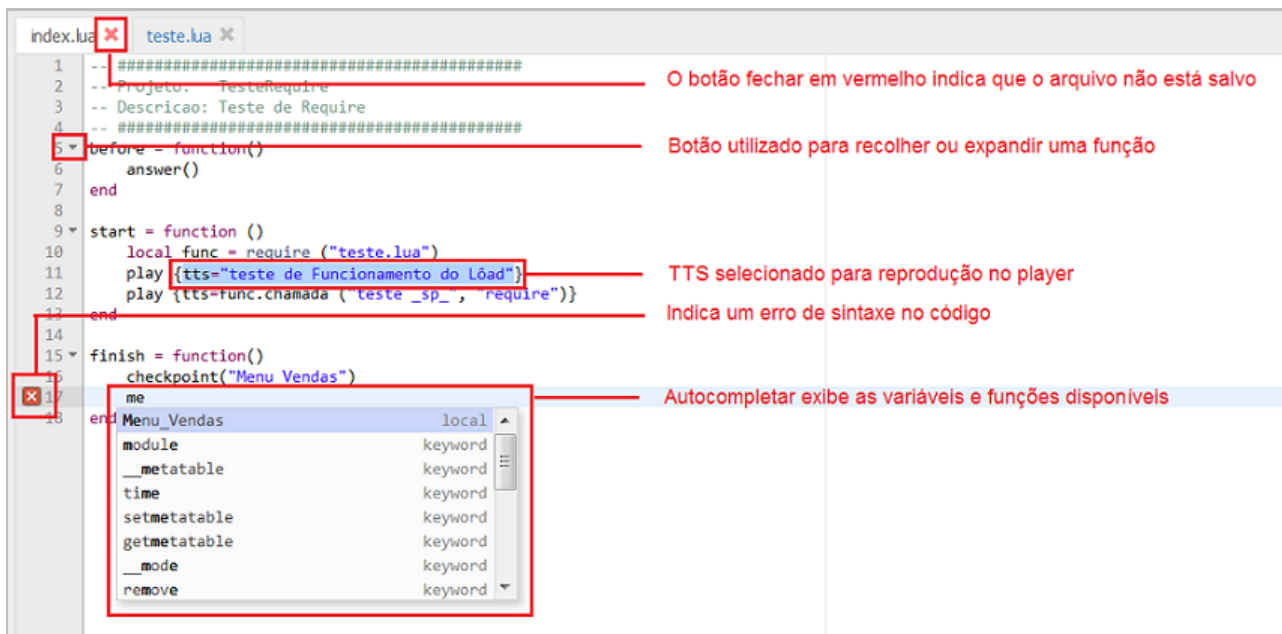


Figura 60. Editor

O editor dispõe de configurações que podem ser alteradas pelo usuário como tema, tamanho da fonte e quantidade de caracteres para quebra de linha.

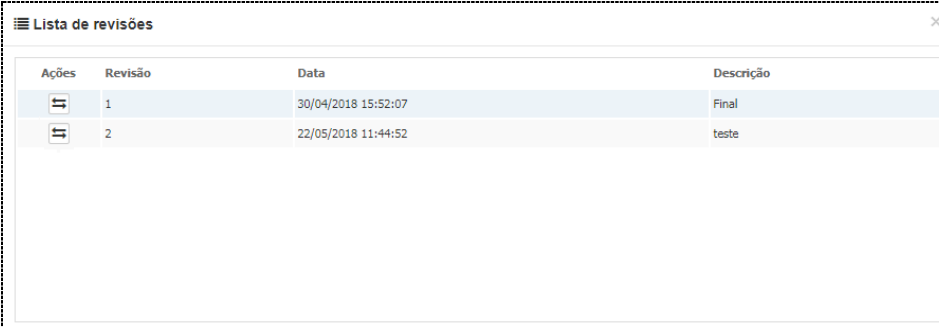
Para isso, clique no botão , é apresentada a Figura 56:

- Os temas estão divididos em **Claros** como Clouds e Eclipse e **Escuros** como Chaos e Terminal.
- O tamanho da fonte é 10 a 24 pixels.
- A quebra de linha tem os valores **Desabilitado** que não realiza a quebra, 40, 80 e 120 caracteres e **Livre** que utiliza toda área do editor antes de realizar a quebra.

Estas configurações são válidas apenas para a área de edição da URA.

COMPARAR REVISÕES

Para verificar a diferença entre as revisões, clique com o botão direito do mouse sobre o arquivo desejado e selecione a opção **Comparar** (Figura 59). Será apresentada a lista de revisões existentes.






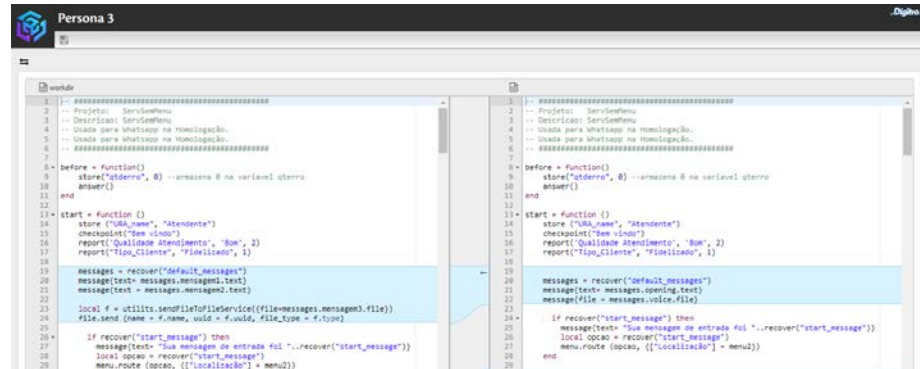
Ações	Revisão	Data	Descrição
	1	30/04/2018 15:52:07	Final
	2	22/05/2018 11:44:52	teste

Figura 61. Comparar revisões

Após clicar no botão **Comparar revisões**  da revisão desejada, é aberta uma nova janela permitindo a comparação dos arquivos.



```

1 |-----|
2 | -- Projeto: ServiMenu
3 | -- Descrição: ServiMenu
4 | -- Usada para whatsapp na homologação.
5 | -----|
6 | --|
7 |
8 | before = function()
9 |   store("idterra", 0) --armazena 0 na variavel idterra
10 |   answer()
11 | end
12 |
13 | start = function ()
14 |   store ("urla_name", "Atendente")
15 |   checkapi("sem video")
16 |   report("Qualidade Atendimento", "Boa", 2)
17 |   report("Tipo_Cliente", "Fidelidade", 1)
18 |
19 | messages = recover("default_messages")
20 | message(text= messages.msgagent.text)
21 | message(text = messages.msgagent2.text)
22 |
23 | local f = utils.sendFileToFileService(file=messages.msgagent3.file)
24 | file.send (name = f.name, user = f.userId, file_type = f.type)
25 |
26 | if recover("start_message") then
27 |   message(text= "Sua mensagem de entrada foi ".recover("start_message"))
28 |   local opcao = recover("start_message")
29 |   menu.route (opcao, [{"localizacao"} + menu2])
30 | end

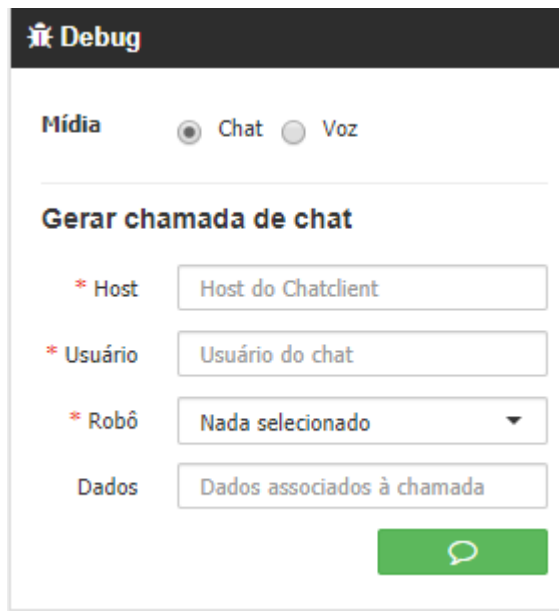
```

Figura 62. Comparação de arquivos

É possível comparar somente arquivos com a extensão **LUA**, **TXT**, ou **JSON**. A comparação de arquivos sempre será realizada entre a versão do arquivo existente no *workdir* e o mesmo arquivo existente em uma revisão publicada.

DEBUG

A interface do **Persona 3** permite ao usuário, com os privilégios específicos, realizar o debug de um serviço a sua escolha. A opção Debug, localizada na parte direita da interface de edição, apresenta o item Mídia com as opções *Chate Voz*.



The image shows a 'Debug' interface with a dark header containing a bug icon and the word 'Debug'. Below the header, there are two radio buttons for 'Mídia': 'Chat' (selected) and 'Voz'. A horizontal line separates this from the section 'Gerar chamada de chat'. Under this section, there are four fields: '* Host' with a text input containing 'Host do Chatclient'; '* Usuário' with a text input containing 'Usuário do chat'; '* Robô' with a dropdown menu showing 'Nada selecionado'; and 'Dados' with a text input containing 'Dados associados à chamada'. A green button with a speech bubble icon is located at the bottom right of the form.

Figura 63. Opção Debug

A opção *Chat* permite testar chamadas de *chat* realizando chamadas reais por meio do **ChatClient**.

Para gerar a chamada de *chat*, preencha os campos obrigatórios, marcados com asterisco. No campo Host, informe o servidor onde o **ChatClient** está instalado; no campo Usu-

ário, o nome do usuário que será exibido no *chat*, e no campo Robô, o robô configurado em uma rota do **Persona 3**.

O campo Dados, de preenchimento opcional, permite associar dados à chamada.

A seguir, clique no botão **Gerar chamada**  para iniciar o teste.

NOTA

- 1) *Outra forma de testar uma chamada de chat é por meio do menu Outros, localizado na página principal do **Persona 3**, opção Testar chamada de chat.*
- 2) *O robô utilizado deve estar cadastrado em uma rota de chat do **Persona**.*

A opção Voz, por sua vez, permite testar as scripts de URA e TTS das chamadas de voz. O Debug é realizado em um PABX virtual e um ambiente independente do processamento das scripts de URA, desta forma, esta funcionalidade não afeta o funcionamento das URAs reais.

Este Debug pode ser realizado de duas maneiras distintas:

- Debug de um serviço utilizando uma rota cadastrada no processo.
- Debug de um serviço sem a utilização de rota.

Caso o usuário opte por utilizar a primeira opção, deverá digitar no teclado virtual o número do grupo de roteamento desejado e, caso necessário, preencher os campos de filtro. Em seguida pressionar o botão **Iniciar debug**. Será feita uma verificação para identificar se o serviço de onde foi realizada a chamada é o mesmo que está realizando o atendimento da rota utilizada para atendimento. Caso não seja, uma mensagem de erro

será apresentada e a chamada finalizada. Quando estiver tudo certo com a rota e o serviço utilizado para gerar o debug, a simulação será iniciada. Caso não exista nenhuma rota cadastrada com as características utilizadas, uma mensagem de erro será apresentada e a chamada finalizada.

Caso o usuário opte por utilizar a segunda opção, basta pressionar o botão **Iniciar debug**. Desta forma o debug será realizado utilizando o serviço de onde a simulação foi iniciada e com o arquivo LUA aberto como principal.

Durante a simulação da chamada, em ambos os casos, as linhas que se encontram em execução aparecerão marcadas (highlight), para que o desenvolvedor consiga identificar em que parte da script o debug se encontra. A navegação poderá ser feita utilizando o teclado virtual e as mensagens serão reproduzidas para o usuário utilizando o player existente no painel de debug. As variáveis locais utilizadas no desenvolvimento da script de URA serão apresentadas no canto inferior direito da tela, em uma tabela que informará seu nome e valor.

Apenas as variáveis declaradas como local serão apresentadas na tabela de variáveis e valores. As variáveis não declaradas, consideradas globais no LUA, não são apresentadas.

Para evitar problemas de conflito, cada serviço poderá ser debugado por apenas um usuário de cada vez. Caso um segundo usuário tente realizar o debug, uma mensagem será apresentada na tela e será oferecida a opção de finalizar o debug que se encontra em execução, ou simplesmente aguardar que seja finalizado.

Durante a execução do debug, caso o usuário queira realizar uma pausa para analisar algum trecho da script, basta que ele pause a mensagem que está sendo reproduzida, desta forma, a chamada ficará congelada até que o botão de Play seja pressionado novamente. Vale ressaltar que, caso o usuário realize alguma alteração na script enquanto o debug estiver pausado, esta alteração não será executada quando o Play for pressionado, ela será interpretada apenas para a próxima chamada.

Mensagens que utilizam o parâmetro phrase, ou do tipo file hospedadas em um servidor de HTTP, não serão reproduzidas no simulador. Nestes casos a chamada irá prosseguir como se o áudio tivesse sido reproduzido, mas o usuário não irá ouvi-la.


ATENÇÃO

No Internet Explorer não é possível a reprodução de arquivos WAV devido à limitação do navegador e a chamada ficará congelada no ponto da reprodução. Por este motivo, não recomendamos a utilização desse navegador para simulação de chamadas.

O painel de debug possibilita simular uma chamada e navegar pelo serviço de URA em desenvolvimento.



Figura 64. Painel Debug

No painel de debug é possível configurar um filtro de origem e destino e assim testar o funcionamento de uma rota antes de colocá-la em operação. Ao clicar no botão 

(Configurar Filtros), é apresentada uma janela com os campos **Filtro de origem** e **Filtro de destino**, conforme mostra a figura a seguir.

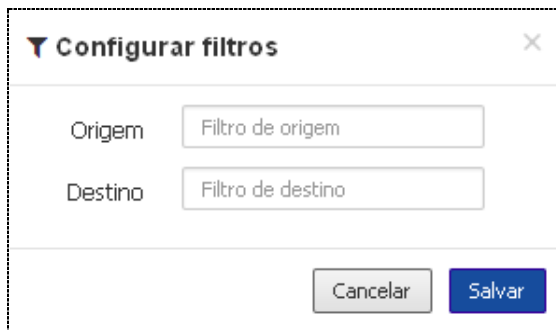


Figura 65. Configurar filtros

Na tabela de variáveis são apresentados os valores atribuídos a determinada variável no decorrer da execução do teste do serviço. Quando a variável recebe como valor um objeto, este é apresentado através de um popover com um clique no botão **Visualizar**.

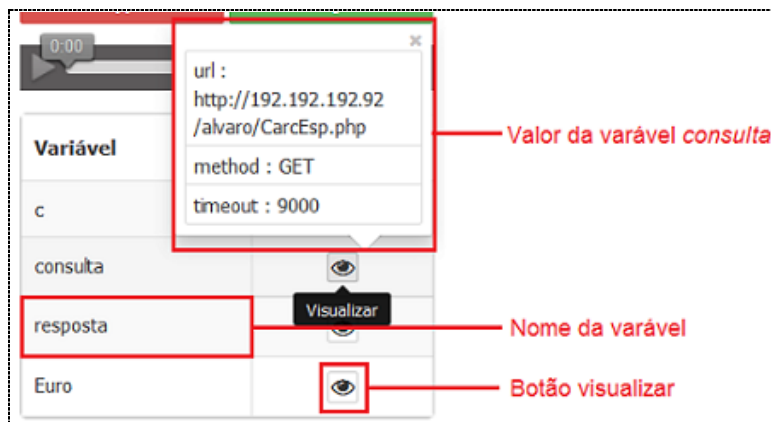


Figura 66. Valores atribuídos a variável

ATENÇÃO

Apenas as variáveis declaradas como local serão exibidas na tabela de variáveis e valores. As variáveis não declaradas, consideradas globais no lua, não são exibidas.

Quando um debug está em andamento o botão **Iniciar debug** fica desabilitado e a linha que está sendo executada no momento recebe um highlight (seleção na cor amarela), conforme mostra a figura a seguir.

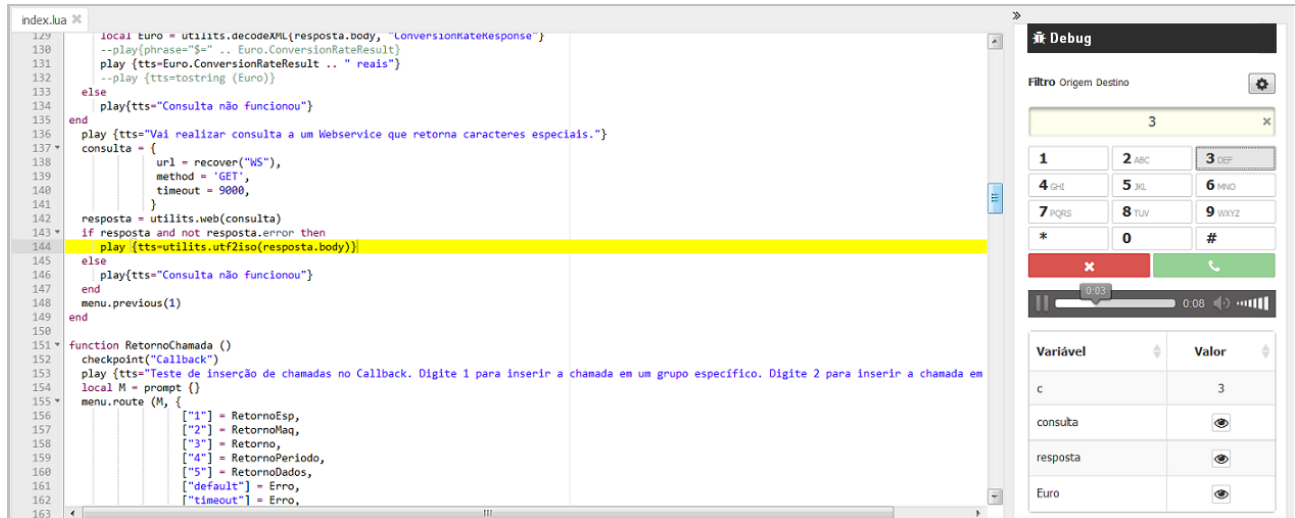


Figura 67. Debug em andamento

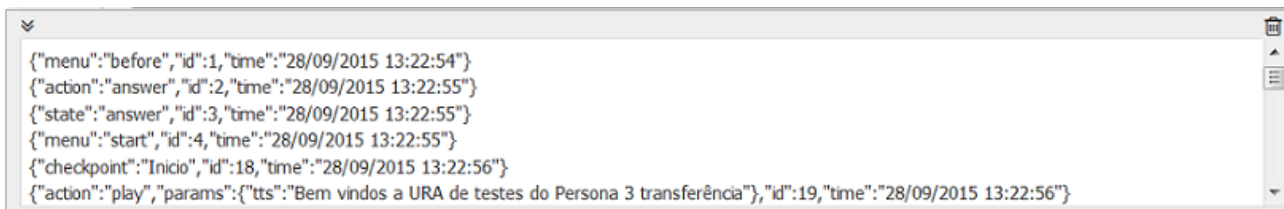
Para iniciar o debug de um serviço é necessário que o arquivo que contém a função **start** esteja aberto.

Se outro debug estiver em andamento na mesma rota, ao tentar iniciar o debug é apresentada uma mensagem informando o problema e permitindo que a outra execução seja encerrada.

Caso a execução retorne algum erro, este será apresentado em uma alerta no topo do editor.

CONSOLE

No console são apresentados todos os eventos recebidos na execução de debug.



```
{ "menu": "before", "id": 1, "time": "28/09/2015 13:22:54" }
{ "action": "answer", "id": 2, "time": "28/09/2015 13:22:55" }
{ "state": "answer", "id": 3, "time": "28/09/2015 13:22:55" }
{ "menu": "start", "id": 4, "time": "28/09/2015 13:22:55" }
{ "checkpoint": "Inicio", "id": 18, "time": "28/09/2015 13:22:56" }
{ "action": "play", "params": { "tts": "Bem vindos a URA de testes do Persona 3 transferência" }, "id": 19, "time": "28/09/2015 13:22:56" }
```

Figura 68. Console

5

SUPERVISÃO

O **Persona 3** fornece duas formas diferentes de supervisão: a supervisão de chamadas ativas e a supervisão das estatísticas das chamadas recebidas durante o dia. Estas opções podem ser acessadas através do menu **Supervisão**.

Para acessar a supervisão, o usuário deve possuir privilégios de supervisão. Por padrão, os perfis **Administrador Persona 3** e **Supervisor Persona 3** já possuem este privilégio.

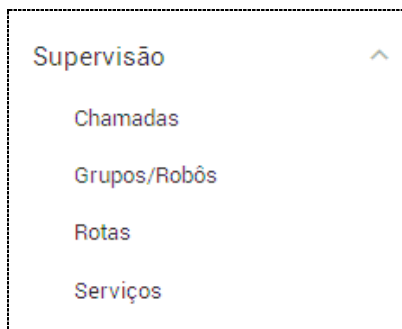





Figura 69. Menu Supervisão

SUPERVISÃO DE CHAMADAS

A interface de supervisão de chamadas possibilita o acompanhamento das 50 chamadas mais antigas que foram atendidas pelo serviço de URA. Caso existam licenças para mais um tipo de mídia (voz  e chat ) , será possível selecionar a mídia que se deseja visualizar, ou ambas ao mesmo tempo. Serão listadas as seguintes informações das chamadas:

Mídia

Mídia da chamada (*Chat*  ou *Voz* ).

Tipo

Indica se uma chamada é **Ativa** ou **Receptiva**.

ID

Identificador da chamada.

Estado

Estado em que a chamada encontra-se no momento.

Origem

Número que está realizando a chamada (número de A).

Destino

Número para o qual a chamada chegou na plataforma (DDR). Quando o Destino for um robô, será exibido, além do nome, o ícone do tipo de submídia deste (Telegram ou Facebook Messenger);

Grupo

Ponto de roteamento que está realizando o atendimento no **Persona 3**.

Rota

Conjunto de configurações que determina o atendimento da chamada no serviço de URA.

Serviço

Serviço de URA que atendeu a chamada.

Revisão

Revisão do serviço.

Menu



Função que está sendo executada dentro do serviço de URA.


Ação

Método com interação com o usuário que atuam como funções básicas utilizados no script de URA.

Duração

Tempo em que a chamada está ativa na plataforma.

Devido à quantidade de informações trafegadas, a interface de supervisão de chamadas não possui um *refresh* automático e, por isso, possui o botão **Recarregar**  para executar a atualização da tabela. O botão  fornece a opção de ocultar qualquer coluna da tabela diminuindo, desta forma, a quantidade de informações visíveis. O campo **Pesquisar** atua em todos os campos da tabela, inclusive os campos ocultos, facilitando a busca por uma chamada específica.

No lado superior direito da janela há o ícone , que apresenta as informações a seguir.

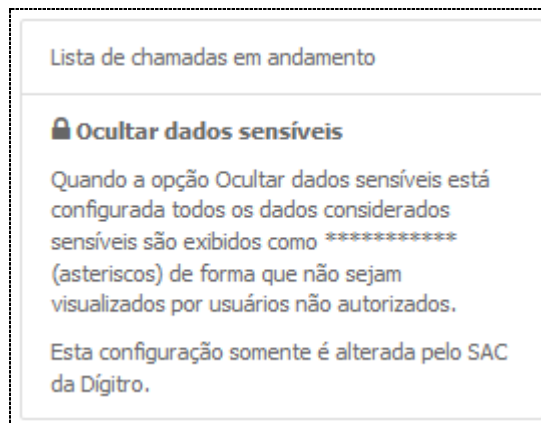


Figura 70. Ocultar dados sensíveis

Quando a opção **Ocultar dados sensíveis** estiver desabilitada, todas informações serão exibidas na interface sem nenhuma restrição e é exibido um ícone de cadeado aberto com o texto **Dados sensíveis visíveis**.

A lista de chamadas possui ordenação em todas as colunas.



The screenshot shows a web interface titled 'Supervisão de chamadas'. It has a search bar and a table with the following columns: 'Hidra', 'Tipo', 'ID', 'Estado', 'Origem', 'Destino', 'Grupo', 'Serviço', 'Menu', 'Ação', and 'Duração'. There are two rows of data in the table.

Hidra	Tipo	ID	Estado	Origem	Destino	Grupo	Serviço	Menu	Ação	Duração
	Ativa	b98c38e0-b44-11e9-af1b-09d71edc269d	end	Robo114	Georgia		-	-	-	00:00:00
	Ativa	b98c3790-b44-11e9-af1b-09d71edc269d	end	Robo114	johnath		-	-	-	00:00:00

Figura 71. Supervisão de Chamadas

Clicando em uma determinada chamada da tabela, é possível acompanhar todos os detalhes dela, desde sua entrada na plataforma, até sua finalização. Ao clicar em uma chamada é apresentada uma janela, conforme mostra a figura a seguir.

Call ID: a32aeb10-5405-11ea-a46f-4d0464d6fd89:0 (encerrada)

Dados sensíveis ocultados Exportar

#	Hora	Estado	Menu	Ação	Dados
2	20/02/2020 14:22:57	-	before	answer	-
3	20/02/2020 14:22:57	answer	before	-	-
4	20/02/2020 14:22:57	answer	start	-	-
5	20/02/2020 14:22:57	answer	start	message	text= teste
6	20/02/2020 14:22:58	answer	start	-	-
7	20/02/2020 14:22:58	answer	start	message	text= teste222
8	20/02/2020 14:22:58	answer	start	sms	*****
9	20/02/2020 14:22:58	answer	start	exception	COROUTINE.RESUMED [message:][false]/axs/bin/uraengine /mod_session.lua:955: attempt to call method 'sms' (a nil value)]
10	20/02/2020 14:22:58	answer	start	drop	-
11	20/02/2020 14:22:58	answer	finish	-	-
12	20/02/2020 - - - -	end	-	-	-

Figura 72. Detalhamento da chamada

Os campos disponíveis são:

Hora

Data e hora completa em que o evento ocorreu.

Estado

Estado em que a chamada encontra-se no momento.

Menu

Função que está sendo executada dentro do serviço de URA.

Ação

Métodos com interação com o usuário que atuam como funções básicas utilizados na especificação dos serviços de URA.

Dados

Parâmetros executados pela ação.

Os registros são atualizados automaticamente à medida que vão ocorrendo e, após sua finalização, a chamada fica disponível para consulta por mais 30 segundos. Após este período, os dados são descartados.



No detalhamento de uma chamada é possível exportar todos os registros através do botão **Exportar**.



Figura 73. Botão Exportar

A exportação dos registros pode ser realizada nos formatos PDF, JSON, XML, TXT, MS-Word e MS-Excel.

SUPERVISÃO DE ESTATÍSTICAS

A Supervisão de Estatísticas é responsável por mostrar as informações recentes (do próprio dia) das chamadas recebidas. Caso existam licenças para mais de um tipo de mídia (*Chat*  e *Voz* ) , será possível selecionar a mídia que se deseja visualizar, ou ambas ao mesmo tempo. Ao selecionar ambas as mídias, serão mostradas as informações das mídias de *chat* e *voz* ao mesmo tempo, porém em contadores separados por mídia.

A Supervisão é dividida em **grupos/robôs**, **rotas** e **serviços**, e possibilita o acompanhamento dos dados de ocupação de licenças de pontos de vista diferentes. Estas interfaces de supervisão apresentam além das informações específicas referentes à visão escolhida, as seguintes informações: **Histórico de ocupação**, **Ocupação atual**, **Pico de**

ocupação e Estatísticas de chamadas. Estas informações são globais, independente da visão escolhida.

Gráfico de histórico de ocupação

Apresenta os picos de ocupação da plataforma referentes à data atual para cada mídia selecionada. Em todos os pontos do gráfico é possível saber o horário exato do pico e o percentual ocupado. A atualização dos dados do gráfico ocorre a cada 5 minutos, sendo registrada a hora exata quando ocorreu o pico de ocupação dentro dos 5 minutos de análise. Deste modo, o intervalo entre dois pontos pode ser de até 10 minutos. Como trata de informação histórica, o último ponto plotado no gráfico poderá não refletir a ocupação atual do sistema.

Barra de ocupação atual

Apresenta o percentual de ocupação do total de licenças em função da quantidade de chamadas em andamento.

Pico de ocupação:

Apresenta o horário, quantidade de chamadas e percentual de ocupação do maior volume de chamadas simultâneas ocorridas na plataforma.

Estatísticas de chamadas

Mostra o total de chamadas recebidas, tempo médio das chamadas e o percentual de tempo ocupado. O tempo médio é contabilizado somente após finalização da chamada.

Este campo apresenta, também, as informações de quantidade de chamadas não atendidas por falta de licença ou por falta de rota cadastrada. As chamadas sem rota são classificadas em: Rota Bloqueada, Limite de Tráfego, Fora do Horário, Destino Inválido (somente para mídia de voz), Origem Inválida (somente para mídia de voz) e Erro de Script. Estas informações são apresentadas nas estatísticas gerais, e também na supervisão de cada grupo.

Uma chamada será contabilizada em **Erro de Script** apenas quando ocorrer um erro na carga da *script* de URA, caso ocorra um erro em tempo de execução, ou seja, durante a navegação de um usuário, ela não será contabilizada neste campo.

As chamadas contabilizadas no campo **Sem rota** também são contabilizadas no contador geral de chamadas, visto que, no pequeno intervalo de tempo em que o **Persona 3** analisa se deve ou não realizar o atendimento, uma licença está sendo consumida. Chamadas contabilizadas no campo **Sem licença** não são contabilizadas no contador geral de chamadas. Para mais detalhes posicione o mouse no valor correspondente, conforme mostra a figura a seguir.

Motivo	Quantidade
Rota Bloqueada	3
Limite de Tráfego	0
Fora do Horário	0
Destino Inválido	0
Origem Inválida	0
Erro de Script	0

Figura 74. Detalhes Sem rota voz

Motivo	Quantidade
Rota Bloqueada	0
Limite de Tráfego	0
Fora do Horário	0
Erro de Script	0

Figura 75. Detalhes Sem rota chat

Gráfico de estatísticas

Apresenta o nome e percentual de ocupação de até 10 grupos/robôs, rotas ou serviços com maior número de chamadas em atendimento naquele momento. No caso de serviços que fazem voz e *chat* ao mesmo tempo, serão contabilizadas ambas as mídias

Além das informações globais, é apresentada uma listagem com o total de chamadas, tempo médio, horário de pico, quantidade de chamadas no pico e ocupação atual de acordo com a visão escolhida: grupo/robôs, rota ou serviço. No caso de serviço, se tiver licenças para as mídias de *chat* e voz, as informações de ambas as mídias serão contabilizadas junto, e se o usuário desejar visualizar os contadores individualmente basta selecionar a mídia desejada.

As estatísticas apresentadas são reiniciadas de acordo com a configuração de Horário de Reinício das Estatísticas, sendo por padrão às 0h. Através do botão **Reiniciar**, é possível reiniciar todas as informações da interface, exceto o gráfico de histórico de ocupação. Serão reiniciadas as informações de **todas as mídias**.

NOTA

*Chamadas encaminhadas para um ponto de roteamento supervisionada pelo **Persona 3**, mas que não possuam uma rota válida para atendimento, serão contabilizadas no total de chamadas e no total do grupo, mas não serão utilizadas para a contabilização do tempo médio de atendimento.*

*Nos casos de chamadas encaminhadas para um ponto de roteamento monitorado pelo **Persona 3**, mas que não existam mais licenças para atendimento, não serão contabilizadas para as estatísticas.*

SUPERVISÃO DE GRUPOS

A supervisão de grupos apresenta os dados de ocupação dos Grupos configurados nas rotas.

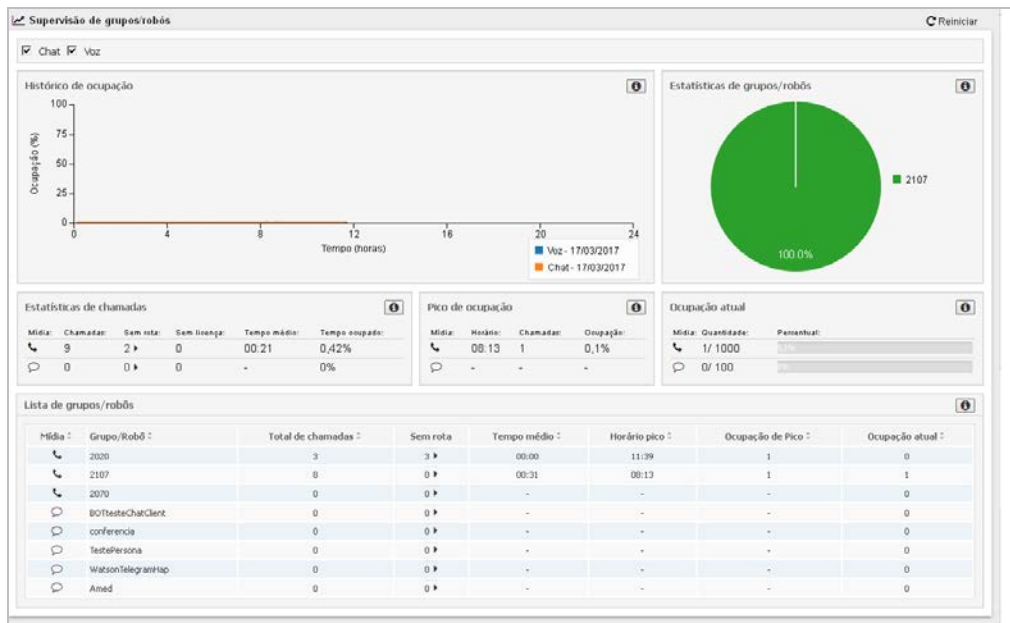


Figura 76. Supervisão de Grupos/Robôs

Na tabela **Lista de Grupos/Robôs** é possível visualizar o detalhamento das falhas ocorridas em determinado grupo. Para isso posicione o cursor do mouse sobre o valor da coluna **Sem rota**.

Motivo	Quantidade
Rota Bloqueada	3
Limite de Tráfego	0
Fora do Horário	0
Destino Inválido	0
Origem Inválida	0
Erro de Script	0

Figura 77. Detalhamento das falhas

SUPERVISÃO DE ROTAS

A supervisão de rotas apresenta os dados de ocupação das Rotas configuradas no sistema.

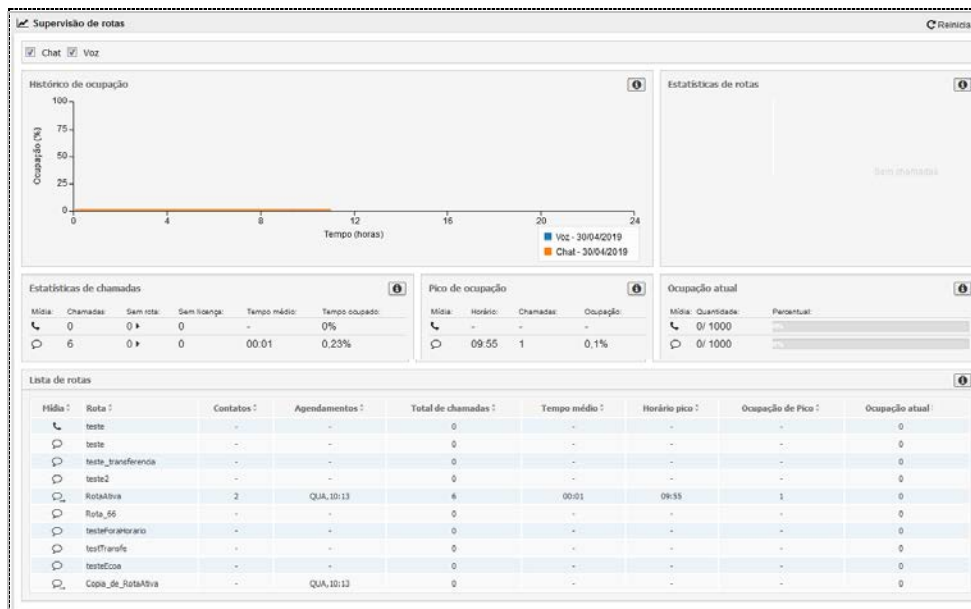


Figura 78. Supervisão de Rotas

SUPERVISÃO DE SERVIÇOS

A supervisão de serviços apresenta os dados de ocupação dos Serviços configurados no sistema.

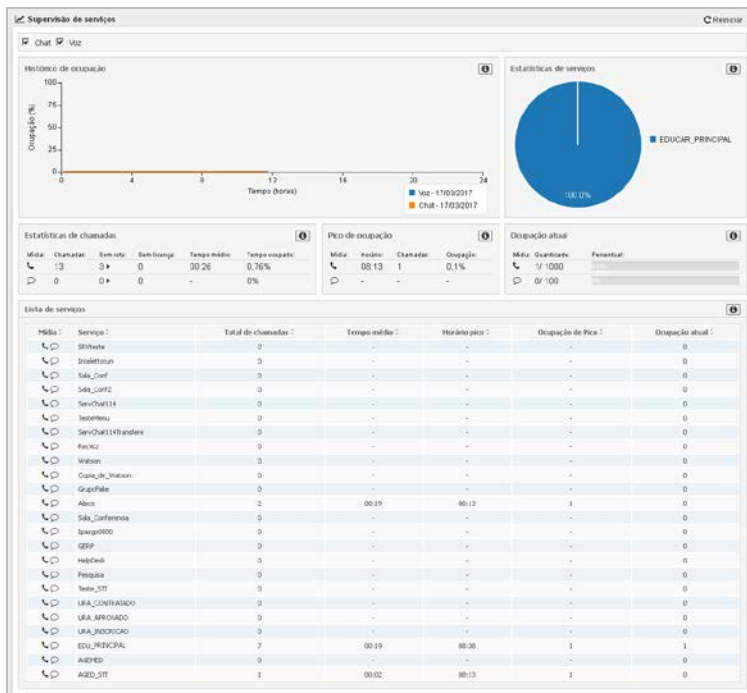


Figura 79. Supervisão de Serviços

6

CONFIGURAÇÕES

A interface de configuração possibilita gerenciar as configurações gerais da interface. São necessários privilégios específicos para acessar as interfaces de configuração. Os idiomas disponíveis atualmente são Português e Espanhol.

CONFIGURAÇÕES GERAIS

No menu **Configurações**, selecione a opção **Configurações Gerais**, é apresentada a janela a seguir.



The screenshot shows a window titled "Configurações gerais" with a close button (X) in the top right corner. Below the title bar, there is a section labeled "* Reinício das estatísticas" with a time input field containing "0:00". Below this, there is a section labeled "* Campos obrigatórios" with two buttons: "Cancelar" and "Salvar".

Figura 80. Configurações Gerais

7

RELATÓRIOS

O menu **Relatórios** será apresentado apenas aos usuários que possuem permissão de acesso ao módulo de relatórios do **Persona 3**. Por padrão, os perfis **Administrador Persona 3** e **Supervisor Persona 3** possuem este privilégio. Esse menu permite a emissão de relatórios históricos, utilizando modelos prontos e informações do banco de dados relacional da plataforma.

Clicando no menu **Relatórios** é acessada a **Interface Web para Emissão de Relatórios** das plataformas Dígito, por meio da qual o usuário poderá emitir relatórios imediatamente ou agendá-los para emissão posterior. Por ser um módulo comum a outros aplicativos, ele é descrito separadamente manual Interface Web para Emissão de Relatórios.

Para mais informações sobre os modelos de relatórios do **Persona 3**, consulte o Anexo A – Emissão de Relatórios **Persona 3**.

8

LOG DE OPERAÇÕES

O menu **Log de Operações** acessa a interface **Log de Operações**, na qual o usuário poderá pesquisar os seguintes registros de operações realizadas no sistema:

- Login e logout.
- Configuração de Alarmes, Dispositivos, URA.
- Cadastro de Domínio, Usuário, Time, Motivo de pausa, URA DXML, Classificação de chamadas e Classificação inicial.
- Supervisão das ações sobre o usuário.
- Relatórios de login, agendamento, emissão e exclusão.

Por ser comum a outros aplicativos, esse módulo será descrito separadamente no manual **Log de Operações**.

NOTA

*A funcionalidade **Log de Operações** é opcional e desabilitada por padrão. Para habilitar/desabilitar essa função, entre em contato com o Serviço de Suporte ao Cliente da Dígitro.*

O menu Log de Operações acessa a interface Log de Operações, onde o usuário pode pesquisar registros de alterações e cadastros realizados na interface do Persona 3, ou ainda, visualizar os registros das operações efetuadas em sistemas que contenham dados sensíveis. O registro deste tipo de informação visa adequar o produto à Lei Geral de Proteção de Dados (LGPD).

Consulta das operações logadas

Parâmetros

Aplicação: PERSONA Operação: Todas Operações

Usuário: Data Inicial:

Endereço IP: Data Final:

Filtros por Campo/Valor

Nome do campo: (nenhum registro encontrado) Valor do campo:

Adicionar filtro

Nome do usuário	Endereço do usuário	Aplicação	Nome da Operação	Data da operação	Detalhes
sacdgt	172.16.50.89	PERSONA	Deletar arquivo ou diretório	20/11/20 15:01:52	Serviço teste - Arquivo: levantamento_hostnames.txt
sacdgt	172.16.50.89	PERSONA	Enviar arquivo	20/11/20 15:01:41	Serviço teste - Arquivo: levantamento_hostnames.txt
sacdgt	172.16.50.89	PERSONA	Salvar arquivo	20/11/20 14:18:32	Serviço teste - Arquivo: testeNewDir/teste11.lua
sacdgt	172.16.50.89	PERSONA	Mostrar arquivo ou diretório	20/11/20 14:18:24	Serviço teste - Arquivo: teste11.lua, Diretório: testeNewDir
sacdgt	172.16.50.89	PERSONA	Deletar arquivo ou diretório	20/11/20 14:18:14	Serviço teste - Arquivo: testeNewDir/testeNew.lua
sacdgt	172.16.50.89	PERSONA	Salvar arquivo	20/11/20 14:17:17	Serviço teste - Arquivo: teste11.lua
sacdgt	172.16.50.89	PERSONA	Salvar arquivo	20/11/20 14:17:17	Serviço teste - Arquivo: testeNewDir/testeNew.lua
sacdgt	172.16.50.89	PERSONA	Salvar arquivo	20/11/20 14:09:32	Serviço teste - Arquivo: testeNewDir/testeNew.lua
sacdgt	172.16.50.89	PERSONA	Salvar arquivo	20/11/20 14:09:16	Serviço teste - Arquivo: testeNewDir/testeNew.lua
sacdgt	172.16.50.89	PERSONA	Salvar arquivo	20/11/20 14:05:16	Serviço teste - Arquivo: teste11.lua

Itens 1 a 10 (10/20)

Figura 81. Log de Operações para LGPD

9

OUTROS

LICENÇAS

No menu LICENÇAS é possível verificar as licenças do **Persona 3** que estão sendo utilizadas em determinado momento.

O acesso simultâneo à interface é limitado ao número de licenças adquirido. A licença é automaticamente liberada quando o usuário sair da aplicação através do menu do usuário. Se o usuário fechar a aplicação sem utilizar o menu **Sair**, a licença ficará ativa por 15 minutos, para então ser liberada pelo sistema.



The screenshot shows a window titled 'Licenças' with a sub-header 'Lista de licenças em uso'. It contains a table with five columns: 'Ações', 'Nome', 'Endereço', 'Data de login', and 'Tempo inativo'. Each row represents an active license for the user 'root'.

Ações	Nome	Endereço	Data de login	Tempo inativo
	root	192.192.192.19	15/03/2017 10:24:30	00:00:47
	root	192.192.192.92	15/03/2017 10:25:04	00:06:33
	root	192.192.192.19, 192.192.192.19	15/03/2017 10:25:08	00:00:19
	root	192.192.192.92	15/03/2017 10:58:09	00:00:00
	root	192.192.192.19	15/03/2017 10:58:09	00:06:18

Figura 82. Licenças

A interface de licenças apresenta a listagem das licenças ocupadas atualmente. Na listagem são apresentados: nome do usuário, endereço IP, data de login e tempo em que o usuário está inativo na aplicação.

Para acessar, o usuário precisa ter privilégios de consulta de licenças. Por padrão, os perfis **Administrador Persona 3** e **Supervisor Persona 3** já possuem estes privilégios.

SOBRE

Apresenta informações do pacote e release do **Persona 3** instalado.

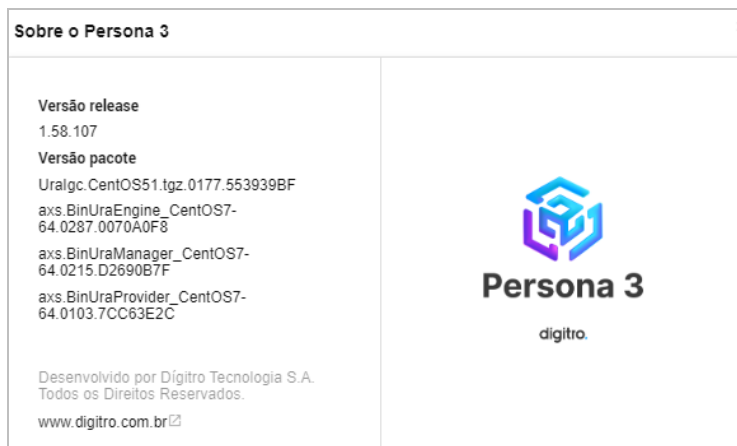


Figura 83. Sobre

TESTAR CHAMADA DE CHAT

No menu Outros > TESTAR CHAMADA DE CHAT é possível gerar uma chamada de *chat* através do **ChatClient**.

A imagem mostra uma janela de diálogo intitulada "Testar chamada de chat" com um ícone de fechar no canto superior direito. O formulário contém os seguintes campos:

- * Host: Campo de texto com o valor "Host do Chatclient".
- * Usuário: Campo de texto com o valor "Usuário do chat".
- * Robô: Menu suspenso com o valor "Nada selecionado".
- Dados: Campo de texto com o valor "Dados associados à chamada".

Na base da janela, há uma legenda "* Campos obrigatórios" e dois botões: "Cancelar" (cinza) e "Iniciar chat" (azul).

Figura 84. Testar chamada de chat

PROCEDIMENTO

Testar chamada de *chat* pelo menu Outros

1. Na página inicial do **Persona 3**, acesse o menu **Outros** > **Testar chamada de chat**. Será aberta a página Testar chamada de chat (Figura 84).
2. Preencha os campos obrigatórios, marcados com asterisco.

3. *No campo Host, insira o servidor onde o **ChatClient** está instalado.*
4. *No campo Usuário, insira o nome de usuário exibido no chat.*
5. *No campo Robô, indique o robô configurado em uma rota.*
6. *O campo Dados, de preenchimento opcional, permite associar dados à chamada de chat.*
7. *Clique no botão **Iniciar chat**, para abrir a interface do **ChatClient**, onde a chamada será testada.*

NOTA

*O robô utilizado deve estar cadastrado em uma rota do chat do **Persona**.*

10

LINGUAGEM DE PROGRAMAÇÃO DE URA

A LPU (Linguagem de Programação de URA) consiste de uma DSL (*Domain-Specific Language*) baseada na linguagem LUA e voltada ao desenvolvimento de serviços de URA.

Embora utilize o interpretador LUA, parte das bibliotecas nativas da linguagem não está disponível. Em contrapartida, foi adicionada uma biblioteca que disponibiliza o acesso aos recursos da plataforma Telecom, constituindo seu domínio e sendo, portanto, referenciada como biblioteca de domínio.

Em termos de sintaxe, trata-se de LUA, mas seu domínio des caracteriza-a como linguagem de propósito geral, induzindo a uma filosofia de programação em menus, pertinente ao domínio de serviço de URA.

A LPU está estruturada em três pilares fundamentais:

- Linguagem LUA;
- Programação de menus;
- Biblioteca de domínio.

Os pilares da LPU são descritos a seguir.

LINGUAGEM DE LUA

Algumas bibliotecas/funções nativas estão disponíveis no contexto de execução da LPU. São elas: *string*, *table*, *tonumber*, *tostring*, *date*, *time*, *ipairs*, *pairs*, *next* e *type*.

Em adição, a LPU disponibiliza sobre a linguagem LUA dois importantes conceitos:

- **Abstração de contexto:** faz com que operações assíncronas (não atômicas) pareçam síncronas.
- **Abstração de instância:** esconde do desenvolvedor o problema de múltiplas chamadas.

Em resumo, a LPU programa simplesmente a lógica do diálogo. Cada chamada executa uma instância de código independente e algumas operações de duração não instantânea (como reproduções de áudio) comportam-se como instantâneas do ponto de vista da programação.

ATENÇÃO

Este manual não descreve a linguagem LUA. Para que a linguagem de URA descrita no manual seja plenamente compreendida, o desenvolvedor deve ter o conhecimento prévio da linguagem LUA. O manual em português desta linguagem pode ser encontrado em <http://www.lua.org/manual/5.1/pt/manual.html>

PROGRAMAÇÃO DE MENUS

Menus são blocos de diálogo implementados como funções LUA. Tais funções não recebem parâmetros nem retornam valores. Também não podem chamar diretamente outras funções (menus). Para isto, há diretivas de navegação.

BIBLIOTECA DE DOMÍNIO

É o conjunto de funcionalidades disponíveis na LPU através da API de domínio.

Dividem-se em três grupos:

- Métodos diretos.
- Auxiliares de navegação.
- Utilidades gerais.
- Métodos Especiais.
- Funções Úteis.

MENU

Existem três funções pré-definidas as quais o *engine* de URA utiliza como baliza durante a execução da *script*.

Estas funções são: ***before***, ***start*** e ***finish***.

A função *before* é executada antes do atendimento da chamada, promovendo a escolha do usuário para realizar a ação de atendimento ou desligamento da chamada mediante um tratamento definido pelo desenvolvedor.

Após o atendimento da chamada a função *start* passa a ser executada. Esta função é a única obrigatória numa *script* de URA e tem o seu escopo completo com as primitivas de URA, diferente das funções *before* e *finish* que possuem escopos reduzidos.

Ao desligar uma chamada a função *finish* é executada permitindo ao desenvolvedor realizar alguma tratativa mesmo após o desligamento, tal como, escrever o tempo da chamada da URA em um banco.

Além destas funções de URA, a *script* pode conter outras funções criadas pelo próprio usuário, que trata o plano do negócio do cliente, como por exemplo: verifica um número digitado, trata um retorno de consulta a um *web service*. Estas funções podem ser criadas sem problemas, a diferença é de como é realizado o acesso.

Buscando um maior controle e mapeamento por parte do *engine* de URA, foi criado o *container* de funções menu. Estes são utilizados para acessar as funções criadas pelo usuário. Vide mais detalhes no item Menu da API.

API

São as primitivas utilizadas no desenvolvimento das *scripts* de URA, as quais abstraem todo o contexto de telefonia envolvida.

Elas podem ser classificadas em métodos diretos, auxiliar de navegação e *utils*.

MÉTODOS DIRETOS

São os métodos com interação com o usuário e atuam com o funções básicas utilizados na especificação dos serviços de URA.

Estes bloqueiam a execução do *script* até que a ação em andamento termine de processar.

Message

Função utilizada para reproduzir os tipo de áudio no diálogo em chamadas de voz, ou enviar mensagens de texto em chamadas de chat. A parametrização de cada tipo de áudio é descrito a seguir.

- profile:
 - **file** - reprodução de arquivos pré-gravados e armazenados localmente ou remotamente para chamadas de voz. As mensagens WAV/MP3/DAT, neste profile file, quando reproduzidas pela LPU devem seguir um padrão de nome M_XXXXX.ext onde X é o número e ext é uma das extensões compatíveis.

Exemplo:

```
message{file="/axs/m sgs/M_00004.WAV"}
```

```
message{file="http://192.168.0.1/URA_Atendimento/M_00123.WAV"}
```

- **text** – envio de mensagem de chat, para chamadas de chat, e reprodução Dinâmica via tecnologia Text-to-Speech para chamadas de voz, quando não for passado nenhum valor para o campo tts, file ou phrase.

Exemplo:

```
message{text="Bem vindo ao atendimento automático de chat"}
```

```
message{text="O valor do seu saldo é de "..saldo"}
```

- **menu** - este parâmetro é utilizado para apresentar as opções de menu de forma mais amigável em URAs de atendimento de chat. Ao utilizá-lo, as opções de menu podem ser apresentadas em formas de botões ou de listas. Quando o usuário seleciona uma dessas opções texto do botão/lista é enviado para a URA.

Observação: a exibição pode variar conforme a aplicação utilizada pelo interlocutor. Além disso, podem ser configuradas outras duas funcionalidades

para os botões. Uma como um atalho para o envio do número de contato. Este parâmetro será utilizado apenas nas chamadas de chat.

- **Quando o interlocutor for WhatsApp**

- Se o modo de exibição escolhido for botão:

- O nome de cada menu deve ter no máximo 20 caracteres ;
- São aceitos no máximo 3 itens.

- Se o modo de exibição escolhido for lista:

- O título deve ter no máximo 20 caracteres;
- O nome de cada menu deve ter no máximo 24 caracteres;
- São aceitos no máximo 10 itens.

- O modo de exibição é configurado no parâmetro `menu_settings`.

Caso as limitações não sejam respeitadas, o envio da mensagem não será aceito.

Exemplo:

```
Message {text='Para compras, digite 1. Para vendas, digite 2. Para RH, digite 3.', menu = {'1', '2', '3'}}
```

```
message {text='Para compras, digite 1. Para vendas, digite 2. Para RH, digite 3. Para outros, digite 4', menu = {'1', '2', {'3', '4'}} }
```

```
message {text='Por favor, informe a sua localização ou o número do seu contato', menu = {{text='localização', request_location = true}, {text='Contato', request_contact=true}}}
```

menu_settings – este parâmetro é utilizado para configurar o modo de exibição do menu no WhatsApp, podendo ser configurado como: lista ou botão. Caso este parâmetro não seja informado, o formato será ajustado automaticamente conforme a quantidade de itens do menu: até 3 itens serão exibidos como botão e acima disso serão exibidos como lista. Quando o

modo de exibição for configurado como lista é possível configurar o tipo de lista.

Exemplo:

```
message {text='Para compras, digite 1.
```

```
Para vendas, digite 2. Para RH, digite 3.', menu = {'1', '2', '3'}, menu_settings = { type='list', title='Opções' }
```

```
message {text='Para compras, digite 1.
```

```
Para vendas, digite 2. Para RH, digite 3.', menu = {'1', '2', '3'}, menu_settings = { type='button' }
```

- **tts** – reprodução Dinâmica via tecnologia Text-to-Speech.

Exemplo:

```
message{tts="Teste de URA"}
```

```
message{tts="O valor do seu saldo é de "..saldo}
```

Não é permitido o uso de quebra de linhas neste profile do message.

- **speaker** - Este parâmetro deverá ser utilizado junto ao TTS dinâmico. Nele é possível selecionar qual voz será utilizada para sintetizar o áudio. Atualmente possuímos duas vozes, o Marcos e a Ana Paula. Para que seja possível utilizar esta funcionalidade é necessário que o servidor que sintetiza as mensagens possua as duas vozes. Caso este parâmetro não seja informado, será utilizado por padrão a voz Ana Paula.

Exemplo:

```
message {text="Bem Vindo a esta URA!", speaker="marcos"}
```

```
message {text="Bem Vindo a esta URA!", speaker="anapaula"}
```

- **phrase** – reprodução de mensagens pré-gravadas pela Dígítro. Para estas mensagens existe a necessidade de algumas parametrizações, conforme descrito a seguir:

Exemplo:

message{phrase="DASHHMM=09:30|ASHHMM=19:30"} - Divulgação "Das nove horas e trinta minutos as dezenove horas e trinta minutos".

DATE - 01/01, 0101, 01/01/2015, 01012015, TODAY - Divulgação "Primeiro de Janeiro" ou "Primeiro de Janeiro de dois mil e quinze" ou "<data atual do sistema DD/MM/AAAA>".

DDMM - 01/01, 0101, TODAY - Divulgação "Primeiro de Janeiro" ou "<data atual do sistema DD/MM>".

DDMMAAAA - 01/01/2015, 01012015, TODAY - Divulgação "Primeiro de Janeiro de dois mil e quinze" ou "<data atual do sistema DD/MM/AAAA>".

HOUR - 01:01, 0101, 01:01:01, 010101, NOW - Divulgação "uma hora e um minuto" ou "uma hora e um minuto e um segundo" ou "<hora atual do sistema HH/MM/SS>".

HHMM - 01:01, 0101, NOW - Divulgação "uma hora e um minuto" ou "<hora atual do sistema HH/MM>".

HHMMSS - 01:01:01, 010101, NOW - Divulgação "uma hora e um minuto e um segundo" ou "<hora atual do sistema HH/MM/SS>".

ASHHMM - 01:01, 0101, NOW - Divulgação "a uma hora e um minuto" ou "<hora atual do sistema HH/MM>".

ASHHMMSS - 01:01:01, 010101, NOW - Divulgação "a uma hora e um minuto e um segundo" ou "<hora atual do sistema HH/MM/SS>".

DASHHMM - 01:01, 0101, NOW - Divulgação "da uma hora e um minuto" ou "<hora atual do sistema HH/MM>".

DASHHMMSS - 01:01:01, 010101, NOW - Divulgação "da uma hora e um minuto e um segundo" ou "<hora atual do sistema HH/MM/SS>".

\$ - 12,32 ou 12.32 - Divulgação "doze reais e trinta de dois centavos".

NUMBER - 123 456 789 máximo de 11 números. - Divulgação "cento e vinte três milhões, quatrocentos e cinquenta e seis mil, setecentos e oitenta e nove".

ONEBYONE - 1234567890 máximo de 20 números. - Divulgação "um, dois, três, quatro, cinco, seis, sete, oito, nove, zero".

MSG - 4 ou 4,4 - Divulgação "Digipro tecnologia" - "Digipro tecnologia, digipro tecnologia".

CPF - 123.345.567-43 ou 12.345.678/9301-23 - Divulgação "um dois três, três quatro cinco, cinco seis sete, quatro três" e "um dois, três quatro cinco, seis sete oito, nove três zero um, dois três".

- **params** – este parâmetro é utilizado em conjunto com frase e serve para divulgar mensagens pré-gravadas com parâmetros.

Exemplo:

message{phrase="5051", params="NUMBER=345{\$=20"} Divulgação "Seu pedido gerado possui o número trezentos e quarenta e cinco com o valor total de vinte reais."

- **cutthrough** ou **cut** – divulgação é finalizada quando usuário digitar qualquer cifra. Para isso o parâmetro deve ser atribuído com true.

Exemplo:

play{file="http://192.168.0.1/URA_Atendimento/M_00124.WAV" cut="true"}

- **contínuos** - Este parâmetro é utilizado para realizar uma reprodução contínua, não bloqueante. isto permite que a divulgação de uma mensagem durante algum processamento da URA, por exemplo uma consulta web que demore para responder. A reprodução é automaticamente cancelada no próximo play. Seu valor padrão é false.

Exemplo:

```
play{file="http://192.168.0.1/URA_Atendimento/M_00124.WAV" continuous="true"}
```

- **type** - Este parâmetro é utilizado para definir se as mensagens de *chat* devem também ser encaminhadas como arquivo de áudio. Para isso, a mensagem é enviada para o servidor de TTS (Text to Speech) da Dígitro onde é convertido para áudio. Este áudio é encaminhado para que o usuário não precise ler os textos.

Este campo pode ser preenchido com dois valores:

- **audio**: com este valor setado serão encaminhados apenas os arquivos de áudio e não as mensagens de texto.
- **both**: com este valor setado serão encaminhadas tanto as mensagens de texto quanto as de áudios.

Para enviar apenas o texto basta omitir este parâmetro.

Exemplo:

```
message {text="Esta é a mensagem 1", type="audio"}
```

```
message {text="Como posso ajudá-lo?", menu = recover("menu"), speaker = "marcos", type="both"}
```

Message não possui retorno.

Não pode ser utilizado dentro das funções pré-definidas *before* e *finish*.

hsm.getFidelization

Função utilizada para obter uma lista de contatos ou os detalhes de um contato fidelizado em uma mensagem HSM.

- Profile:
- **tokenapi** - Chave de acesso a API gerada no cadastro da submídia.

- Opcionais:
- **destination** - Utilizado para filtrar por um número de telefone específico, ou a parte de um número utilizando asterisco. Ao utilizar um número exato são retornados os detalhes do número fidelizado e ao utilizar partes de um número é retornada a lista de telefones fidelizados.

Exemplo:

```
local resp = hsm.getFidelization({  
  tokenapi = 'YMJ3NZE7LzylNOAsn4Pww',  
  destination = "5548999999999" })
```

Quando nenhum filtro é utilizado ou é realizada filtragem por partes de um número, o retorno é uma table contendo a lista de telefones fidelizados. Quando a consulta é filtrada por um número específico, o retorno é uma table contendo os campos `associated_data`, `destination`, `fidelization`, `application` e `user`.

hsm.getMessages

Função utilizada para obter uma lista de templates HSM cadastrados no broker.

- Profile:
- **tokenapi** - Chave de acesso a API gerada no cadastro da submídia.
- Opcionais:
- **status**: situação em que se encontra o template após o envio para análise do WhatsApp. Os estados podem ser `approved`, `rejected`, `pending` e `review`.
- **id** – Hash identificador do template.

Exemplo:

```
local resp = hsm.getMessages({  
  tokenapi = 'YMJ3NZE7LzylNOAsn4Pww',  
  status = "approved" })
```

O retorno é uma table contendo a lista de templates cadastrados no broker com os campos locale, name, id, text, text_params e status, além da lista de aplicações disponíveis para fidelização.

hsm.sendMessage

Função utilizada pra enviar um template HSM para um contato do WhatsApp.

- Profile:
 - **tokenapi** - Chave de acesso a API gerada no cadastro da submídia.
 - **destination** - Número de telefone do usuário que receberá a mensagem.
 - **id** - Hash identificador do template.
 - **params** - Lista com os parâmetros da mensagem, ou seja as strings que serão substituídas no texto template (obrigatório caso o template possua parâmetros).
- Opcionais:
 - **name** - Nome do template informado no momento do cadastro.
 - **locale** - Idioma do template.
 - **contact** - Dados para fidelização do contato que podem ser utilizados na resposta do usuário.
 - **data** - Dados associados a fidelização do contato

- **fidelization** - Dados da fidelização do contato contendo a aplicação e o login.
- **application** - Nome da aplicação onde será efetuado o atendimento da chamada, são aceitos os valores UNAOu Interact.
- **service** - Nome do serviço que atenderá a chamada quando o usuário do whatsapp responder a mensagem.
- **user** - Login do usuário na aplicação escolhida que irá atender a chamada quando o usuário do whatsapp responder a mensagem.
- **expire** - Tempo de validade da fidelização, em minutos, para o usuário responder. Padrão e valor máximo de 1440 minutos.

Exemplo:

```
local resp = hsm.sendMessage({  
  tokenapi = 'YMJ3NZE7LzylNOAsn4Pww',  
  id = "4e56-8a40-581c2998f3f7"  
  destination = "5548999999999",  
  params = {  
    name = "João",  
    productName = "Interact"  
  }  
})
```

O retorno é uma table contendo os campos `command`, `destination` e `status`. Em caso de erro o retorno é uma table com os campos `code` e `message`.

Play

Função utilizada para reproduzir todos os tipos de áudios no diálogo. A parametrização de cada tipo de áudio é descrito a seguir.

- profile:
- **file** - reprodução de arquivos pré-gravados e armazenados localmente ou remotamente.

OBSERVAÇÃO

*As mensagens WAV/MP3/DAT, neste **profile file**, quando reproduzidas pela LPU devem seguir um padrão de nome M_0XXXX.ext onde X é o número e ext é uma das extensões compatíveis.*

Exemplo: `play{file="M_00004.WAV"}`
`play{file="http://192.168.0.1/URA_Atendimento/M_00123.WAV"}`

- **tts** - reprodução Dinâmica via tecnologia Text-to-Speech

Exemplo: `play{tts="Teste de URA"}`
`play{tts="O valor do seu saldo é de"..saldo}`

Não é permitido o uso de quebra de linhas neste profile do play.

- **phrase** - reprodução de mensagens dinâmicas pré-gravadas pela Dígitro. Para estas mensagens existe a necessidade de algumas parametrizações, conforme é descrito a seguir.

Exemplo: play{phrase="DASHHMM=09:30|ASHHMM=19:30"} - Divulgação "Das nove horas e trinta minutos as dezenove horas e trinta minutos"

DATE - 01/01, 0101, 01/01/2015, 01012015, TODAY - Divulgação "Primeiro de Janeiro" ou "Primeiro de Janeiro de dois mil e quinze" ou "<data atual do sistema DD/MM/AAAA>"

DDMM - 01/01, 0101, TODAY - Divulgação "Primeiro de Janeiro" ou "<data atual do sistema DD/MM>"

DMMAAAA - 01/01/2015, 01012015, TODAY - Divulgação "Primeiro de Janeiro de dois mil e quinze" ou "<data atual do sistema DD/MM/AAA>"

HOUR - 01:01, 0101, 01:01:01, 010101, NOW - Divulgação "uma hora e um minuto" ou "uma hora e um minuto e um segundo" ou "<hora atual do sistema HH/MM/SS>"

HHMM - 01:01, 0101, NOW - Divulgação "uma hora e um minuto" ou "<hora atual do sistema HH/MM>"

HHMMSS - 01:01:01, 010101, NOW - Divulgação "uma hora e um minuto e um segundo" ou "<hora atual do sistema HH/MM/SS>"

ASHHMM - 01:01, 0101, NOW - Divulgação "a uma hora e um minuto" ou "<hora atual do sistema HH/MM>"

ASHHMMSS - 01:01:01, 010101, NOW - Divulgação "a uma hora e um minuto e um segundo" ou "<hora atual do sistema HH/MM/SS>"

DASHHMM - 01:01, 0101, NOW - Divulgação "da uma hora e um minuto" ou "<hora atual do sistema HH/MM>"

DASHHMMSS - 01:01:01, 010101, NOW - Divulgação "da uma hora e um minuto e um segundo" ou "<hora atual do sistema HH/MM/SS>"

\$ - 12,32 ou 12.32 - Divulgação "doze reais e trinta de dois centavos"

NUMBER - 123 456 789 (máximo de 11 números). - Divulgação "cento e vinte três milhões, quatrocentos e cinquenta e seis mil, setecentos e oitenta e nove"

ONEBYONE - 1234567890 (máximo de 20 números). - Divulgação "um, dois, três, quatro, cinco, seis, sete, oito, nove, zero"

MSG - 4 ou 4,4 - Divulgação "Dígito tecnologia" - "Dígito tecnologia, Dígito tecnologia"

CPF - 123.345.567-43 ou 12.345.678/9301-23 - Divulgação "um dois três, três quatro cinco, cinco seis sete, quatro três" e "um dois, três quatro cinco, seis sete oito, nove três zero um, dois três"

- **params** - este parâmetro é utilizado em conjunto com frase e serve para divulgar mensagens pré-gravadas com parâmetros.

Exemplo: `play{phrase="5051", params="NUMBER=345|$,=20"}` Divulgação "Seu pedido gerado possui o número trezentos e quarenta e cinco com o valor total de vinte reais."

- **cutthrough ou cut** – a divulgação é finalizada quando usuário digitar qualquer cifra. Para isso o parâmetro deve ser atribuído com *true*.

Exemplo: `play{file="http://192.168.0.1/URA_Atendimento/M_00124.WAV" cut="true"}`

- **contínuos** - este parâmetro é utilizado para realizar uma reprodução contínua, não bloqueante. Permite divulgar uma mensagem durante algum processamento da URA, por exemplo, uma consulta web que demore para responder. A reprodução é automaticamente cancelada no próximo *play*. Seu valor padrão é *false*.

Exemplo: `play{file="http://192.168.0.1/URA_Atendimento/M_00124.WAV" contínuos="true"}`

Play não possui retorno.

Não pode ser utilizado dentro das funções pré-definidas **before** e **finish**.

Prompt

Função de interação com o usuário da URA que permite receber cifras digitadas e voz.

- profile:
 - **timeout** - tempo em segundos para aguardar o recebimento de uma informação do usuário. Valor padrão é cinco segundos e o máximo é 99 segundos.
Exemplo: `c = prompt{timeout=1}`
 - **chat_timeout** – tempo em segundos para aguardar o recebimento de uma informação do usuário para chamadas de chat. O máximo é de 900 segundos. Caso este profile não seja setado, será utilizado o profile timeout.
Exemplo: `c = prompt{chat_timeout=1}`
 - **stop_digit** - Cifra que determina a parada do recebimento de informações do usuário.
Exemplo: `c = prompt{stop_digit="#"}`
 - **accumulate** - Quantidade de cifras a serem acumuladas pela função. Valor padrão é um e o máximo é 99 cifras.
Exemplo: `c = prompt{accumulate=7}`
 - **verbose** - Ecoa cifras digitadas pelo usuário.
Exemplo: `c = prompt{verbose="true"}`
 - **recognize** - Utilizado para reconhecimento de voz. Atualmente ela apresenta as tecnologias STT (*Speech To Text*) e KWS. Depende mais de uma

boa URA de reconhecimento de voz e um processamento na LPU para que se tenha um bom aproveitamento deste recurso. O mesmo *prompt* não fará o reconhecimento de cifras e voz em conjunto. Valor padrão é `false`.

Exemplo: `x = prompt{recognize=true}`

- **config** - Deverá ser utilizado quando o **recognize** for configurado como *true*. Neste parâmetro serão enviadas as informações como tipo de servidor de reconhecimento que será utilizado, em qual contexto o áudio será reconhecido quando necessário e o tempo máximo de silêncio para que o reconhecimento seja encerrado.

kind - este campo deve ser utilizado para informar o tipo de reconhecimento de voz que será utilizado, STT.

- **vadsilence** - neste campo deverá ser informado o tempo, em milissegundos, que o usuário poderá ficar em silêncio antes que o reconhecimento de voz. O valor mínimo deste parâmetro é 500, e o máximo 5000. Caso seja inserido um valor fora destes limites, será setado 500 como padrão.

Exemplo 1: `local file = prompt{recognize=true, timeout=40, config={kind="KWS", context="lojas", vadsilence=1000}}`

Exemplo 2: `local file = prompt{recognize=true, timeout=40, config={kind="STT", vadsilence=1000}}`

- **accept_file** - quando este campo for configurado com o valor *true*, é possível encaminhar para esta ação um arquivo ou um texto qualquer. Quando for enviado um arquivo, será retornado um objeto contendo a URL do arquivo no `file_service` bem como as outras informações.

Exemplo: `c = prompt{chat_timeout=90, accept_file="true"}`

Prompt possui retorno com as informações recebidas do usuário, no caso de cifras será o conjunto de cifras digitadas, no caso de voz será a URL do arquivo com as informações reconhecidas. Para realizar o tratamento desta URL, veja em **sttparse**.

Não pode ser utilizado dentro das funções pré-definidas **before** e **finish**.

Typing

Primitiva utilizada para enviar o comando de início e finalização da identificação de digitação, quando o chatbot é acessado por meio do **Chatclient** da Dígitro. Para que o texto "digitando" seja exibido ao usuário, basta chamar a primitiva conforme o exemplo abaixo:

Ex: `typing.start()`

Esta primitiva faz com que o símbolo seja exibido ao usuário até que a script execute o comando para finalizar o modo digitando, ou até que a script execute uma primitiva prompt.

Para deixar de exibir o texto "digitando", basta chamar a primitiva conforme o exemplo abaixo:

Ex: `typing.stop()`

NOTA

*Esta primitiva somente funcionará para chamadas de chat acessadas pelo **ChatClient**.*

File.receive

Função de interação com o usuário da URA para recebimento de arquivos em chamadas de chat.

- profile:
- **timeout** – tempo em segundos para aguardar o recebimento do arquivo. Valor padrão é 5 e o máximo é 900 segundos.

Exemplo:

```
c = file{timeout=60}
```

Arquivos enviados fora deste período não serão coletados pelo **Persona 3**, e a mensagem "Arquivo recusado" será encaminhada para o usuário.

File.send

Função utilizada para que a URA encaminhe um arquivo para o usuário.

- profile:
- **name** - nome do arquivo que será encaminhado.
- **uuid** – código identificador do arquivo. Este código é recebido pela URA ao utilizar a função file receive ou utils.sendFileToFileService.

Exemplo:

```
file.send {name = "M_00012.MP3", uuid = "f6d5e350-b55c-11e6-9ef4-25e80a022cf6"}
```

Xfer

Função utilizada para encaminhar à chamada para um número interno ou externo.

- profile:
 - **destiny** - número para onde será encaminhada a chamada.
Exemplo: e = xfer{destiny=2009}
 - **chat_destiny** – nome do serviço que receberá a transferência da chamada de chat.
Exemplo: e = xfer{chat_destiny="servico1",application = "interact"}
 - **application** – identificação do processo que receberá a transferência da chamada de chat.
Exemplo: e = xfer{chatdestiny="servico1",application = "interact"}
 - **route** - rota a ser utilizada para geração de chamadas externas. A rota deve ser criada via interface do Explorador do PABX.
Exemplo: e = xfer{destiny=32817000,route=0}
 - **account** - conta a ser utilizada para geração de chamadas externas. A conta deve ser criada via interface do Explorador do PABX.
 - **password** - senha da conta a ser utilizada para geração de chamadas externas. A senha deve ser criada via interface do Explorador do PABX.
Exemplo: e = xfer{destiny=32817000,account=2000,password=4567}
 - **identity** - número que será encaminhado ao destino como originador da chamada. Válido somente para chamadas externas.
Exemplo: e = xfer{destiny=32817000,account=2000,password=4567,identity="08001234578"}
 - **cpd** - indica se haverá detecção de voz humana na chamada externa, para encaminhar somente chamadas que uma pessoa realizou o atendimento.

- Exemplo:** e = xfer{destiny=32817000, route=0, cpd="true"}
- **data** - campo para inserir dados a chamada encaminhada.
Exemplo: e = xfer{destiny=32817000, route=0, data="CHAMADA_JA_VALIDA_URA" }
- **priority** - Prioridade da chamada quando encaminhada internamente.
Exemplo: e = xfer{destiny=2009 priority=5}

O **xfer** possui retorno quando ocorrer erro no encaminhamento da chamada. Com isso a LPU pode realizar o tratamento deste erro e tentar encaminhar uma chamada novamente.

Retornos OPERACAO_NAO_PERMITIDA OPERACAO_NAO_OK RAMAL_INVALIDO
CONTA_INVALIDA SENHA_INVALIDA NUMERO_DISCADO_INCOMPLETO PREFI-
XO_BLOQUEADO FALHA_DISCAGEM RAMAL_DISCADO_NULO ROTA_INVALIDA
NUMERO_CHAMADO_INVALIDO DESTINO_OCUPADO RECUR-
SO_NAO_DISPONIVEL PARAMETROS_INCOMPLETOS SENHA_CONTA_INVALIDA
CHAMADA_NAO_ATENDIDA NAO_DETECTOU_VOZ_HUMANA_CPD=\$ (2- Secretária
eletrônica, 3- Fax, 4- Sem licença, 5- Não realizou detecção, 6- Secretária)

Não pode ser utilizado dentro das funções pré-definidas *before* e *finish*.

Drop

Função para desligar a chamada.

- profile:
- Não possui.

Exemplo: drop()

É possível mandar desligar a chamada sem o envio de mensagem de finalização, basta utilizar `drop({sendMessage = false})`

O *drop* não possui retorno.

Dialer.getInfo

Função utilizada para reunir mais informações de um contato de uma chamada gerada pelo discador do Interact

- **profile:**
- **callid** – ID da chamada, gerada pelo discador e que se encontra em andamento, da qual se deseja as informações
local resposta = dialer.getInfo {callid=recover("callid")}

Como resposta, o sistema receberá um JSON contendo as informações do contato, semelhante a:

```
{
  "response": {
    "resource": "dialer",
    "command": "get_info",
    "status": "ok",
    "id": 9,
    "service_name": "teste141",
    "contact": "Carlos",
    "complement": "CPF",
    "phone1": "5000"
  }
}
```

Dialer.setContactState

Função utilizada para alterar as configurações de um contato do discador do Interact.

- profile:
 - **callid** - ID da chamada, gerada pelo discador e que se encontra em andamento, da qual se deseja as informações.
 - **action** - Estado para qual se deseja configurar o contato: reschedule (reagendamento) ou end (Atendimento finalizado na URA).
 - **contact** - Nome do contato.
 - **complement** - Complemento da chamada. Pode ser qualquer tipo de informação que o supervisor do discador ache importante enviar ao agente via dados associados.
 - **phone_list** - Lista de telefones na qual é possível se comunicar com o contato.
 - **schedule** - Faixa de horário na qual o discador pode efetuar a chamada para o contato.

```
local schedule = {}  
schedule.day = "10"  
schedule.month = "10"  
schedule.year = "2019"  
schedule.start = {}  
schedule.start.hour = "11"  
schedule.start.min = "00"  
schedule.finish = {}  
schedule.finish.hour = "11"  
schedule.finish.min = "00"
```

```
local teste2 = dialer.setContactState(callid=recover("callid"), action="reschedule", schedule=schedule)
```

Como resposta, o sistema receberá um JSON contendo as informações do contato, semelhante a:

```
{
  "result": {
    "body": {
      "response": {
        "resource": "dialer",
        "command": "set_contact_state",
        "status": "ok"
      }
    }
  },
  "id": "3"
}
```

Sleep

Função para LPU parar o seu processamento durante um determinado tempo.

- profile:
- Não possui, basta passar o tempo em segundos na função.

Exemplo: sleep(5)

O sleep não possui retorno.

Trace

Função para adicionar log a chamada, auxiliando o debug.

- profile:
 - Não possui, basta passar o texto a ser registrado.
Exemplo: `trace("Chamada passou por aqui")`

Sms

Função para enviar uma mensagem de texto via SMS.

- profile:
 - **destiny** - Número para onde será encaminhado o SMS.
 - **data** - Texto enviado no SMS.
Exemplo: `t = sms{destiny=88880000, data="CREDITO APROVADO"}`
 - **account** - Conta a ser utilizada para envio do SMS.
 - **password** - Senha da conta a ser utilizada para envio do SMS.
Exemplo: `t = sms{destiny=88880000, data="LIGUE PARA 0800 345 6788 e participe do sorteio de 3 casas", account=2000, password=4567}`

O **sms** retornará o resultado somente em caso de ERRO no envio, do contrário será *nil*.

Answer

Função utilizada para atender a chamada dentro da função nativa *before*.

- profile:
 - Não possui.
Exemplo: `answer()`.

O *answer* não possui retorno.

Não pode ser utilizado dentro das funções pré-definidas *starte finish*.

AUXILIAR DE NAVEGAÇÃO

Atuam na navegação do usuário da URA.

Menu

Container de funções utilizadas para navegar entre as funções criadas pelo desenvolvedor da *script* de URA. Atua como facilitadora desta navegação e realiza o mapeamento das funções já navegadas.

- profile:
 - **route** - funciona como uma instrução *switch/case* que a partir do tratamento de um argumento de entrada encaminha para diferentes saídas. Sendo que todas estas saídas são funções definidas pelo desenvolvedor da URA.

Exemplo:

```
menu.route(c, {
  ["1"] = menuAdm,
  ["2"] = menuRH,
  ["3"] = menuCompras,
  ["4"] = menuVendas,
  ["5"] = menuSuporte,
  ["default"] = tratamentoDefault,
  ["timeout"] = timeout,
})
```

- **previous** - Volta a execução da função indicada a partir de uma navegação já realizada. Seu parâmetro é o número de saltos para trás que se deseja realizar.

Exemplo: `menu.previous(2)` - volta para a penúltima função executada.

- **setmain** - Redefine a função onde esta instrução é executada como a principal da *script*. Caso isso não seja feito, a função *start* é considerada como principal.
Exemplo: `menu.setmain()`
- **main** - Direciona a execução para o função principal. Esta pode ser a função *start* ou a função redefinida via *setmain*.
Exemplo: `menu.main()`
- **restart** - Executa novamente a função.

Exemplo: `menu.restart()`

O menu não possui retorno.

Não pode ser utilizado dentro das funções pré-definidas *before* e *finish*.

UTILIDADES GERAIS

Métodos especiais

São funções que dependem de módulos adicionais de software ou hardware

Callback

Função para inserir um contato nas ferramentas de Callback EasyCall e Interact.

- **profile:**
- **number** - Número ao qual a chamada será retornada pelo Callback. Valor padrão é o número originador.
Exemplo: `c = callback{number=32817000}`
- **group** - Número do grupo onde será armazenado o contato. Valor padrão é o grupo do qual esta chamada foi transbordada.

OBSERVAÇÃO

Esta informação não é recuperada em estruturas multi-site.

Exemplo: `c = callback{group=2020}`

- **data** - Dados a serem inseridos no registro do Callback.
Exemplo: `c = callback{dados="PENDENCIA_FINANCEIRA"}`
- **site** - *host* onde deverá ser registrado o Callback quando em estrutura multi-site. É opcional. Quando não informado, é considerado local.
Exemplo: `c = callback{site="ecgsp"}`
- **aplication** - informa qual a aplicação será utilizada: interact ou easycall. Caso não seja informada, será identificado em tempo de execução.
Exemplo: `c = callback{aplication="easycall"}`
Exemplo: `c = callback{aplication="interact"}`

Parâmetros específicos por aplicação:

- EasyCall:
 - **period** - Faixa de horário para o retorno da chamada pelo callback, no formato 'HH:MM-HH:MM'. É opcional.
Exemplo: `c = callback{period="08:00-10:00"}`
- Interact:
 - **name** - nome do cliente/usuário que aparecerá para o agente. Caso não seja informado, será configurado o próprio número de origem da chamada.

- **start** - Opcional. Data/hora inicial para o período no qual é permitida a geração da chamada, no formato DD/MM HH:MM.
- **stop** - Opcional. Data/hora final para o período de geração da chamada, no formato DD/MM HH:MM.

O Callback retornará o resultado somente em caso de ERRO no registro, do contrário será *nil*.

Report

Função para inserir informações nos relatórios de URA do tipo atributo valor. Este tipo possibilita o envio de informações dinâmicas da LPU para os relatórios, tais como CPF, Códigos Digitados, Pesquisa de Satisfação e Classificação da Chamada.

- **profile:**
 - Não possui, mas existe a necessidade de preencher três campos (<nome>, <valor>, <tipo>).
 - **nome** - Nome do atributo
 - **valor** - Valor do atributo
 - **tipo:** informa qual o tipo de variável será enviada: 0 Geral, 1 Classificação/Filtro, 2 Pesquisa de Satisfação.
Geral, deve ser utilizado para envio de informações granulares, tais como CPF, Códigos digitados na URA. Visualização nos modelos 30604 e 30609.
Exemplo: report("ID", recover("callid"), 0)

Classificação/filtro, como o próprio nome já diz é utilizado para classificação ou filtro das chamadas. Ou seja, propicia que as chamadas sejam filtradas por um resultado do atributo. Classificação: Modelos 30601,30602,30604,30605,30606. Visualização: Modelo 30608.

Exemplo: `report("Tipo_Cliente", "Vip", 1)`

Pesquisa, neste tipo o desenvolvedor pode criar suas próprias Pesquisas de Satisfação. Ele funciona de maneira simples, as perguntas são os nomes das variáveis e os valores são as respostas. Visualização: Modelo 30607.

Exemplo: `report("Qualidade Atendimento", "1", 2)`

O `report` não possui retorno.

Checkpoint

Função para inserir pontos de verificação. Quando uma chamada passar por este ponto serão geradas informações estatísticas para o relatório. Os modelos dos relatórios que utilizam estas informações são: 30601, 30602 e 30604.

- `profile`:
 - Não possui, mas existe a necessidade de preencher o nome do ponto de verificação.

Exemplo: `checkpoint("Menu_Vendas")`

O `checkpoint` não possui retorno.

sendAlarm

Função utilizada para gerar um alarme. Quando uma chamada passar por este ponto será gerado um alarme que será exibido na Supervisão de Alarmes.

- profile:

Exemplo: `sendAlarm("Texto do alarme gerado pela script de URA")`

O `sendAlarm` não possui retorno.

nlp.dialogflow

Função utilizada para comunicação com a plataforma de linguagem natural Dialogflow.

- profile:

- Não possui, mas existe a necessidade de enviar os parâmetros de consulta à plataforma.
- **url** – URL da API;
- **text(query)** – texto utilizado para processar a linguagem natural;
- **autorization** – cabeçalho HTTP com o *token* de acesso a API;
- **sessionId** – identificador único da conversa (máximo de 36 caracteres).
- opcionais:
 - **lang** (default pt-BR)
 - **timezone** (default America/São_Paulo)

Em redes que utilizam proxy é necessário o envio do parâmetro **proxy** na mesma estrutura da consulta.

Exemplo:

```
proxy = 'http://<USUARIO>:<SENHA>@<IP_PROXY>:<PORTA_PROXY>'
```

O retorno é uma table contendo os campos **id**, **timestamp**, **lang**, **result**, **status** e **sessionid**.

nlp.watson

Função utilizada para comunicação com a plataforma de linguagem natural Watson.

- profile:
 - Não possui, mas existe a necessidade de enviar os parâmetros de consulta à plataforma.
 - **username** - Nome de usuário do serviço
 - **password** - Senha do serviço
 - **url** - URL do workspace
 - **text** - Texto utilizado para processar a linguagem natural
 - **conversation_id** - Identificador único da conversa

Em redes que utilizam proxy é necessário o envio do parâmetro **proxy** na mesma estrutura da consulta.

Exemplo:

```
proxy = 'http://<USUARIO>:<SENHA>@<IP_PROXY>:<PORTA_PROXY>'
```

O retorno é uma table contendo os campos **intents**, **entities**, **input**, **output** e **context**.

record.start

Função utilizada para iniciar a gravação de chamada na URA. Quando uma chamada passar por este ponto é enviado um comando ao gravador solicitando o início da gravação.

- profile:
 - **operador** – login do operador associado à programação de gravação no Record
- opcional:
 - **description** – descrição que será associada a gravação.

Em caso de falta de parâmetros, parâmetros incorretos ou erro na comunicação com o gravador a chamada será derrubada.

Exemplo: local resposta = record.start {operator = "persona"}

O retorno é uma table contendo os campos **recordId**, **dateTime** e **file**.

record.stop

Função utilizada para finalizar a gravação de chamada na URA. Quando uma chamada passar por este ponto é enviado um comando ao gravador solicitando o fim da gravação atual.

- profile:
 - Não possui.

Caso não exista uma gravação em andamento ou ocorra erro na comunicação com o gravador a chamada será derrubada.

Exemplo: `record.stop ()`

O **record.stop** não possui retorno.

contact.insert

Função `contact.insert` utilizada para inserir um contato no banco de dados da aplicação.

Campos:

- `id_bot` - Identificador do robô utilizado no chat.
- `id_tipo_submidia` - Tipo da submídia vinculada ao robô.
- `identificacao` - Identificador enviado pela submídia utilizada pelo robô.
- `id_chat` - Identificador do chat fornecido pela submídia utilizada pelo robô.
- `complemento` - Informações complementares adicionadas ao cadastro do contato.

Pode ser utilizado nenhum ou mais parâmetros no `insert`, os parâmetros não utilizados serão preenchidos com os valores padrão fornecidos pelo chat.

Exemplo:

```
contact.insert {complemento = "CPF 000.000.000-00"}
```

O `contact.insert` não possui retorno

contact.get

Função `contact.get` utilizada para recuperar um contato cadastrado na aplicação.

Campos:

- `id_chat` - Identificador do chat fornecido pela submídia utilizada pelo robô.
- `id_bot` - Identificador do robô utilizado no chat.

Exemplo:

```
local chatData = recover ("bot_data")
contact.get {id_chat= chatData.id_chat, id_bot= chatData.id_bot}
```

O retorno é um objeto contendo o campo `rows` que contém um ou mais objetos com os campos `identificacao`, `id_chat`, `id_tipo_submidia`, `id_bot` e complemento quando este possuir valor cadastrado no banco de dados

contact.remove

Função utilizada para remover um contato do banco de dados da aplicação.

Campos:

- `id_chat` - Identificador do chat fornecido pela submídia utilizada pelo robô
- `id_bot` - Identificador do robô utilizado no chat

Caso os identificadores do chat e robô não forem especificados, será removido o próprio contato que está na conversa.

Exemplo:

```
contact.remove {}
```

O `contact.remove` não possui retorno

VARIÁVEIS DE AMBIENTE

Variáveis setadas no início da chamada.

- **associated_data** - dado associado à chamada.
- **to** - número para qual a chamada foi originalmente realizada.
- **start_message** - recupera a primeira mensagem digitada pelo usuário no primeiro atendimento realizado na plataforma ao acessar a solução via chat.
- **form** – Array de informações que serão enviadas na transferência da mídia chat para o agente Interact. Esta variável poderá vir preenchida em chamadas do ChatClient onde poderá ser recuperada e alterada na URA. Em chamadas originadas de outras mídias como: WhastApp, Telegram, etc., não terá conteúdo, mas poderá ser preenchida para envio ao agente.

Exemplos: `[{cpf="000.000.000-00"}, {navegaçãoURA="Abertura;Suporte;Atendimento"}, {FILE=[{name="file_5.jpg", path="http://url_do_arquivo"}]}`

- **from** - número do originador da chamada.
- **callid** - id da chamada.
- **group** - grupo da URA que está realizando o atendimento da chamada.

- **lastgroup** – último grupo por onde a chamada passou antes de ser encaminhada para a URA.
- **media_type** - tipo de mídia que está navegando na URA (chat ou voice).
- **messageType** - para utilizar o mesmo tipo de envio em todas as mensagens de *chat* basta setar um dos valores a seguir:

audio: com este valor setado serão encaminhados apenas os arquivos de áudio e não as mensagens de texto.

both: com este valor setado serão encaminhados tanto as mensagens de texto quanto as de áudios.

Para setar esses valores deve-se utilizar a função *store*.

Para utilizar estas variáveis na LPU, basta utilizar as funções *store* ou *recover*.

Exemplo: `dado_associado = recover("associated_data")`

`store ("messageType", "audio")`

- **speaker** - para utilizar o mesmo locutor em todas as divulgações de TTS sem setar o parâmetro *speaker* de cada mensagem basta configurar o nome do locutor neste campo (marcos, anapaula).

FUNÇÕES ÚTEIS

Funções que simplificam o processamento de informações na LPU.

Calendar

Função para utilizar os calendários de atendimento.

- profile:
- Não possui, basta passar os nomes dos calendários separados por vírgula.

Exemplo: t = calendar("CAL_DIA,FERIADOS")

O **calendar** retornará **true** caso dentro do horário ou o calendário não existir, fora do horário retornará **false**. Os calendários deverão estar previamente cadastrados no sistema. Caso o calendário não exista, a função retornará **true**.

Store

Função utilizada para atribuir uma variável dentro da LPU. A variável permanecerá acessível até o final da execução da *script* de URA.

- profile:
- Não possui, mas existe a necessidade de preencher dois campos (<nome>, <valor>)

nome - Nome do variável

valor - Valor do variável

Exemplo: store("cifra", c)

O **store** não possui retorno.

Recover

Função utilizada para recuperar uma informação de uma variável da LPU.

- profile:
- Não possui, mas existe a necessidade de preencher um campo (<nome>).

nome - Nome da variável

Exemplo: local codigo = recover("cifra")

O **recover** retorna o valor da variável a partir do seu nome.

Format

Função para facilitar formatação de informações.

- profile:
- **currency** - formata um determinado valor monetário.

Exemplo: local cotacao = format.currency(dolar_comercial)

O **format** retorna o valor da variável formatado de acordo com o profile solicitado.

Funções do Lua exportadas para a LPU

- funções:
- **string** - função de manipulação de *strings*.

- **table** - função de manipulação de *tables*.
- **tonumber** - converte conteúdo da variável para um número.
- **tostring** - converte conteúdo da variável para uma *string*.
- **date** - retorna uma cadeia ou uma tabela contendo data e hora.
- **time** - retorna o tempo corrente quando chamada sem argumentos ou um tempo representando a data e a hora especificados pela tabela fornecida.
- **ipairs** - retorna três valores: uma função iteradora, a tabela **t** e 0, de modo que a construção irá iterar sobre os pares (1,t[1]), (2,t[2]), ..., até a primeira chave inteira ausente da tabela.
- **pairs** - retorna três valores: a função **next**, a tabela **t** e **nil**, de modo que a construção irá iterar sobre todos os pares chave.valor da tabela **t**.
- **next** - permite a um programa percorrer todos os campos de uma tabela. Seu primeiro argumento é uma tabela e seu segundo argumento é um índice nesta tabela.
- **type** - retorna o tipo de seu único argumento, codificado como uma cadeia de caracteres. Os resultados possíveis desta função são "*nil*" (uma cadeia de caracteres, não o valor *nil*), "*number*", "*string*", "*boolean*", "*table*", "*function*", "*thread*" e "*userdata*".

Utils

Container de funções utilizadas para validações de cartões, acessos externos, banco de dados, etc.

- funções:
- **web** - Utilizado para comunicação com *web services* via HTTP e HTTPS nos métodos GET e POST. Esta função prevê a formatação de todo conteúdo do *request*. O argumento obrigatório é a URL ou URI, o restante é opcional.

onal, mas podemos citar algumas opções: qs, useQuerystring, method, headers, body, form, formData, multipart, auth, json, jsonReviver, preamble-CRLF, followRedirect, maxRedirects, encoding, pool, timeout, proxy, oauth, hawk, strictSSL, agentOptions, time, jar, was, httpsSignature, localAddress, gzip, tunnel, proxyHeaderWhiteList, proxyHeaderExclusiveList, removeRefererHeader, rejectUnauthorized entre outros.

```
Exemplo: local consulta = {  
    url =  
'http://www.webservicex.net/CurrencyConvertor.aspx/ConversionRate?FromCurrency=B  
RL&ToCurrency=USD',  
    method = 'GET',  
}  
resposta = utilits.web(consulta)
```

retorno é uma *table* contendo *headers*, *body*, *statusCode* e *error*.

decodeXML - lê o conteúdo XML e transforma em uma *table*. É possível entrar somente com o XML para ser tratado, ou passar uma *table*. Quando passado *table*, é possível informar no segundo parâmetro a partir de qual tag deseja-se realizar o tratamento. Quando existirem atributos e conteúdos para uma mesma tag, o conteúdo será armazenado como *content*.

Exemplo 1:

```
local xml = "<body><result><name>Jose</name><id>12345</id></result></body>"  
local resultado = utilits.decodeXML(resposta.body)  
play{"tts:Olá " .. resultado.body.result.name}
```

Exemplo 2:

```
local xml = "<body><result name=Jose id=12345/></body>"  
local resultado = utils.decodeXML{resposta.body}  
play{"tts:Olá "..resultado.body.result.name}
```

Exemplo 3:

```
local xml = "<body><result name=Jose id=12345/></body>"  
local resultado = utils.decodeXML{resposta.body, "result"}  
play{"tts:Olá "..resultado.name}
```

Exemplo 4:

```
local xml = "<body><result name=Jose id=12345>Bem vindo Jose.</result></body>"  
local resultado = utils.decodeXML{resposta.body, "result"}  
play{"tts:"..resultado.content}
```

O retorno é uma *table* com o conteúdo do XML.

decodeJSON - lê o conteúdo no formato JSON e transforma em uma *table*.

Exemplo: local resultado = utils.decodeJSON(resposta)

O retorno é uma *table* com o conteúdo do JSON.

encodeJSON - lê o conteúdo de uma *table* e transforma no formato JSON.

Exemplo: local payload = utils.encodeJSON(contenido)

O retorno é um conteúdo no formato JSON.

sendFileToFileService - utilizado para enviar arquivos para o processo de gerenciamento e envio de arquivos para mídias de chat. Esta ação pode ser utilizada para enviar arquivos de URLs ou arquivos presentes no projeto da URA.

url - utilizado para baixar um arquivo de um servidor externo e enviá-lo para o gerenciador de arquivos das conversas de chat

Exemplo:

```
utils.sendFileToFileService({url='https://intranet.digitro.com.br/daf/manuais/  
EasyCall_Persona_3-1.28-v1.pdf'})
```

file - Utilizado para enviar um arquivo, presente no projeto da URA, para o gerenciador de arquivos das conversas de chat

Exemplo:

```
utils.sendFileToFileService({file='M_00012.MP3'})
```

Como retorno, esta função devolverá um objeto contendo dois parâmetros. O primeiro, **name**, possuirá o nome do arquivo, e o segundo, **uuid**, conterá o ID do arquivo necessário para utilização da função `file.send`.

sendFileToSTT - utilizado para enviar um arquivo de áudio para o servidor de reconhecimento de voz STT (*Speech to Text*). Esta primitiva retornará apenas o texto que foi reconhecido no áudio, ou então um objeto, contendo o erro ocorrido durante a comunicação com o servidor.

Os formatos de áudio suportados são OGG, MP3 e WAV.

- **file** – arquivo recebido através da primitiva `prompt`.

Exemplo:

```
local resposta = utils.sendFileToSTT {"file = teste", timeout = 120}
```

- **timeout** – tempo máximo, em segundos, que o processo esperará por uma resposta do STT..

Exemplo:

```
local resposta = utils.sendFileToSTT {file = teste, "timeout = 120"}
```

A assertividade do texto reconhecido é diretamente proporcional a qualidade do áudio enviado ao STT. O Messenger do Facebook envia áudios com muito ruído, dificultando o reconhecimento.

Como retorno esta função retornará apenas o texto que foi reconhecido no áudio, ou então um objeto, contendo o erro ocorrido durante a comunicação com o servidor.

SQL

Utilizado para comunicação com Banco de Dados. Atualmente possuímos acessos aos bancos: Oracle, MS SQL Server e Postgres. Para acessar estes bancos basta utilizar alguns parâmetros definidos dentro da própria LPU. A primitiva `sql` deve ser chamada informando-se uma configuração contendo:

- **type** - tipo de conexão (mssql, oracle, postgres, jdbc).
- **statementType** - tipo de comando SQL ('query').
- **query** - consulta SQL.
- **url** - configuração para acesso ao banco de dados (pode variar conforme o banco de dados).
- **server** - IP da máquina.
- **port** - porta tcp.
- **database** - nome do database ou SID.
- **user** - nome do usuário.
- **password** - senha do usuário.
- **options** - opções específicas.
- **timeout** – define o tempo máximo da conexão e execução do comando SQL no banco de dados (tempo em ms). Valor *default* 10000 ms (10 s).
- **limit** – define a quantidade máxima de registros que serão retornados na resposta. Valor *default* 100.

Oracle

```
var oraclecfg = { user = 'uraip',  
                 password = 'uraip',  
                 server = '192.168.167.208',  
                 port = 1521,  
                 database = 'sid' };
```

```
var query = {
```

```
type = 'oracle',  
url = oraclecfg,  
query = 'SELECT valor FROM salario_minimo where ano = 1999',  
statementType : 'query'  
};
```

Exemplo: resp = utils.sql(mssql)

MS SQL Server

```
var mssqlcfg = { user = 'uraip',  
password = 'uraip',  
server = '192.168.167.197',  
port = 1433,  
database = 'uraip',  
options = { connectTimeout = 10000, tdsVersion = '7_1' } ;  
var mssql = {  
type = 'mssql',  
url = mssqlcfg,  
query = 'SELECT valor FROM salario_minimo where ano = 1999',  
statementType = 'query'  
};
```

Exemplo: resp = utils.sql(query)

PostgreSQL

```
var pgconf= { server= '192.168.167.208', port= 5432, database = 'uraip',  
user= 'uraip', password = 'uraip' }  
var pgquery= {  
    type = 'postgres',  
    url = pgconf,  
    query = 'SELECT valor FROM salario_minimo where ano = 1999',  
    statementType = 'query'  
};
```

Exemplo: resp = utils.sql(pgquery)

Retorno:

- **error:** indica que uma falha ocorreu.
- **rows:** retorna as linhas e colunas da consulta em uma *table*.
- Se ocorrer alguma falha na execução, retorna o atributo *l*.
- Se a operação retornar algum registro, retorna uma *table* com as linhas e colunas no atributo *rows*.
- Se a operação não retornar nenhum registro ou for uma operação que não retorna dados (INSERT, UPDATE, DELETE) e não ocorrer erro, não retorna nenhum atributo.

Exemplo:

```
resp = utils.sql(pgquery)  
if not resp or resp.error then  
    play { tts = "Falha na consulta ao banco de dados!" }
```

```
elseif resp.rows then
  util.listResp(resp.rows)
  for k,v in ipairs(resp.rows) do
    if v.valor ~= nil then
      ...
    end
  end
else
  -- não retornou dados
end
```

Socket

Função utilizada para conectar a um determinado servidor externo e enviar um dado. Para o preenchimento dos parâmetros basta seguir o modelo:

```
var dados_socket={
  host="127.0.0.1",
  port="1234",
  timeout="5",
  data="SEND_DATA=123456789",
}
```

Exemplo: resp = utils.socket(dados_socket)

Retorno do *socket* é uma *table* contendo o campo *errors* somente em caso de ERRO no envio, do contrário será *nil* e no campo *content* uma *string* retornada pelo servidor caso houver esse retorno.

valida_cpf

Validador de CPF basta passar o número do CPF.

Exemplo: `local res = utilits.valida_cpf(c)`

Retorno do *valida_cpf* é *true* se o CPF for válido e *false* em caso de falha na validação

valida_cnpj

Validador de CNPJ basta passar o número do CNPJ.

Exemplo: `local res = utilits.valida_cnpj(c)`

Retorno do *valida_cnpj* é *true* se o CNPJ for válido e *false* em caso de falha na validação

readfile

Lê arquivo local a partir do *path* informado.

Exemplo: `local lines = utilits.readfile("lista_vip")`

Retorno é um *array* de linhas lidas no arquivo.

file2string

Lê arquivo local a partir do *path* informado.

Exemplo: local lista_t9 = utils.file2string("lista_telefonica.cfg")

Retorno é uma *string* com todo conteúdo do arquivo lido.

damm_generate

Gerador de dígito verificador.

Exemplo: local digit = utils.damm_generate(47135336)

Retorno é o dígito verificador, no exemplo acima o resultado é 2.

damm_validate

Validador de dígito verificador.

Exemplo: local valid = utils.damm_validate(471353362)

Retorno é *true* se o dígito é válido e *false* se inválido

luhn_generate

Gerador de dígito verificador para cartão de crédito.

Exemplo: `local number = utilits.luhn_generate(545762389823411)`

Retorno é o dígito verificador, no exemplo acima o resultado é 3.

luhn_validate

Validador de cartão de crédito.

Exemplo: `local result = utilits.luhn_validate(5457623898234113)`

Retorno é *true* se o dígito é válido e *false* se inválido.

utf2iso

Converte o conteúdo UTF-8 em ISO

Exemplo: `local txt_iso = utilits.utf2iso(txt_utf8)`

Retorno é o conteúdo convertido no formato ISO.

iso2utf

Converte o conteúdo ISO em UTF-8

Exemplo: `local txt_utf8 = utilits.iso2utf(txt_iso)`

Retorno é o conteúdo convertido no formato UTF-8.

string2table

Transforma uma *string* em uma *table* a partir de um *pattern*.

Exemplo: `local agenda = utilits.file2string("listatelefonica.cfg")`

`local content = utilits.string2table({ string=agenda, pattern="%d+)=([^\;]*);([^\;]*);([^\;]*);[^\n]*" })`

Retorna a *table* contendo as informações passadas na *string*, se houver um cabeçalho na *string* a formatação seguirá esse cabeçalho. Ex: "NOME, TELEFONE, Joao luiz, 32817000, Tiago Pedro, 88880000" será igual a `table = { NOME=Joao Luiz, TELEFONE=32817000, NOME=Tiago Pedro, TELEFONE=88880000 }`

filter

Utilizado para localizar um conteúdo dentro de uma *table*. Passando como entrada a tabela a ser pesquisada, a(s) coluna(s) onde serão pesquisadas, podendo ser o número ou nome da coluna e o *pattern* para pesquisa.

Exemplo: `local newDic = utilits.filter({list=dic, columns="2", pattern=regex})`

Retorna a *table* com as linhas que bateram com o filtro.

sttparse

Função utilizada para especializar o reconhecimento de voz do tipo STT. Para o uso basta passar o parâmetro link. link é a URL gerada a partir do reconhecimento de voz retornado do prompt.

Exemplo: local j = utils.sttparse(link=file)

O retorno será uma *table* contendo todas as frases reconhecidas naquela ação. Caso o usuário fale com algumas pausas, as frases poderão vir separadas, em posições diferentes do vetor, ou então virem juntas em uma mesma posição.

A seguir um exemplo contendo o padrão do retorno.

```
["Como você o chama", "Pode ser de Carlos"]
```

EXEMPLO DE SCRIPT DE URA

```
start = function ()  
  store ("URLFrase", "http://192.168.0.1/msgs/M_04000.WAV")  
  store ("Calendario", "Cliente")  
  store ("Arquivo", "ArquivosLeitura/texto.txt")  
  store ("Arquivo2", "ArquivosLeitura/texto2.txt")  
  store ("WS", "http://192.168.0.1/teste/CarcEsp.php")
```

```
store ("Mensagem", "Teste de SMS.")
store ("MaqCallback", "192.168.0.1")
store ("Periodo", "08:00-10:00")
store ("DadoCallback", "Teste URA")
```

```
checkpoint("Inicio")
play {tts="Bem vindos a URA de testes do Persona 3"}
menu.next (menuPrincipal)
```

```
end
```

```
function menuPrincipal ()
```

```
    checkpoint("Menu Principal")
```

```
    menu.setmain ()
```

```
    play {tts="Digite 1 para teste de frases. Digite 2 para consulta a banco de dados. Digite
3 para consulta a um webservice. Digite 4 para registrar callback. Digite 5 para transfe-
rência de chamada. Digite 6 para enviar um SMS. Digite 7 para teste de calendário. Digi-
te 8 para testes de funções internas.", cut = "true"}
```

```
    local c = prompt ()
```

```
    menu.route (c, {
```

```
        ["1"] = Frases,
```

```
        ["2"] = BancoDados,
```

```
        ["3"] = Webservice,
```

```
        ["4"] = RetornoChamada,
```

```
        ["5"] = Transferencia,
```

```
        ["6"] = MensagemTexto,
```

```
        ["7"] = Calendario,
```

```
        ["8"] = FuncoesInternas,  
        ["default"] = Erro,  
        ["timeout"] = Erro,  
    }  
)  
end  
function Erro ()  
    checkpoint("Erro")  
    play{file="http://192.168.0.1/mmsgs/M_04000.WAV", cut="true"}  
    menu.previous(1)  
end  
function Frases()  
    checkpoint("Divulga Frases")  
    local URL = recover ("URLFrase")  
    play {tts="Frase 1."}  
    play{file=URL, cut="true"}  
    play {tts="Frase 2."}  
    play{phrase="DATE=TODAY"}  
    play {tts="Frase 3."}  
    play{phrase="DDMM=TODAY"}  
    play {tts="Frase 4."}  
    play{phrase="DDMMAAAA=TODAY"}  
    play {tts="Frase 5."}  
    play{phrase="HOUR=NOW"}
```

```
play {tts="Frase 6."}
play{phrase="HHMM=NOW"}
play {tts="Frase 7."}
play{phrase="HHMMSS=NOW"}
play {tts="Frase 8."}
play{phrase="ASHHMM=NOW"}
play {tts="Frase 9."}
play{phrase="ASHHMMSS=NOW"}
play {tts="Frase 10."}
play{phrase="DASHHMM=NOW"}
play {tts="Frase 11."}
play{phrase="DASHHMMSS=NOW"}
play {tts="Frase 12."}
play{phrase="$=10,20"}
play {tts="Frase 13."}
play{phrase="NUMBER=123456789"}
play {tts="Frase 14."}
play{phrase="ONEBYONE=123456789"}
play {tts="Frase 15."}
play{phrase="MSG=4"}
play {tts="Frase 16."}
play{phrase="CPF=08229964998"}
menu.main()
end
```

```
function BancoDados ()
  checkpoint("Consulta Banco de Dados")
  play {tts="Teste de consulta a banco de dados Postgres"}
  local serverBD = {
    server = '192.168.0.1',
    port = 5432,
    database = 'data',
    user = 'user',
    password = 'senha',
  }
  local consulta = {
    tipo = 'postgres',
    url = serverBD,
    bd = 'POSTGRES',
  }
  local Pedido = "select * from consulta_cliente where dt_inicial = '2014-08-05 14:12:14.0'
LIMIT 3"

  local resposta = utils.sql({ type = consulta.tipo, url = consulta.url, query = Pedido, statementType = "query"})

  play {tts="Consulta Realizada."}
  play {tts="O valor da identificação 1 é " .. resposta.rows[1].identificacao}
  play {tts="O valor da identificação 2 é " .. resposta.rows[2].identificacao}
```

```
play {tts="O valor da identificação 3 é " .. resposta.rows[3].identificacao}  
play {tts="teste de execução de procedure"}  
Pedido = "select get_municipio(41,'32349453',null)"  
resposta = utils.sql({ type = consulta.tipo, url = consulta.url, query = Pedido, state-  
mentType = "query"})  
play {tts="O valor do retorno da procedure é " .. resposta.rows[1].get_municipio}  
menu.previous (1)  
end
```

```
function Webservice ()  
  checkpoint("Consulta Webservice")  
  local consulta = {  
    url = 'http://www.webservicex.net/CurrencyConvertor.asmx',  
    method = 'POST',  
    proxy = 'http://user:senha@192.168.0.1:3333',  
    body = '<?xml version="1.0" encoding="utf-8"?><soap:Envelope  
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
xmlns:xsd="http://www.w3.org/2001/XMLSchema"  
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"> <soap:Body> <Conver-  
sionRate xmlns="http://www.webserviceX.NET/"> <FromCurren-  
cy>EUR</FromCurrency> <ToCurrency>BRL</ToCurrency> </ConversionRate>  
</soap:Body></soap:Envelope>',  
    timeout = 5000,  
  }  
  play {tts="Vai realizar a consulta a um Webservice SOAP"}  
  local resposta = utils.web(consulta)
```

```
if resposta and not resposta.error then
  play {tts="Um Euro vale "}
  local Euro = utilits.decodeXML(resposta.body, "ConversionRateResponse")
  play{tts=Euro.ConversionRateResult}
else
  play{tts="Consulta não funcionou"}
end
play {tts="Vai realizar consulta a um Webservice que retorna caracteres especiais."}
consulta = {
  url = recover("WS"),
  method = 'GET',
  timeout = 5000,
}
resposta = utilits.web(consulta)
if resposta and not resposta.error then
  play {tts=utilits.utf2iso(resposta.body)}
else
  play{tts="Consulta não funcionou"}
end
menu.previous(1)
end

function RetornoChamada ()
  checkpoint("Callback")
end
```


play {tts="Teste de inserção de chamadas no Callback. Digite 1 para inserir a chamada em um grupo específico. Digite 2 para inserir a chamada em um grupo específico de outra máquina. Digite 3 para inserir chamada no grupo transbordado, Digite 4 para inserir chamada com um período específico. Digite 5 para inserir chamada com dados associados.", cut="true"}

```
local M = prompt {}
menu.route (M, {
    ["1"] = RetornoEsp,
    ["2"] = RetornoMaq,
    ["3"] = Retorno,
    ["4"] = RetornoPeriodo,
    ["5"] = RetornoDados,
    ["default"] = Erro,
    ["timeout"] = Erro,
})
```

end

```
function RetornoEsp ()
    checkpoint("Callback Especifico")
    play {tts="Digite o número do grupo onde o callback deverá ser registrado.", cut="true"}
    local grupo = prompt {accumulate=4}
    local result = callback { number=32812003, group=grupo}
    if not result then
        play {tts="Callback registrado com sucesso."}
```

```
else
  play {tts="Falha no registro do Callback."}
end
menu.previous(2)
end
```

```
function RetornoMaq ()
  checkpoint("Callback outra máquina")
  local Maq = recover("MaqCallback")
  play {tts="Digite o número do grupo onde o callback deverá ser registrado.", cut="true"}
  local grupo = prompt {accumulate=4}
  local result = callback {number=32812006, group=grupo, site=Maq}
  if not result then
    play {tts="Callback registrado com sucesso."}
  else
    play {tts="Falha no registro do Callback."}
  end
  menu.previous(2)
end
```

```
function Retorno ()
  checkpoint("Callback simples")
  play {tts="Digite o número do grupo onde o callback deverá ser registrado.", cut="true"}
  local grupo = prompt {accumulate=4}
```

```
local result = callback {number=32812003}
if not result then
  play {tts="Callback registrado com sucesso."}
else
  play {tts="Falha no registro do Callback."}
end
menu.previous(2)
end
function RetornoPeriodo ()
  checkpoint("Callback com periodo")
  local Maq = recover("MaqCallback")
  local periodo = recover ("Periodo")
  play {tts="Digite o número do grupo onde o callback deverá ser registrado, com período.", cut="true"}
  local grupo = prompt {accumulate=4}
  local result = callback {number=32812676, site=Maq, group=grupo, period=periodo}
  if not result then
    play {tts="Callback registrado com sucesso."}
  else
    play {tts="Falha no registro do Callback."}
  end
  menu.previous(2)
end
```

```
function RetornoDados ()
  checkpoint("Callback com dados")
  local Maq = recover("MaqCallback")
  --local periodo = recover ("Periodo")
  local dadosAss = recover ("DadoCallback")
  play {tts="Digite o número do grupo onde o callback deverá ser registrado, com dados
associados.", cut="true"}
  local grupo = prompt {accumulate=4}
  local result = callback {number=32812668, site=Maq, group=grupo, data=dadosAss}
  if not result then
    play {tts="Callback registrado com sucesso."}
  else
    play {tts="Falha no registro do Callback."}
  end
  menu.previous(2)
end
function Transferencia()
  checkpoint("Transferencia")
  play {tts="Digite 1 para transferência normal, 2 para transferência com rota 0, 3 para
transferência com conta e senha. Ou 4 para transferência com CPD.", cut = "true"}
  local M = prompt {accumulate=1, timeout=5}

  menu.route (M, {
    ["1"] = transfNormal,
    ["2"] = transfRota,
```

```
        ["3"] = transfConta,  
        ["4"] = transfCPD,  
        ["default"] = Erro,  
        ["timeout"] = Erro,  
    }  
)  
end  
  
function transfNormal()  
    checkpoint("Transferencia normal")  
    play {tts="Digite o número para geração de chamada.", cut="true"}  
    local ramal = prompt {accumulate=4}  
    if ramal then  
        xfer{destiny=ramal}  
    else  
        menu.next(Erro)  
    end  
end  
  
function transfRota ()  
    checkpoint("Transferencia por rota zero")  
    play {tts="Digite o numero para geracao da chamada via rota 0", cut="true"}  
    local ramal = prompt {accumulate=4}  
    if ramal then
```

```
xfer{destiny=ramal, route=0}  
else  
menu.next(Erro)  
end  
end
```

```
function transfConta ()  
  checkpoint("Transferencia conta e senha")  
  play {tts="Digite o número para geração da chamada por conta e senha", cut="true"}  
  local ramal = prompt {accumulate=4}  
  play {tts="Digite o número da conta."}  
  local conta = prompt {accumulate=4}  
  play {tts="Digite a senha."}  
  local senha = prompt {accumulate=4}  
  if ramal and conta and senha then  
    xfer {destiny=ramal, account=conta, password=senha}  
  else  
    menu.next(Erro)  
  end  
end
```

```
function transfCPD ()  
  checkpoint("Transferencia com CPD")  
  play {tts="Digite o número para geração de chamada via rota 0 com CPD", cut="true"}
```

```
local ramal = prompt {accumulate=4}
if ramal then
  xfer{destiny=ramal, route=0, cpd="true"}
else
  menu.next(Erro)
end
end
```

```
function MensagemTexto ()
  checkpoint("Envia SMS")
  play {tts="Digite o número para onde a mensagem será enviada.", cut="true"}
  local ramal = prompt{accumulate=4}
  if not sms{destiny=ramal, data=recover("Mensagem")} then
    play {tts="Mensagem enviada"}
  else
    play {tts="Falha no envio da mensagem"}
  end
  menu.previous(1)
end
```

```
function Calendario ()
  checkpoint("Verifica Horário de Atendimento")
  local Calendario = recover ("Calendario")
  play {tts="Verificação do calendário " .. Calendario}
```

```

if calendar(Calendario)=="true" then
    play {tts="Dentro do horário de atendimento, ou calendário não existe."}
else
    play {tts="Fora do horário de atendimento"}
end
menu.main ()
end

function FuncoesInternas ()
    checkpoint("Funcoes internas")
    play {tts="Teste das Funções internas do Persona 3. Digite 1, para validações em geral. Digite 2 para ler um arquivo e mandar reproduzir uma linha. Digite 3 para ler um arquivo e reproduzir todo o conteúdo.", cut="true"}
    local M = prompt {}
    menu.route (M, {
        ["1"] = Validacoes,
        ["2"] = LerLinha,
        ["3"] = LerArquivo,
        ["default"] = Erro,
        ["timeout"] = Erro,
    })
end

function Validacoes ()

```



```
checkpoint("Validacoes, CPF, CNPJ, cartao de credito")
play {tts="Teste 1. Verificação da validade de um CPF. Digite o número do CPF.",
cut="true"}
local CPF = prompt {accumulate=11}
if utils.valida_cpf (CPF) == true then
  play {tts="CPF válido."}
else
  play {tts="CPF inválido."}
end
play {tts="Teste 2. Verificação de validade de um CNPJ. Digite p número do CNPJ.",
cut="true"}
local CNPJ = prompt {accumulate=14}
if utils.valida_cnpj (CNPJ) == true then
  play {tts="cnpj válido."}
else
  play {tts="cnpj inválido."}
end
play {tts="Teste 3. Digite os 15 primeiros digitos de seu cartão de crédito.", cut="true"}
local Cartao1 = prompt {accumulate=15}
play {tts="O último dígito do seu cartão é " .. utils.luhn_generate(Cartao1)}
play {tts="Teste 4. Digite os 16 dígitos do seu cartão de crédito.", cut="true"}
local Cartao2 = prompt {accumulate=16}
if utils.luhn_validate (Cartao2) == true then
  play {tts="Cartão válido."}
else
```

```
    play {tts="Cartão inválido."}
end

play {tts="Teste 5. Gerador de dígito DAMM. Digite um número com no máximo 19 caracteres.", cut="true"}
local Numero = prompt {accumulate=19}
play {tts="O dígito verificador é " .. utils.damm_generate (Numero)}
play {tts="Tete 6. Verificador por dígito DAMM. Digite um número com no máximo 20 caracteres.", cut="true"}
local Numero2 = prompt {accumulate=20}
if utils.damm_validate (Numero2) == true then
    play {tts="Código correto."}
else
    play {tts="Código incorreto."}
end
play {tts="Sua chamada será finalizada."}
end

function LerArquivo ()
    checkpoint("Ler Arquivo")
    local arquivo = recover ("Arquivo2")
    local texto = utils.file2string(arquivo)
    play {tts=texto}
    menu.previous(1)
end

function LerLinha()
```

```
checkpoint("Ler Linha")
local arquivo = recover ("Arquivo")
local texto = utilits.readfile (arquivo)
for k,v in ipairs (texto) do
  play {tts=v}
end
menu.previous(1)
end
```

ANEXO A – EMISSÃO DE RELATÓRIOS PERSONA 3

Os relatórios do **Persona 3** são emitidos com base em parâmetros de pesquisa configurados pelo usuário.

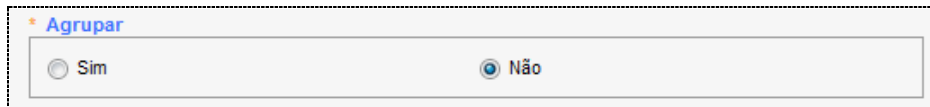
Cada parâmetro corresponde a dados específicos armazenados no sistema, e o resultado da pesquisa está condicionado à existência desses dados.

ABA PARÂMETROS

A aba Parâmetros apresenta opções de pesquisa do **Persona 3**. A seguir são apresentados os parâmetros, em ordem alfabética, referentes a essa aba.

Agrupar

Este parâmetro permite configurar se as chamadas virão em uma listagem ou se serão agrupadas.



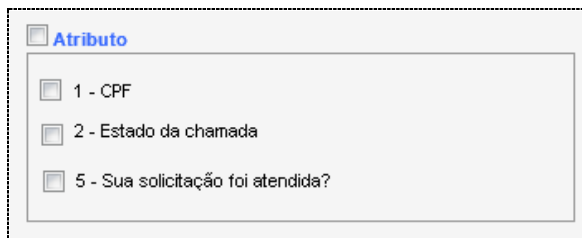
Agrupar

Sim Não

Figura 85. Parâmetro Agrupar por Grupo

Atributo

Este parâmetro permite selecionar quais atributos devem ser exibidos no relatório.



The image shows a configuration window titled 'Atributo'. It contains a list of three items, each with a checkbox to its left:

- 1 - CPF
- 2 - Estado da chamada
- 5 - Sua solicitação foi atendida?

Figura 86. Parâmetro Atributo

Classificação

Este parâmetro permite selecionar a classificação que será exibida no relatório.



The image shows a configuration window titled 'Classificação'. It contains a list of one item with a radio button to its left:

- 5 - Tipo_Testes

Figura 87. Parâmetro Classificação

Classificado Como

Este parâmetro permite selecionar os valores possíveis para uma determinada classificação, selecionada anteriormente no parâmetro **Classificação**. Para isso, basta marcar a caixa de seleção ao lado de “Classificado Como” para habilitar o parâmetro e selecionar os valores possíveis na lista que é apresentada.



Figura 88. Parâmetro Classificado Como

Colunas de resultado de envio de mensagens HSM – Exportação

Neste parâmetro,

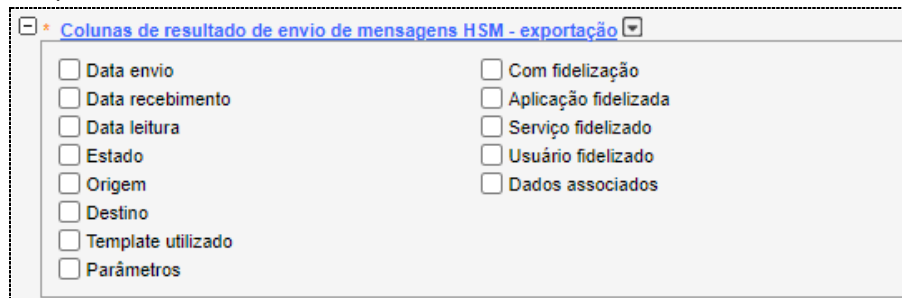



Figura 89. Colunas de resultado de envio de mensagens HSM - Exportação

Contabilizar apenas último ponto de verificação

Neste parâmetro, obrigatoriamente deverá ser indicado se será contabilizado apenas o último ponto de verificação. Caso seja selecionada a opção Não, todos os pontos de verificação serão exibidos no relatório.



* Contabilizar apenas o último ponto de verificação

Sim Não

Figura 90. Parâmetro Contabilizar apenas último ponto de verificação

Detalhes da exportação CSV

Inclui os parâmetros utilizados.



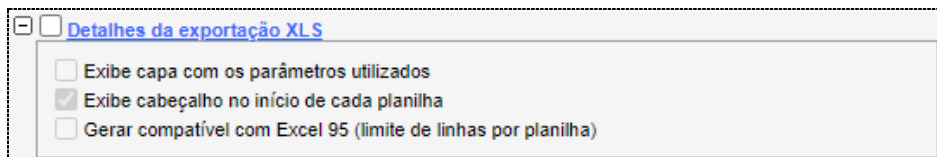
Detalhes da exportação CSV

Incluir os parâmetros utilizados

Figura 91. Detalhes da exportação CSV

Detalhes da exportação XLS

Parâmetro opcional que permite que o usuário realize algumas customizações nos relatórios gerados no formato .XLS:



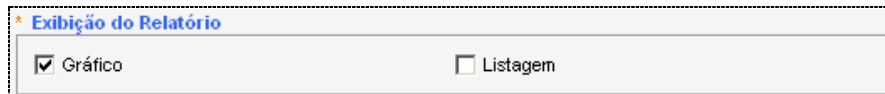
The screenshot shows a dialog box titled "Detalhes da exportação XLS" with a close button in the top-left corner. Inside the dialog, there are three checkboxes:

- Exibe capa com os parâmetros utilizados
- Exibe cabeçalho no início de cada planilha
- Gerar compatível com Excel 95 (limite de linhas por planilha)

Figura 92. Detalhes da exportação XLS

Exibição do relatório

Permite definir se deve ser exibido somente o gráfico, somente a listagem ou ambos no relatório.



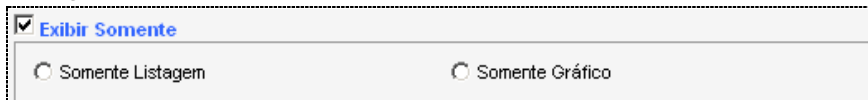
The screenshot shows a dialog box titled "Exibição do Relatório" with a close button in the top-left corner. Inside the dialog, there are two checkboxes:

- Gráfico
- Listagem

Figura 93. Parâmetro Exibição do relatório

Exibir Somente

Este parâmetro permite configurar o relatório para exibir somente a listagem das chamadas ou o gráfico das chamadas.



The screenshot shows a dialog box titled "Exibir Somente" with a close button in the top-left corner. Inside the dialog, there are two radio buttons:

- Somente Listagem
- Somente Gráfico

Figura 94. Parâmetro Exibir Somente

Gerar totalizador de chamadas por ponto de verificação

Este parâmetro permite definir se o relatório exibirá um totalizador de chamadas por ponto de verificação ou não.



* Gerar totalizador de chamadas por ponto de verificação

Sim Não

Figura 95. Parâmetro Gerar totalizador de chamadas por ponto de verificação

Grupos

Este parâmetro permite selecionar os grupos responsáveis por realizar atendimento das chamadas de URA, que serão exibidos no relatório.



* Grupos

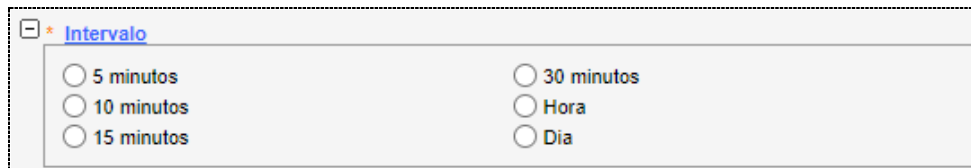
2105 - Persona 2107 - Persona_DDR

2106 - PersonaB

Figura 96. Parâmetro Grupos

Intervalo

Este parâmetro permite selecionar o intervalo em minutos, hora ou dia para divisão da análise das submídias a serem listadas no relatório.



The screenshot shows a configuration window titled "Intervalo" with a close button (X) in the top-left corner. Inside the window, there are two columns of radio button options. The left column contains "5 minutos", "10 minutos", and "15 minutos". The right column contains "30 minutos", "Hora", and "Dia".

Figura 97. Parâmetro Intervalo

Informações adicionais

Este parâmetro permite incluir informações adicionais caso o serviço de URA seja selecionado.

NOTA

O parâmetro **Informações adicionais** só apresentará opções se o serviço URA ou chatbot do **Persona3** for utilizada a primitiva 'report' com o tipo 0 (geral).

Exemplo: report ("CPF", "00000000000", 0). Neste caso será apresentada a opção CPF como opção do parâmetro na emissão do relatório.



The screenshot shows a configuration window titled "Informações adicionais" with a close button (X) and a checked checkbox in the top-left corner. Inside the window, there are two unchecked checkbox options: "CPF" and "RG".

Figura 98. Informações adicionais

Origem

Este parâmetro permite informar a origem das chamadas a serem listadas no relatório.

Figura 99. Parâmetro Origem

Resultado do Atendimento no Grupo

Este parâmetro permite selecionar os resultados de atendimentos que serão exibidos no relatório.

<input type="checkbox"/> Resultado do Atendimento no Grupo	
<input type="checkbox"/> Atendimento normal	<input type="checkbox"/> Fora do Horário
<input type="checkbox"/> Sem licença	<input type="checkbox"/> Destino Inválido
<input type="checkbox"/> Rota Bloqueada	<input type="checkbox"/> Origem Inválida
<input type="checkbox"/> Limite de Tráfego	<input type="checkbox"/> Erro de Script

Figura 100. Parâmetro Resultado do Atendimento no Grupo

Robôs

Este parâmetro permite filtrar as chamadas de chat pelos robôs desejados.

The screenshot shows a configuration window for 'Robôs'. It contains a sub-section with two input fields: 'TestePBO' and 'teste01230', each with an unchecked checkbox.

Figura 101. Parâmetro Robôs

Separador de Campos para Arquivo CSV

Este parâmetro permite definir o tipo de separador que deverá ser utilizado entre as colunas do relatório gerado no formato .CSV.

The screenshot shows a configuration window for 'Separador de campos para arquivo CSV'. It contains three radio button options: 'Ponto-e-vírgula (;)', 'Vírgula (,)', and 'Pipe (|)'. The 'Ponto-e-vírgula (;)' option is selected.

Figura 102. Separador de Campos para Arquivo .CSV

Serviço de URA

Este parâmetro permite definir qual serviço de URA será exibido no relatório.

The screenshot shows a configuration window for 'Serviço de URA'. It contains two radio button options: 'Rapido' and 'web2'. The 'Rapido' option is selected.

Figura 103. Parâmetro Serviço de URA

Submídia

Este parâmetro permite selecionar a submídia para a visualização dos dados referentes às submídias de chat.


A imagem mostra um formulário de configuração com um campo de seleção rotulado "Submídia". O campo contém um ícone de menu suspenso, um ícone de caixa de seleção vazia e o texto "Submídia".

Figura 104. Parâmetro Submídia

Submídia WhatsApp

Este parâmetro permite selecionar uma ou mais submídias para a visualização dos dados referentes às submídias de chat.

A imagem mostra um formulário de configuração com um campo de seleção rotulado "Submídia WhatsApp". O campo contém um ícone de menu suspenso, um ícone de caixa de seleção marcada com um checkmark azul, e o texto "Submídia WhatsApp". Abaixo, há uma lista de opções com ícones de caixa de seleção vazias: "Infobip Oficial Dígitro", "whatsapp oficial" e "Zenvia oficial".

Figura 105. Submídia WhatsApp

Telefone

Este parâmetro permite adicionar somente dígitos.



Figura 106. Telefone

Template HSM

Este parâmetro permite selecionar os templates.

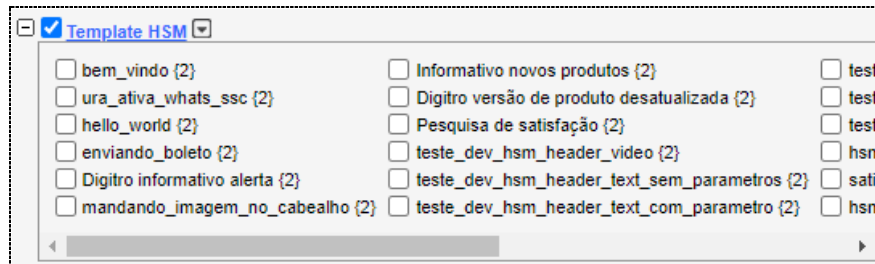
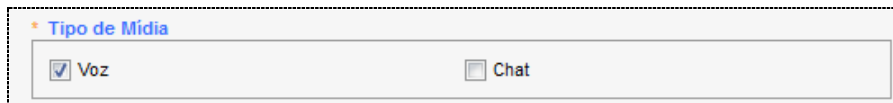


Figura 107. Template HSM

Tipo de Mídia

Este parâmetro permite definir quais chamadas serão apresentadas no relatório em função do tipo de mídia: voz, chat ou ambos.

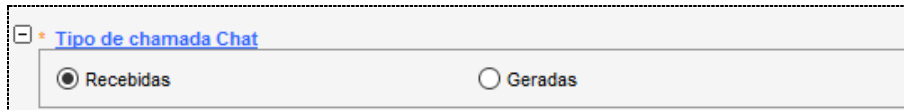


The screenshot shows a configuration box titled '* Tipo de Mídia'. Inside the box, there are two radio button options: 'Voz' (checked) and 'Chat' (unchecked).

Figura 108. Parâmetro Tipo de Mídia

Tipo de Chamada de Chat

Este parâmetro permite filtrar as chamadas de chat por Recebidas e Geradas.

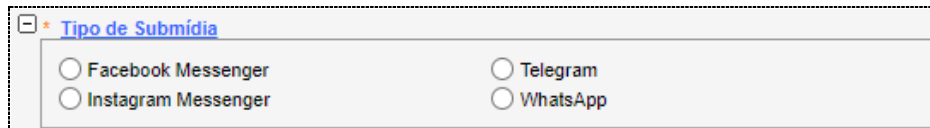


The screenshot shows a configuration box titled '* Tipo de chamada Chat'. Inside the box, there are two radio button options: 'Recebidas' (checked) and 'Geradas' (unchecked).

Figura 109. Parâmetro Tipo de chamada Chat

Tipo de Submídia

Este parâmetro permite selecionar o tipo de submídia.



The screenshot shows a configuration box titled '* Tipo de Submídia'. Inside the box, there are four radio button options: 'Facebook Messenger', 'Instagram Messenger', 'Telegram', and 'WhatsApp', all of which are currently unselected.

Figura 110. Parâmetro Tipo de Submídia

MÓDULO PERSONA 3

Este módulo compreende os modelos de relatórios relativos a Mídias Sociais e ao sistema de URA ou URA VXML, e permitem a análise de dados dos serviços de URA/VXML ativados na plataforma e das chamadas que passaram por esses serviços.

Modelo	Título	Utilização
7000	Integração Mídias Sociais - Estatísticas de Mensagens	Análise das Mídias Sociais da plataforma.
7001	Integração Mídias Sociais - Resultado do envio de mensagens HSM	
7501	Integração Mídias Sociais - Resultado do envio de mensagens HSM - Exportação	
Modelo	Título	Utilização
30601	URA – Chamadas, Totalizador de Acessos por Ponto de Verificação	Análise das URAs da plataforma.
30602	URA – Exportação da Listagem dos Pontos de Verificação	
30603	URA – Chamadas, Totalizador de Chamadas no Grupo por Serviço	
30604	URA – Chamadas, Informações da Chamada no Serviço	
30605	URA – Chamadas, Gráfico das Chamadas por Hora	
30606	URA – Chamadas, Gráfico das Chamadas por Dia	
30607	URA – Agentes, Relatório de Pesquisa de Satisfação	

30608	URA – Chamadas, Totalizadores por Classificação da Chamada	
30609	URA – Chamadas, Totalizadas por atributos valores do serviço	
30610	URA – Lista Chamadas do Grupo com resultado	

Modelo 7000 - Integração Mídias Sociais - Estatísticas de Mensagens

Parâmetros de emissão: na aba Emissão/Programação, defina o Intervalo de Análise e, na aba Parâmetros, defina o Intervalo, Tipos de Submídia, Submídia e o Telefone.

Relatório emitido: exibe as seguintes informações:

- Horário.
- Mensagens enviadas.
- Mensagens recebidas.
- Arquivos enviados.
- Arquivos recebidos.

Digitro Digitro Tecnologia
Rua Professora Sofia Quint da Souza, 187
Fátima/Oeste - SC - Brasil CEP 88085-040
DDD: (47) 339 4111

7000 - Integração Mídias Sociais - Estatísticas de Mensagens Intervalo: Hora

Período: 10/05/2022 00:00 até 11/05/2022 23:59
Tipo de Substância: WhatsApp
Substância: #b0b0 Oficial Digitro

Horário	Mensagens enviadas	Mensagens recebidas	Arquivos enviados	Arquivos recebidos
10:00 - 10:30	1	0	0	0
11:00 - 11:30	3	0	0	0
13:00 - 13:30	2	0	0	0
Total	6	0	0	0

Digitro Emitido em 11/05/2022 16:42:09 Página 1 de 1

Figura 111. Modelo de Relatório 7000

Modelo 7001 - Integração Mídias Sociais - Resultado do envio de mensagens HSM

Parâmetros de emissão: na aba Emissão/Programação, defina o Intervalo de Análise e, na aba Parâmetros, defina a Orientação da página para impressão, Colunas de resultado de envio de mensagens HSM, Submídia WhatsApp, Telefone e o Template HSM.

Relatório emitido: exibe as seguintes informações:

- Data de envio.
- Data de recebimento.
- Data leitura.
- Estado.
- Origem: é o número de telefone da submídia whatsapp que enviou a mensagem.
- Destino da mensagem: número do telefone.
- Template utilizado: id do template utilizado.
- Parâmetros enviados: string com os parâmetros do template.
- Com fidelização.
- Aplicação fidelizada: texto (String).
- Serviço fidelizado: texto (String).
- Usuário fidelizado: texto (String).

Digitro Digitro Tecnologia
Rua Professora Sofia Couti de Souza, 167
Florianópolis - SC - Brasil CEP 88080-040
SSC 0300 789 8111

7001 - Integração Mídias Sociais - Resultado do envio de mensagens HSM

Período: 08/05/2023 00:00 até 14/05/2023 23:59

Telefone:

Submídia: Todos(as)

Template HSM: bem_vindo , Digito Informativo alerta , Digito envio de produto desatualizado , digtro_informativo_de_atualizacao , botao_recabimento_fature

Data envio	Data recebimento	Data leitura	Estado	Origem	Destino	Template utilizado	Parâmetros	Com destaque	Aplicação destaque	Serviço destaque	Localidade destaque
08/05/2023 10:37:30	08/05/2023 10:37:45	08/05/2023 10:41:11	Sucesso	0548908346777	0548094293077	bem_vindo		Não			
10/05/2023 10:26:56	10/05/2023 10:26:31	10/05/2023 10:31:26	Sucesso	0548908346777	0548094293077	digtro_informativo_de_atualizacao	{**"phone":**"7160"}]	Sim	Interact	9_phone	
10/05/2023 11:30:20			Falha	0548908346777	0548090800234	bem_vindo		Sim	Interact		
10/05/2023 11:31:09	10/05/2023 11:31:15	10/05/2023 11:32:05	Sucesso	0548908346777	0548090800234	bem_vindo		Sim	Interact	LeiteRecap	
10/05/2023 11:30:19			Falha	0548908346777	0548090800234	botao_recabimento_fatura	{"responseButton":"test","responseButton":"test"}	Não			
10/05/2023 11:31:19	10/05/2023 11:31:23	10/05/2023 11:31:48	Sucesso	0548908346777	0548090800234	botao_recabimento_fatura	{"responseButton":"test","responseButton":"test"}	Sim	Interact	LeiteRecap	
10/05/2023 13:40:10	10/05/2023 13:39:14	10/05/2023 13:42:48	Sucesso	0548908346777	0548090800234	bem_vindo		Sim	Interact	LeiteRecap	
10/05/2023 13:40:01	10/05/2023 13:40:06	10/05/2023 13:42:48	Sucesso	0548908346777	0548090800234	digtro_informativo_de_atualizacao	{**"test1":**"test2"}]	Não			
Total Geral : 8											

Figura 112. Modelo de Relatório 7001

Modelo 7501 - Integração Mídias Sociais - Resultado do envio de mensagens HSM – Exportação

Parâmetros de emissão: na aba Emissão/Programação, defina o Intervalo de Análise e, na aba Parâmetros, defina Colunas de resultado de envio de mensagens HSM- exportação, Submídia WhatsApp, Telefone, Detalhes da exportação XLS, Separador de campos para arquivo CSV, Detalhes da exportação CSV e o Template HSM.

Relatório emitido: exibe as seguintes informações:

- Data de envio.

Emissão de Relatórios
ANEXO A

- Data de recebimento.
- Data leitura.
- Estado.
- Origem: é o número de telefone da submídia whatsapp que enviou a mensagem.
- Destino da mensagem: número do telefone.
- Template utilizado: id do template utilizado.
- Parâmetros enviados: string com os parâmetros do template.
- Com fidelização.
- Aplicação fidelizada: texto (String).
- Serviço fidelizado: texto (String).
- Usuário fidelizado: texto (String).

	A	B	C	D	E	F	G	H	I	J	K	L
1	Data envio	Data recebimento	Data leitura	Estado	Origem	Destino	Template utilizado	Parâmetros	Com fidelização	Aplicação fidelizada	Serviço fidelizado	Usuário fidel
2	2023-05-08 15:37:35	2023-05-08 15:37:45	2023-05-08 15:41:11	Sucesso	5548996346777	55489984293077	bem_vindo		Não			
3	2023-05-10 11:30:20			Falha	5548996346777	5548999890934	bem_vindo		Sim	Interact		
4	2023-05-10 11:31:09	2023-05-10 11:31:15	2023-05-10 11:32:05	Sucesso	5548996346777	5548999890934	bem_vindo		Sim	Interact	LeiaRecep	
5	2023-05-10 13:58:10	2023-05-10 13:58:14	2023-05-10 19:13:49	Sucesso	5548996346777	554899890934	bem_vindo		Sim	Interact	LeiaRecep	
6												

Figura 113. Modelo de Relatório 7501

Modelo 30601 – URA – Chamadas, Totalizador de Acessos por Ponto de Verificação

Parâmetros de emissão: na aba Emissão/Programação, defina o Intervalo de Análise e, na aba Parâmetros, defina o Serviço, Tipos de Mídia e se deverá ser listado apenas os Últimos Pontos de Verificação* acessados pela chamada. Se preferir, defina uma Classi-

ficação, os possíveis valores no parâmetro Classificado Como e o Tipo de Mídia, o qual pode ser Voz ou Chat. Ao escolher apenas a opção Chat, abre-se o parâmetro Tipo de Chamada de Chat, que permite optar pelas chamadas recebidas ou geradas.

Relatório emitido: exibe, para cada Serviço e Ponto de Verificação (código e descrição), as seguintes informações:

- Ponto de verificação*:
 - Descrição: nome atribuído ao ponto de verificação.
- Acessos: quantidade de acessos ao ponto de verificação e percentual em relação ao total de acessos ao Serviço.
- Chamadas: quantidade de chamadas que passaram pelo ponto de verificação e percentual em relação ao total de chamadas.
- Tempos:
 - Médio: tempo médio de duração das chamadas no ponto de verificação.
 - Máximo: tempo máximo de duração das chamadas no ponto de verificação.
- Total de acessos ao serviço.
- Total de chamadas ao serviço.

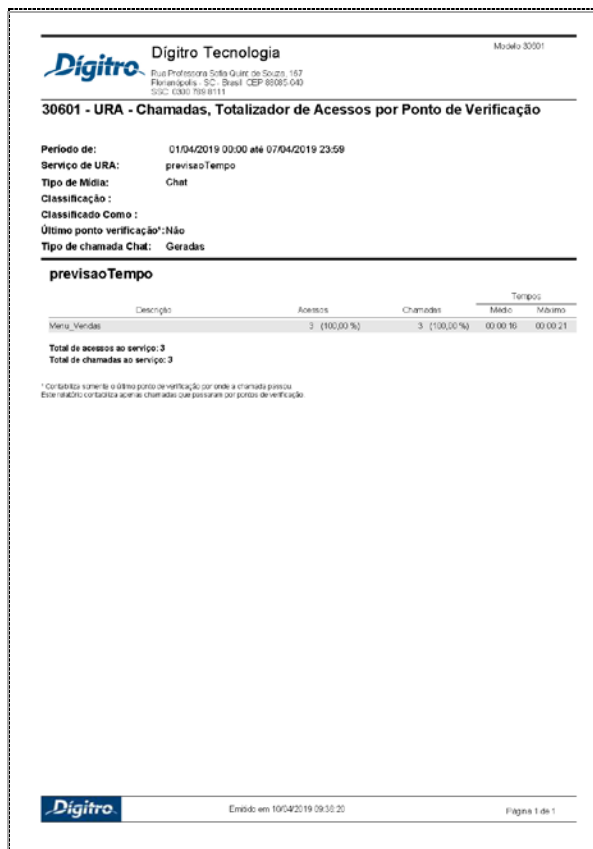


Figura 114. Modelo de Relatório 30601

Modelo 30602 – URA – Chamadas, Exportação da Listagem dos Pontos de Verificação

Parâmetros de emissão: na aba Emissão/Programação, defina o Intervalo de Análise e, na aba Parâmetros, defina o Serviço, Tipos de Mídia e se deverá ser listado apenas os Últimos Pontos de Verificação* acessados pela chamada. Se preferir, defina uma Classificação, os possíveis valores no parâmetro Classificado Como e o Tipo de Mídia, o qual pode ser Voz e/ou Chat. Ao escolher apenas a opção Chat, abre-se o parâmetro Tipo de Chamada de Chat, que permite optar pelas chamadas recebidas ou geradas. Informações adicionais, que permite incluir informações adicionais, caso o serviço de URA seja selecionado.

Relatório emitido: são exibidas as chamadas recebidas pelo serviço de URA e os pontos de verificação por onde elas passaram. Na listagem, são apresentadas as seguintes informações:

- Data: data de entrada da chamada na plataforma
- Telefone: número de origem da chamada
- Data de recebimento no serviço: data em que a chamada entrou no serviço de URA.
- Desligamento após: tempo da chamada em segundos.
- Por: quem desligou a chamada (A* ou B*).
- Data fim na plataforma.
- Data fim no serviço de URA.
- Ponto de verificação: código e descrição do ponto de verificação.
- Data de recebimento no ponto: data/hora em que o ponto de verificação foi acessado.
- Mídia: mídia da chamada (Voz ou Chat).

O totalizador geral exibe, para cada ponto de verificação, o total de chamadas que o acessaram.

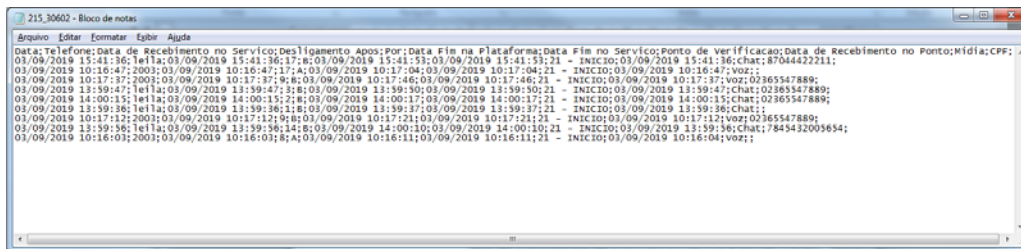


Figura 115. Modelo de Relatório 30602

Modelo 30603 – URA – Chamadas, Totalizador de Chamadas no Grupo por Serviço

Parâmetros de emissão: na aba Emissão/Programação, defina o Intervalo de Análise e, na aba Parâmetros, os Serviços pelos quais as chamadas passaram, os Tipos de Mídias desejados o qual pode ser Voz e/ou Chat. Ao escolher apenas a opção Chat, abre-se o parâmetro Tipo de Chamada de Chat, que permite optar pelas chamadas recebidas ou geradas.

Relatório emitido: exibe um totalizador das chamadas recebidas pelo(s) Serviço(s) de URA selecionado(s) e os seguintes dados:

- Chamadas recebidas: contabiliza chamadas reentrantes. Por exemplo, a chamada foi atendida pelo Serviço A, transferida para o B e novamente transferida para A (reentrante em A).
- Chamadas externas recebidas pelo serviço.

- Chamadas saintes transferidas para o serviço.
- Chamadas internas.
- Chamadas atendidas.
- Tempo médio de atendimento.
- Chamadas transferidas pelo serviço.
- Chamadas abandonadas.
- Chamadas bloqueadas.

Digitro Tecnologia		Módulo: 30603
Rua Professorª Sônia Quad de Souza, 157 Fátimaópolis - SC - Brasil CEP: 86095-560 SQC 0203 199-9111		
30603 - URA - Chamadas, Totalizador de Chamadas por Serviço		
Período de:	01/04/2019 00:00 até 07/04/2019 23:59	
Serviço de URA:	previsão Tempo	
Tipo de mídia:	Chat	
Tipo de chamada Chat:	Geradas	
Geradas	67	
Chamadas Atendidas	67 (9% de 2)	
Tempo Médio Atendimento	00:00:16	
Chamadas transferidas pelo serviço	0 (0% de 2)	
Chamadas Abandonadas	0 (0% de 2)	
Chamadas Bloqueadas	0 (0% de 2)	

Digitro

Em: do em 16/04/2019 10:45:35

Página 1 de 1

Figura 116. Modelo Relatório 30603

Modelo 30604 – URA – Chamadas, Informações da Chamada no Serviço, Ponto de Verificação

Parâmetros de emissão: na aba Emissão/Programação, defina o Intervalo de Análise e, na aba Parâmetros, defina o Serviço, Tipos de Mídia e se deverá ser listado apenas os Últimos Pontos de Verificação* acessados pela chamada. Se preferir, defina uma Classificação, os possíveis valores no parâmetro Classificado Como e o Tipo de Mídia, o qual pode ser Voz e/ou Chat. Ao escolher apenas a opção Chat, abre-se o parâmetro Tipo de Chamada de Chat, que permite optar pelas chamadas recebidas ou geradas.

Relatório emitido: apresenta informações detalhadas das chamadas recebidas pelo serviço de URA, conforme descrito a seguir:

- Informações apresentadas no relatório para cada chamada:
 - Data inicial da chamada.
 - Mídia: mídia da chamada (Voz ou Chat).
 - Tipo: Entrante, Sainte ou Interna.
 - Origem: número de origem da chamada.
 - Juntor*: número do juntor alocado para a chamada.
 - Ocupação: tempo de ocupação da chamada.
 - Destino: número de destino da chamada.
 - Serviço original: cifra de origem da chamada.
 - DDD.
 - Desligado por*: quem desligou a chamada.
- Informações apresentadas no relatório para cada serviço:
 - Data de recebimento: data em que a chamada entrou no serviço de URA.
 - Tempo total: tempo em que a chamada ficou no serviço de URA.

- Resultado atendimento URA: indica o resultado do atendimento da chamada pela URA. Pode ou não identificar o número que originou (#A) e o número que recebeu (#B) a chamada, dependendo da disponibilidade dos mesmos a partir da central.
 - Informações apresentadas no relatório para cada ponto de verificação:
 - Descrição.
 - Data de acesso.
 - Tempo em que a chamada permaneceu no ponto de verificação.
 - Apresenta também os atributos e valores associados à chamada.

Digitro		Digitro Tecnologia	Modelo: 30604
Rua Professora Sônia Guimarães, 167 Pitangueiras - SC - Brasil CEP 88085-040 SSC: 0300.759.8111			
30604 - URA - Chamadas, Informações da Chamada no Serviço, Ponto de Verificação			
Período de:	01/04/2019 00:00:00 até 07/04/2019 23:59		
Serviço de URA:	previsãoTempo		
Tipo de Mídia:	Chat		
Origem:			
Classificado Como:	2 Mip		
Classificação:	4 Tipo_Cliente		
Tipo de chamada Chat:	Gerados		
Chamada:			
Tipo:	Ocupação:	00:00:00	Desligado por:
Origem:	Destino:		
Serviço:			
Data de recebimento:			
Tempo Total:	00:00:00		
Resultado Atendimento URA:			
<hr/>			
Atividade	YMSR		
<hr/>			
Digitro		Emissão em 10/04/2019 05:39:33	Página 1 de 1

Figura 117. Modelo Relatório 30604

Modelo 30605 – URA – Chamadas, Gráfico das Chamadas por Hora

Parâmetros de emissão: na aba Emissão/Programação, defina o Intervalo de Análise e, na aba Parâmetros, defina o Serviço, Tipos de Mídia e se deverá ser listado apenas os Últimos Pontos de Verificação* acessados pela chamada. Se preferir, defina uma Classificação, os possíveis valores no parâmetro Classificado Como e o Tipo de Mídia, o qual pode ser Voz e/ou Chat. Ao escolher apenas a opção Chat, abre-se o parâmetro Tipo de Chamada de Chat, que permite optar pelas chamadas recebidas ou geradas.

Relatório emitido: apresenta um gráfico de linha com as seguintes informações agrupadas por intervalo de tempo:

- Quantidade de chamadas recebidas.
- Quantidade de chamadas atendidas.
- Quantidade de chamadas transferidas.
- Quantidade de chamadas abandonadas.

Apresenta também uma listagem com as informações abaixo, agrupadas de acordo com o parâmetro "Intervalo".

- Quantidade de chamadas recebidas.
- Quantidade de chamadas atendidas e a porcentagem dessas em relação às chamadas recebidas.
- TMA: Tempo Médio de Atendimento*.
- Quantidade de chamadas abandonadas e a porcentagem dessas em relação às chamadas recebidas.

- Quantidade de chamadas transferidas e a porcentagem dessas em relação às chamadas recebidas.

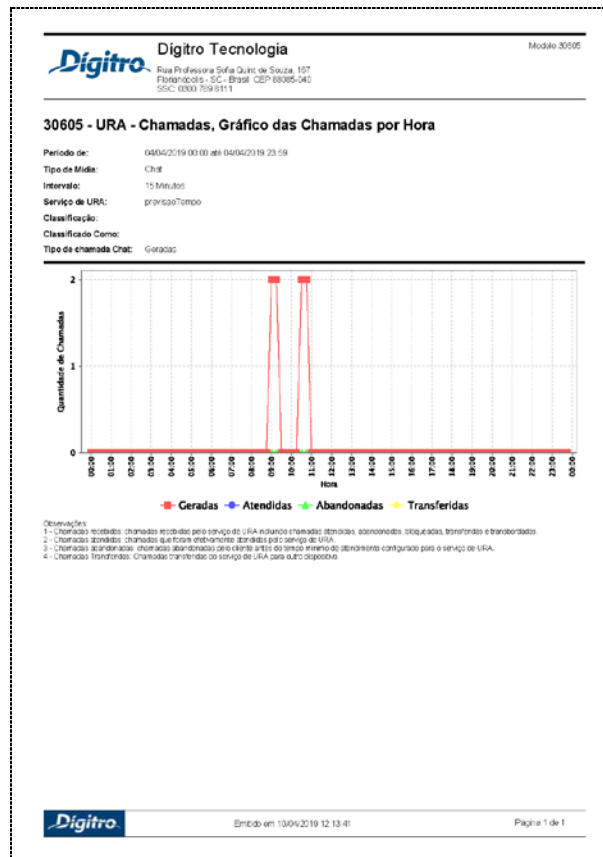


Figura 118. Modelo de Relatório 30605

Modelo 30606 – URA – Chamadas, Gráfico das Chamadas por Dia

Parâmetros de emissão: na aba Emissão/Programação, defina o Intervalo de Análise e, na aba Parâmetros, defina o Serviço, Tipos de Mídia e se deverá ser listado apenas os Últimos Pontos de Verificação* acessados pela chamada. Se preferir, defina uma Classificação, os possíveis valores no parâmetro Classificado Como e o Tipo de Mídia, o qual pode ser Voz e/ou Chat. Ao escolher apenas a opção Chat, abre-se o parâmetro Tipo de Chamada de Chat, que permite optar pelas chamadas recebidas ou geradas.

Relatório emitido: exibe um gráfico agrupando as informações abaixo por dia.

- Quantidade de chamadas recebidas.
- Quantidade de chamadas atendidas.
- Quantidade de chamadas abandonadas.
- Quantidade de chamadas transferidas.

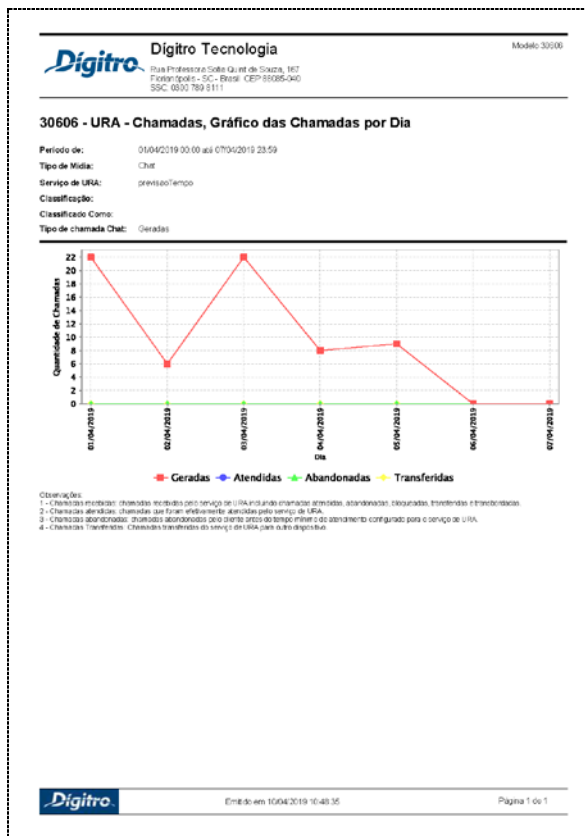


Figura 119. Modelo de Relatório 30606

Modelo 30607 – URA – Agentes, Relatório de Pesquisa de Satisfação

Parâmetros de emissão: na aba Emissão/Programação, defina o Intervalo de Análise e, na aba Parâmetros, os serviços pelos quais as chamadas passaram.

Relatório emitido: apresenta informações referentes à pesquisa de satisfação. Inicialmente, são exibidos gráficos de pizza que consolidam as respostas dadas para cada pergunta. Esses gráficos contabilizam todas as perguntas referentes aos agentes que são exibidos no relatório.

Posteriormente, é exibido um totalizador para cada pergunta realizada para cada agente.

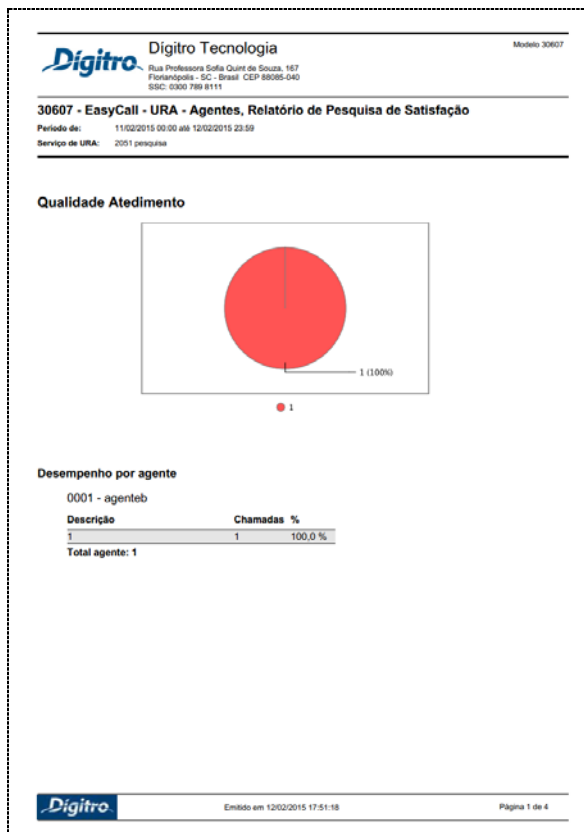


Figura 120. Modelo de Relatório 30607 1

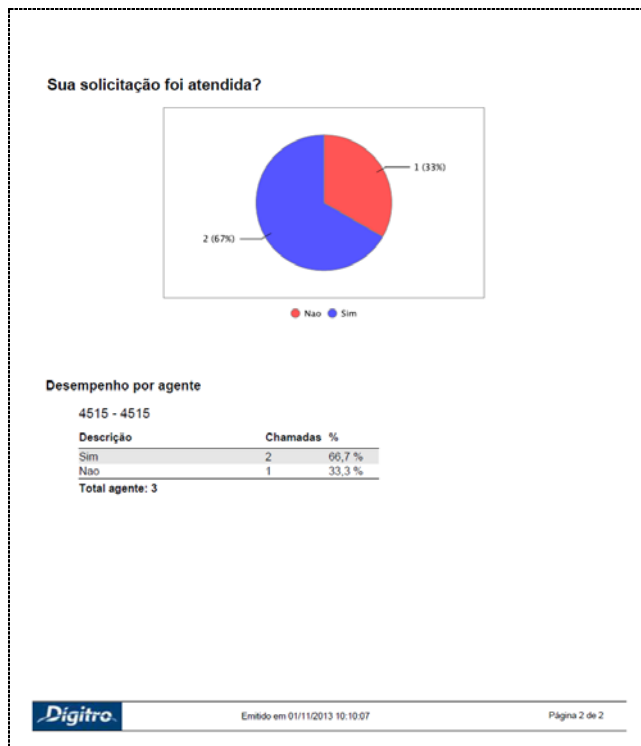


Figura 121. Modelo de Relatório 30607 - 2

Modelo 30608 – URA – Chamadas, Totalizadores por Classificação da Chamada

Parâmetros de emissão: na aba Emissão/Programação, defina o Intervalo de Análise e, na aba Parâmetros, os Tipos de Mídia e os Serviços pelos quais as chamadas passaram o Número de origem desejado e, se preferir, selecione uma Classificação e, para essa, os possíveis valores no parâmetro Classificado Como. Se preferir, defina uma Classificação, os possíveis valores no parâmetro Classificado Como e o Tipo de Mídia, o qual pode ser Voz e/ou Chat. Ao escolher apenas a opção Chat, abre-se o parâmetro Tipo de Chamada de Chat, que permite optar pelas chamadas recebidas ou geradas.

Relatório emitido: exibe as informações abaixo agrupadas por Serviço e Classificação:

- DDD da chamada.
- Data inicial e horário da chamada.
- Origem da chamada.
- Destino da chamada.
- Tempo de duração da Chamada.

Digitro		Digitro Tecnologia		Modelo: 30608
Rua Professora Sofia Gust de Souza, 107 Florianópolis - SC, Brasil CEP 88005-040 SSC: 0030 759-0111				
30608 - URA - Chamadas, Totalizadores por Classificação da Chamada				
Período de: 01/04/2019 00:00 até 07/04/2019 23:59				
Serviço de URA: previsaoTempo				
Tipo de Mdia: Chat				
Classificação: 4 Tipo_Cliente				
Classificado				
Tipo de chamada Chat: Recebidas				
Serviço: previsaoTempo				
Classificação: Tipo_Cliente				
Vip				
DDD	Data Inicial	Origem	Destino	Tempo Chamada
	05/04/2019 11:41:01	Almiró Silva	weather	00:00:10
				Total: 1
Total de chamadas por classificação				
Descrição	Chamadas			
Vip	1			
				Total Geral: 1
Digitro		Emisso em 10/04/2019 15:19:55		Página 1 de 1

Figura 122. Modelo de Relatório 30608

Modelo 30609 – URA – Chamadas, Totalizadas por atributos valores do serviço

Parâmetros de emissão: na aba Emissão/Programação, defina o Intervalo de Análise e, na aba Parâmetros, os Tipos de Mídias e os Serviços pelos quais as chamadas passaram, o Número de origem desejado e, se preferir, filtre as informações por Atributo e o Tipo de Mídia, o qual pode ser Voz e/ou Chat. Ao escolher apenas a opção Chat, abre-se o parâmetro Tipo de Chamada de Chat, que permite optar pelas chamadas recebidas ou geradas.

Relatório emitido: exibe as informações abaixo agrupadas por atributo:

- Valor.
- Quantidade.
- Porcentagem em relação ao atributo.
- Porcentagem em relação ao total de atributos.

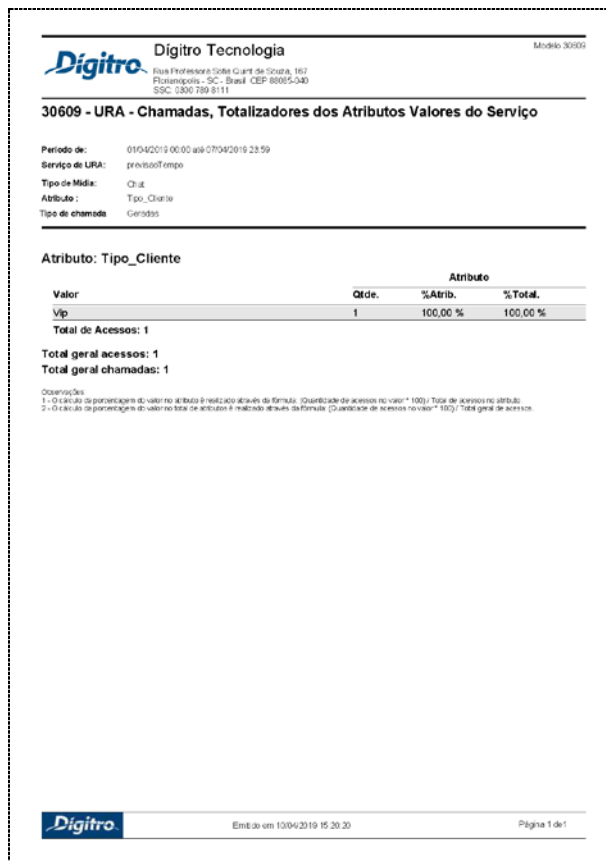


Figura 123. Modelo de Relatório 30609

Modelo 30610 – URA – Lista Chamadas com Resultado

Parâmetros de emissão: na aba Emissão/Programação, defina o Intervalo de Análise e, na aba Parâmetros, o Tipo de Mídia desejado. Para mídia de Voz, defina os Grupos responsáveis pelo atendimento das chamadas de URA. Para mídia de Chat, defina o Tipo de Chamada de Chat, que permite optar pelas chamadas recebidas ou geradas e os robôs responsáveis. Permite agrupar as chamadas por Grupo (mídia de Voz) ou Robô (mídia de Chat), definir quais resultados de atendimento serão exibidos no relatório.

Relatório emitido: são exibidas as chamadas recebidas pelo serviço de URA com seus resultados. Na listagem são apresentadas as seguintes informações:

- **Data Inicial:** data de entrada da chamada na plataforma.
- **Origem:** número de origem da chamada.
- **Destino:** número de destino da chamada.
- **Grupo:** grupo responsável pelo atendimento das chamadas de URA (quando mídia de Voz).
- **Robô:** robô responsável pelo atendimento das chamadas de URA (quando mídia de Chat).
- **Resultado do Atendimento no Grupo:** indica o resultado do atendimento no grupo. Caso o **Persona 3** encontre uma rota para atendimento este campo será preenchido com Atendimento normal e, caso contrário, será preenchido com os seguintes valores:
 - Sem licença.
 - Rota Bloqueada.
 - Limite de Tráfego.
 - Fora do Horário.

- Destino Inválido (somente em média de Voz).
- Origem Inválida (somente em média de Voz).
- Erro de Script .



 Digitro Tecnologia <small>Rua Professora Sofia Quint de Souza, 167 Florianópolis - SC - Brasil CEP 88095-040 DDD: 030 759 9111</small>		Modelo 30610		
30610 - URA - Lista Chamadas com Resultado				
Período de:	01/04/2019 00:00 até 07/04/2019 23:59			
Tipo de Mídia:	Chat			
Robô:	wheather			
Agrupar:	Sim			
Resultado Atendimento:				
Tipo de chamada Chat:	Geradas			
wheather : previsaoTempo				
Data Inicial	Origem	Destino	Robô	Resultado do Atendimento
01/04/2019 09:12:59	Alvaro Schmitt	wheather	wheather	Atendimento normal
01/04/2019 09:13:50	Alvaro Schmitt	wheather	wheather	Atendimento normal
01/04/2019 09:14:40	Alvaro Schmitt	wheather	wheather	Atendimento normal
01/04/2019 09:20:00	Alvaro Schmitt	wheather	wheather	Atendimento normal
01/04/2019 09:20:51	Alvaro Schmitt	wheather	wheather	Atendimento normal
01/04/2019 09:47:00	Alvaro Schmitt	wheather	wheather	Atendimento normal
01/04/2019 09:47:50	Alvaro Schmitt	wheather	wheather	Atendimento normal
01/04/2019 09:59:59	Alvaro Schmitt	wheather	wheather	Atendimento normal
01/04/2019 10:00:49	Alvaro Schmitt	wheather	wheather	Atendimento normal
01/04/2019 10:00:50	Alvaro Schmitt	wheather	wheather	Atendimento normal
01/04/2019 10:02:59	Alvaro Schmitt	wheather	wheather	Atendimento normal
01/04/2019 10:03:39	Alvaro Schmitt	wheather	wheather	Atendimento normal
01/04/2019 10:21:59	Alvaro Schmitt	wheather	wheather	Atendimento normal
01/04/2019 10:22:38	Alvaro Schmitt	wheather	wheather	Atendimento normal
01/04/2019 10:29:59	Alvaro Schmitt	wheather	wheather	Atendimento normal
01/04/2019 10:30:47	Alvaro Schmitt	wheather	wheather	Atendimento normal
01/04/2019 10:30:48	Alvaro Schmitt	wheather	wheather	Atendimento normal
01/04/2019 10:57:00	Alvaro Schmitt	wheather	wheather	Atendimento normal
01/04/2019 10:57:50	Alvaro Schmitt	wheather	wheather	Atendimento normal
		Enviado em 10/04/2019 15:22:08		Página 1 de 2

Figura 124. Modelo de Relatório 30610 – Lista Chamadas com Resultado– Mídia Chat

01/04/2019 10:57:51	Alvaro Schmitt	wheather	wheather	Atendimento normal
		Quantidade	Resultado do Atendimento no Robô	
		20	Atendimento normal	
		20	Total	
		Quantidade	Resultado do Atendimento Geral	
		20	Atendimento normal	
		20	Total	


 Emitido em 10/04/2019 15:22:08 Página 2 de 2

Figura 125. Modelo de Relatório 30610 – Lista Chamadas com Resultado – Mídia Voz

GLOSSÁRIO

Palavra	Descrição
Acessos	Número de vezes em que o valor relacionado foi selecionado dentre todos os valores do atributo relacionado.
Atributo	<p>É o nome do atributo associado à chamada.</p> <p>Por exemplo, uma URA pode definir que suas chamadas terão o atributo 'CPF', o qual poderá assumir valores diferentes a cada chamada.</p>
Chamadas abandonadas	<p>Chamadas abandonadas pelo cliente antes do atendimento pelo agente, ou nos primeiros segundos de atendimento. Pode ser de três tipos:</p> <ul style="list-style-type: none">▪ Abandonadas antes da fila: chamadas abandonadas pelo cliente antes de serem distribuídas pelo DAC (geralmente durante a mensagem de abertura).▪ Abandonadas na fila: chamadas abandonadas pelo cliente durante aguardo de recurso para atendimento.▪ Abandonadas no Ramal (tempo mínimo de abandono configurado no EasyCall Visor): chamadas abandonadas nos primeiros segundos de atendimento.

Glossário

	<p>NOTA</p> <p><i>Há outra situação em que a chamada pode ser classificada pelo sistema como Abandonada no ramal, conforme segue:</i></p> <ul style="list-style-type: none"> - Quando a chamada for desligada pela origem enquanto estiver tocando no ramal do agente (PA). Isto pode ocorrer quando o agente está operando com ramal “No gancho” ou “Fora do gancho manual”. <ul style="list-style-type: none"> ▪ Abandonadas por transferência: ver o item Abandono por transferência.
Chamadas atendidas	Chamadas que são efetivamente atendidas por agentes ou dispositivos/serviços específicos. As chamadas abandonadas no ramal não são contabilizadas como chamadas atendidas.
Chamadas atendidas em até N segundos	Indica o número de chamadas que foram atendidas em até N segundos, onde N é o parâmetro “Tempo médio de espera para cálculo de nível de serviço” definido pelo usuário na configuração do Dispositivo pelo EasyCall Visor.
Chamadas bloqueadas	Uma chamada que não foi completada. O usuário escuta um sinal de ocupado, indicando que todas as linhas estão ocupadas.
Chamadas com atendimento imediato	São chamadas que não ficam em fila de espera antes de serem atendidas.
Chamadas Não DAC	São chamadas que foram geradas/recebidas e que não passaram por nenhum dispositivo DAC, por exemplo: chamadas feitas

Glossário

	entre agentes ou chamadas externas recebidas diretamente no ramal. Quando ocorre a transferência de uma chamada entre os agentes, para o agente que recebeu a chamada transferida também será contabilizado o recebimento de uma chamada não DAC.
Chamadas Recebidas	Número total de chamadas recebidas pelo dispositivo, incluindo chamadas bloqueadas, abandonadas, atendidas e transbordadas. São as chamadas do tipo entrante (recebida externa), internas (recebidas) e também as chamadas saintes transferidas para o dispositivo.
Chamadas Transbordadas	Indica o total de chamadas que foram repassadas de um serviço para outro devido a exceções como, por exemplo, ultrapassar o tempo máximo em fila ou quando não há agente "logado".
Condição Término	<p>Condição da chamada ao finalizar seu tratamento pelo serviço/dispositivo. Possíveis condições de término:</p> <ul style="list-style-type: none"> ▪ Chamada normal. ▪ Chamada derrubada – serviço fora do horário. ▪ Chamada derrubada – serviço inexistente ou inativo. ▪ Chamada derrubada – PA, GRUPO ou OPERADOR indisponível. ▪ Chamada transferida para RAMAL/SERVIÇO - situação fora do horário. ▪ Chamada transferida para RAMAL/SERVICO - situação fila. ▪ Chamada transferida para RAMAL/SERVICO - situação PA/GRUPO/OPERADOR indisponível.

Glossário

	<ul style="list-style-type: none"> ▪ Chamada derrubada – excedeu máximo de instâncias para este serviço. ▪ Chamada Transferida para SERVIÇO – situação geral. ▪ Chamada Derrubada no Pré-Atendimento. ▪ Chamada Derrubada – Bloqueio p/ celular ativo. ▪ Chamada Derrubada – Bloqueio p/ TP ativo. ▪ Chamada Derrubada – Bloqueio DDC (a cobrar) ativo. ▪ Falha na transferência entre serviços. ▪ Falha na operação de recursos. ▪ Erro de processamento. ▪ Falha na geração da chamada.
Data Inicial	Data e horário em que a chamada entrou na plataforma Dígitro.
DDD	<p>Apresenta o código de Discagem Direta à Distância utilizado para realizar a chamada ou da origem da chamada recebida. Casos especiais:</p> <ul style="list-style-type: none"> ▪ 00 = Chamadas internacionais. ▪ 0300, 0500, 0800 e 0900 = correspondem respectivamente aos serviços do tipo 0300, 0500, 0800 e 0900. ▪ Desconhecido = quando a chamada é feita ou recebida por ramal remoto/VoIP ou ainda quando a Central Telefônica não fornece a informação.
Desligamento após	Consulte Duração ou Tempo de Ocupação.

Glossário

Desligado por	Informa o tipo de desligamento da chamada: A = chamada desligada por quem a originou; B = chamada desligada por quem a atendeu; F = falha na identificação; I = não pôde ser determinado quem desligou a chamada; O = EasyCall Memory liberou a chamada sem receber informação da liberação da chamada pela switch.
Destino	Identidade de quem recebe a chamada discada pelo chamador.
Juntor	Número físico do juntor utilizado para atender ou realizar a chamada. Quando se tratar de uma chamada feita de ramal a ramal, o juntor é informado como "interno".
Origem	Identidade de quem realizou a chamada. Esta identidade pode apresentar também o DDD e CSP utilizado pelo número chamador.
Pontos de Verificação	Registros especiais, inseridos no fluxo da URA, que permitem o armazenamento de informações específicas sobre determinados trechos do serviço. Sua função principal é registrar pontos do fluxo por onde a chamada passou.
Resultado	Apresenta os resultados de atendimento ou realização da chamada pelo ramal.
Serviço original	Cifras de roteamento da chamada antes desta ser atendida.
Tempo de fila (por serviço/dispositivo)	Tempo (hh:mm:ss) decorrido do momento de recebimento da chamada pelo serviço/dispositivo ao momento do primeiro atendimento pelo ramal ou PA.

Glossário

Tempo Máximo de Espera (TME) ou de Fila	Tempo máximo de espera na fila antes do atendimento por agente do Call Center (estão contabilizadas nesse tempo as chamadas abandonadas na fila). Para este cálculo só está sendo contabilizado o tempo de fila.
Tempo Médio de Abandono na fila (TMAb)	É o resultado da divisão da soma de todos os tempos de abandono das chamadas abandonadas na fila pela quantidade destas chamadas.
Tempo Médio de Atendimento (TMA)	Tempo (hh:mm:ss) total de atendimento das chamadas atendidas dividido pela quantidade destas chamadas.
Tempo Médio de Espera ou de Fila	É o tempo médio de espera na fila antes do primeiro atendimento por agente do Call Center. É calculado pela soma dos tempos de espera na fila das chamadas recebidas por um dispositivo DAC dividido pela quantidade de chamadas que aguardaram na fila.
Tipo da chamada	Informa o tipo da chamada gerada (ou recebida): <ul style="list-style-type: none">▪ Entrante: gerada por um número externo e recebida por um serviço/ dispositivo da plataforma.▪ Sainte: gerada por um serviço/ dispositivo da plataforma e recebida por um número externo.▪ Interna: gerada por um serviço/ dispositivo e recebida por outro serviço/ dispositivo da plataforma.