
omnileads Documentation

Release develop

omnileads

Oct 04, 2022

Contents

1	How can I get it?	3
2	How can I install it?	5
3	Omnileads Features and Functionalities	7
3.1	WebRTC - OMniLeads underlying technology	7
3.2	OMniLeads Features and Functionalities	7
4	Where and how can I use it?	9
4.1	OMI as a Contact Center integrated to a PBX based on SIP	9
4.2	OML on a Customer Contact services company	10
4.3	OML for carriers o Cloud PBX providers	11
5	¿Cómo me capacito ?	13
6	ARCHITECTURE	15
6.1	Architecture and Components	15
7	INSTALLATION	29
7.1	OMniLeads deploy	29
8	SECURITY	39
8.1	Security Considerations	39
9	INITIAL CONFIGURATIONS	43
9.1	Initial configuration	43
10	PSTN ACCESS CONFIGURATION	57
10.1	Telephony module configuration	57
11	CAMPAIGNS	67
11.1	Telephone Campaigns	67
12	METRICS, REPORTS, RECORDINGS AND SUPERVISION	137
12.1	Metrics, Recordings and Supervision	137
13	AGENTS AUDIT	159
13.1	Backoffice module	159

14 AGENT MANUAL	165
14.1 Agent Manual	165
15 EASY PBX INTEGRATION	209
15.1 Integration between OMniLeads & PBXs	209
16 IT ADMINISTRATOR MANAGEMENT	225
16.1 IT administrator managment	225
17 CRM INTEGRATION	241
17.1 CRM integration	241
18 OML API	255
18.1 OMniLeads RESTful API	255
19 KNOWN ISSUES	269
19.1 Known Issues	269
20 RELEASE NOTES	271
20.1 Release Notes 1.25.0	271

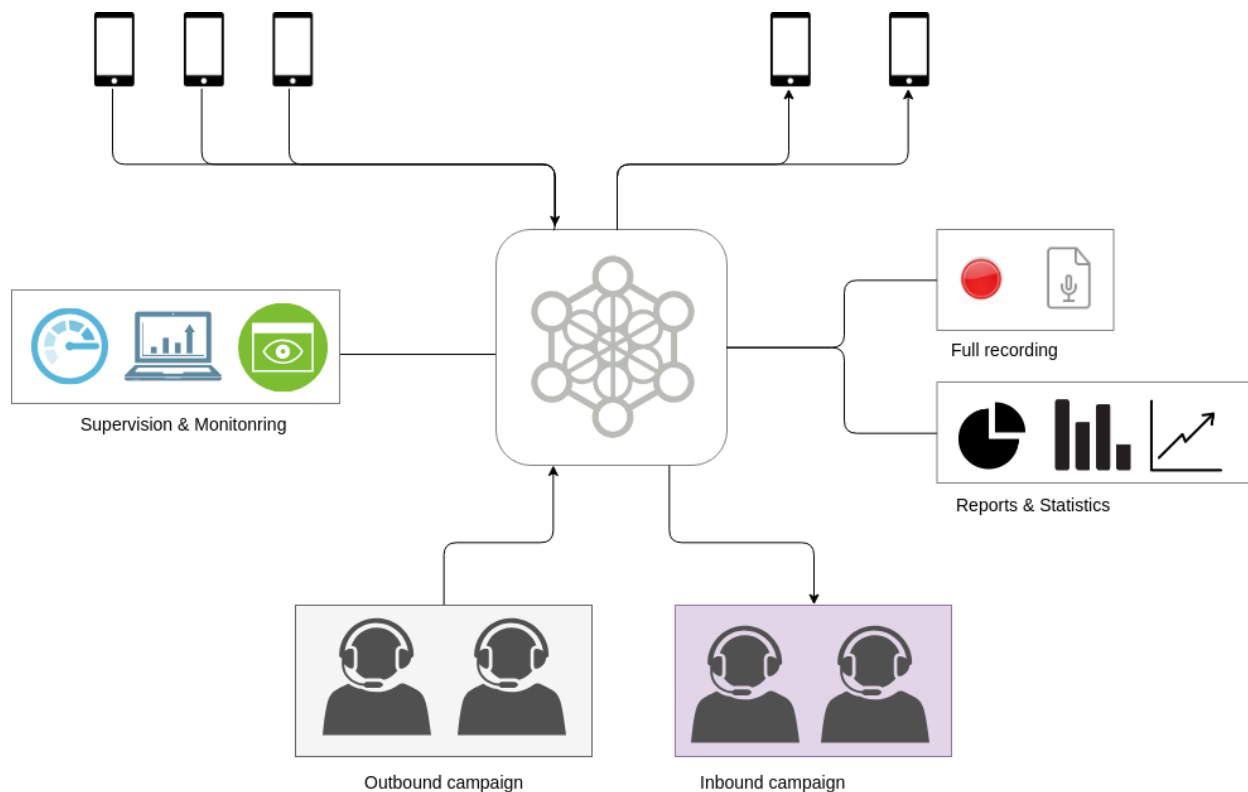
OMniLeads is a Web Application for Contact Centers based on the free software license [GPLV3](#).

This application allows you to implement and manage both incoming and outgoing Contact Center operations and in Blended mode. The App provides metrics, reports and indicators, real-time supervision of agents, audit modules for backoffice and other advanced QA functionalities, contact and campaign management.

The fact of being 100% WebRTC makes it ideal for setting up operations with agents in home-office mode due to the efficiency and cryptographic security that WebRTC technology implies in its default operation when maintaining voice and / or video sessions through from Internet.

The different user profiles (agents, supervisors, administrators or clients) access OMniLeads from any modern browser. As it is not required the use of desktop applications (*softphones*), then it is not necessary to perform any Workstation configurations. So just by accessing the HTTPS web address where the application resides, both agents and supervisors can be online managing communications with customers. This facility implies a great advantage when providing cloud services CCaaS (Contact Center as a Service).

OMniLeads can be adapted to a company or organization that needs to set up your own Contact Center integrated into your PBX, as well as also scale to companies that provide Customer Services (BPO - Business Process Outsourcing), either over on-premise environments as well as cloud deployments.



CHAPTER 1

How can I get it?

The repository is available in [GitLab](#), for free download, installation, modification and use of the Software.

CHAPTER 2

How can I install it?

The section *OMniLeads deploy* talk about this, exposing the steps needed to install the software.

Omnileads Features and Functionalities

3.1 WebRTC - OMniLeads underlying technology

Before to cite use cases, we want to emphasize the WebRTC benefits;the OMniLeads core

WebRTC adds to the browser the ability to perform real-time communications in voice, video, chat and screen sharing. OMniLeads use this technology to nucleate communicationsand the web interface, avoiding desktop apps, “soft-phones”, which gives a fast workflow in creating and registering users. With only a login the users are on-line processing communications.



Another WebRTC advantages

- Workstations failure points are minimized
- Helpdesk tasks are minimized and for that reason IT employees demand also is minimized
- We work with audip Opus and video VP8 codecs, both conceived for a maximun performance on Internet environments (internet native codecs)
- In a security level all communications must travels encrypted, in terms of signaling and media.

3.2 OMniLeads Features and Functionalities

- Inbound and Outbound campaigns management.

- WebRTC agent console (does not requires any installation or plugin,100% Web Browser).
- Opus® as default codec for all Agent communications.
- WebRTC supervision console; detailed report of agent & campaign states.
- Agents and campaigns productivity reports
- Recordings search with filter of date, campaign, agent
- Campaign recycling by agent disposition and/or phone call status.
- Contact database change over the same campaign
- Answering machines detection with audio message play
- CRM / ERP integration trough RestFull API
- Ready for virtualization ! OML was conceived as technology oriented to virtual environments.
- Ready to Dockerize ! OML is available in terms of Docker images, which allows be executed on every OS but also orchestrating clusters Docker HA and/or deployments on Cloud providers(AWS, GCloud, etc.).
- Ready to scale ! Scalability is possible due to reuse underlying technologiesvery powerful like Postgres, Nginx, Rtpengine, that also can runs on independent hosts (horizontal scalability) with mininum setup.
- Complementary addons that gaves to OML extra features and/or vertical segments
- 100% Contact Center oriented. It is not a PBX software with reporting addons and/or supervision. The system was conceived from zero as a una plataform oriented y optimized for Contact Center needs

Where and how can I use it?

4.1 OMI as a Contact Center integrated to a PBX based on SIP

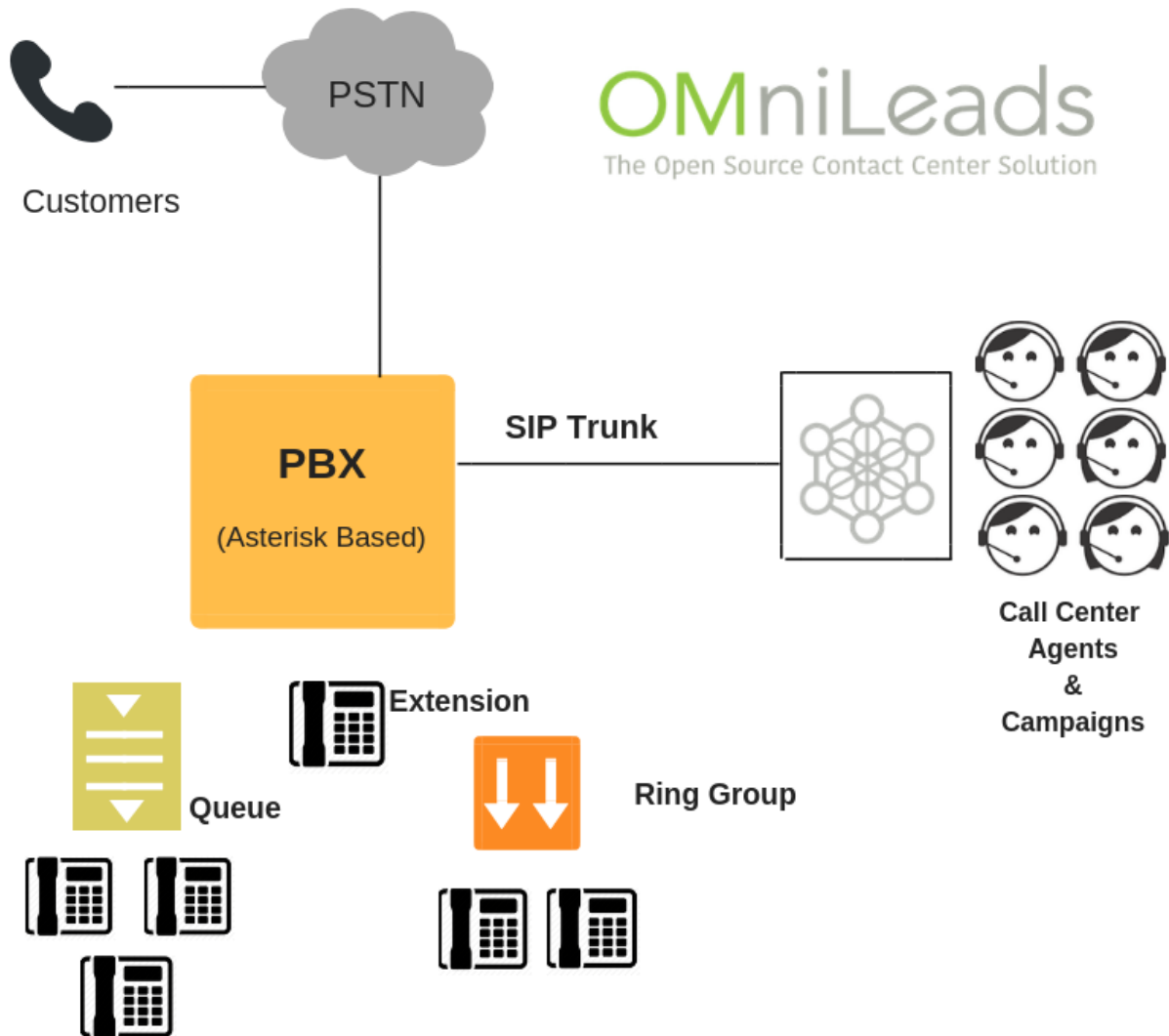
OMniLeads is ideal for companies that demands typical Contact Center features, that PBX systems does not fulfill by its own nature. For that reason OMniLeads emerges as an alternative for complement these PBX system, from an independent instance (bare-metal host, virtual machine or cloud provider) integrated to the PBX, allowing secure, liable and transparent communications.

We propose to expand the traditional paradigm of the adquisition of a reports/supervision software stack to install on the PBX, to in stead of deploy a complete and independen Contact Center application (use its own Asterisk) that allows at the same time a simple integration with the PBX software. This way we could derive an IVR option of the PBX to an OMniLead inbound campaign or realize a transfer from and IVR extension to OMniLeads or viceversa

The more remarkable advantages are:

- Avoid the economic costs that involves software licenses of the typical complementing tools of the market used for add to the PBX some queues reporting and supervision features.
- Avoid the cost in terms of performance of the *PBX*, sacrificed in order to run complex reporting and monitoring tools that implies execute a «call center modules» over the PBX system.

On operations where exists a huge reports extraction or is needed to scale in terms of agents, is quite simple deploy OMniLeads *out of the box*; on a VM, VPS or server without loose the PBX integration.

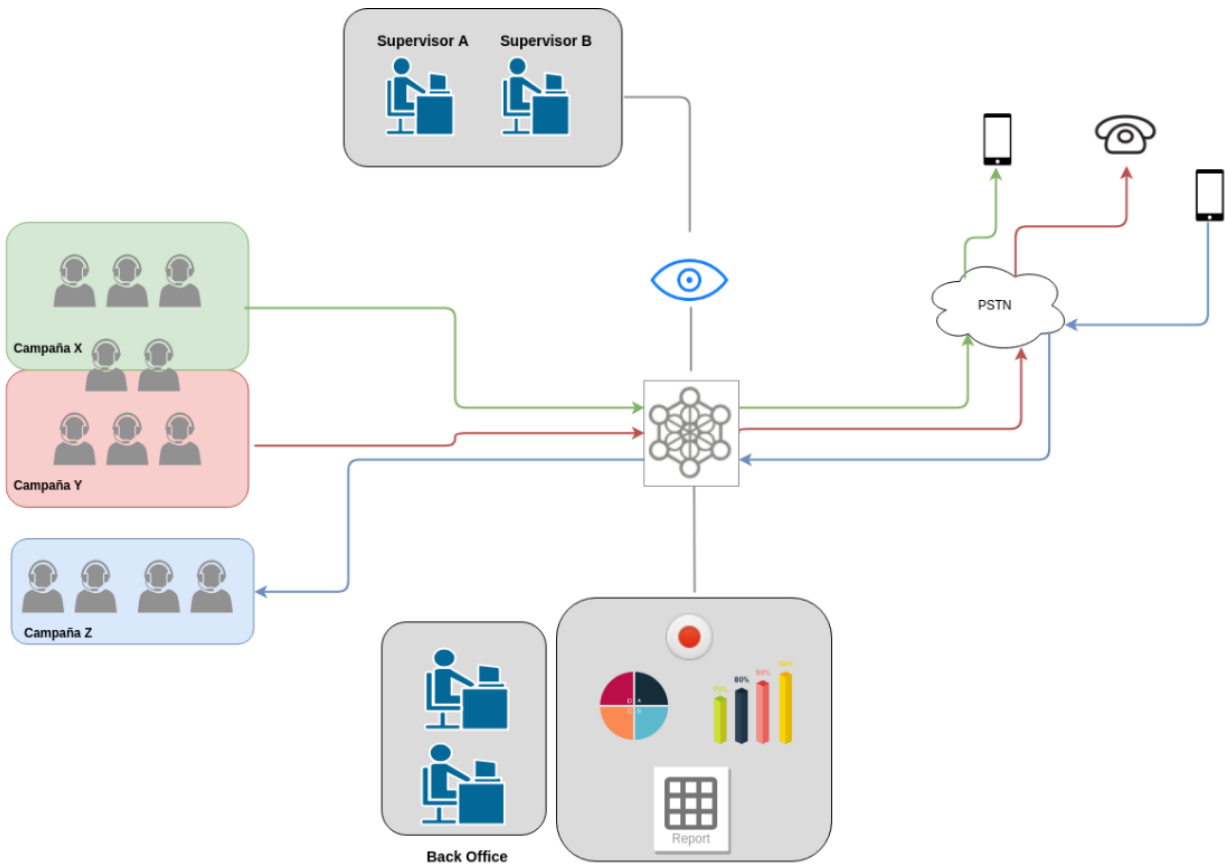


4.2 OML on a Customer Contact services company

In this scenario, OMniLeas can work as a Contact Center communications core with dozens or even hundreds of agents. This way can handle multiple SIP trunks at the same time, with its pertinent inbound and outbound communications routes

In these contexts scalability is basic requirement, because of operations are very dynamics y can demand peaks of users working in paralel. Scalability is guaranteed since our solution was conceived to be easily deployedon high availability cluster mode.

At the same time the RestFull API allows easily generate CRMs or web workflowsfor each campaign in order to fit client requirements.

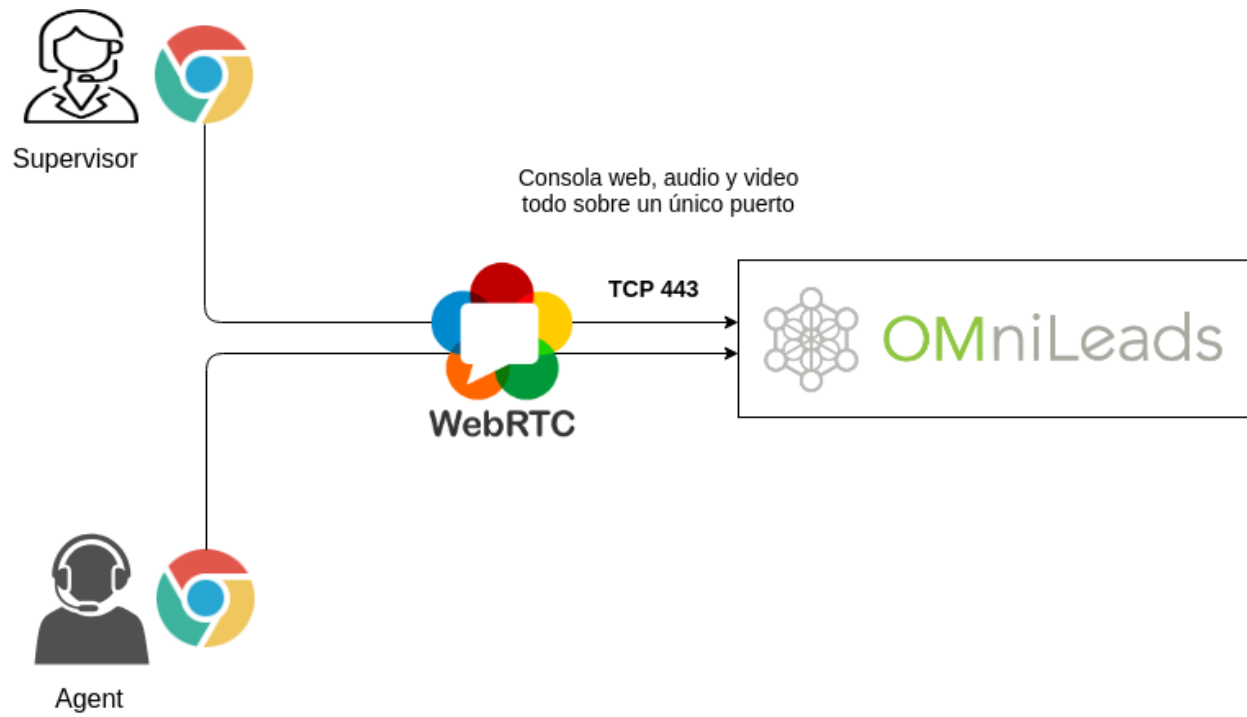


4.3 OML for carriers o Cloud PBX providers

If is needed to implement a CCaaS (Contact Center as a Service) OMniLeads is ideal having the advantage to use WebRTC and Docker as a technology bases.

We can cite as advantages:

- **WebRTC** removes the need to install desktop softphone applications, since voice and video flows through the browser for agents and supervisors. This removes a failure point and maintenance efforts on workstations.
- The **Codecs** implemented for audio and video are Opus and VP8, both designed for work on Internet and has the ability to dynamic adaptation to the available bandwidth, which avoids the uncomfortable call cuts off in traditional VoIP
- **Security**: information exchange between workstations and OML instance on Cloud, is encrypted with HTTPS, SRTP and DTLS standards.
- **Kamailio** is part of the core of OMniLeads communications stack components. Is a FLOSS advanced Proxy-SIP crucial for give security to VoIP servers.
- Docker allows easily OMniLeads deployment abstracting it of the underlying infrastructure, allowing run the system without any trouble on VPS, AWS, GCloud, etc



CHAPTER 5

¿Cómo me capacito ?

This documentation covers all aspects of the product, from technical issues inherent to the IT administrator, even functional aspects oriented to the agents, supervisors or leaders of the Contact Center.

ARCHITECTURE

In this section, the architecture of the project is introduced, together with the description of all its components:

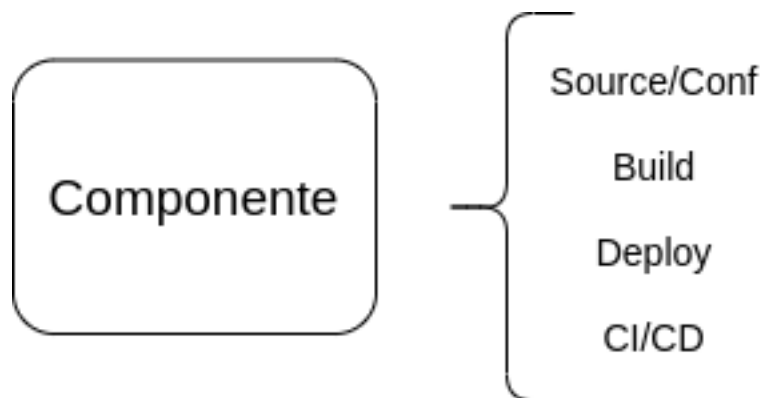
6.1 Architecture and Components

OMniLeads is a multi-component based application that resides in individual GitLab repositories, where the code is stored source and / or configuration, build, deploy and pipelines scripts CI / CD.

Although when executing an instance of OMniLeads the components interact as a unit through TCP / IP connections, the reality is that each one is its own entity with its GitLab repository and DevOps cycle.

At the build level, each component is distributed from RPMs (for installations on Linux) and Docker-IMG (for installations on Docker Engine). Regarding the deploy, each component has a script **first_boot_installer.tpl**, with which it can be invoked as *Provisioner* and thus easily deploy on a Linux host way automated, or run **manually** by editing variables about the script in question.

We can think of each component as a piece of a puzzle with its attributes:



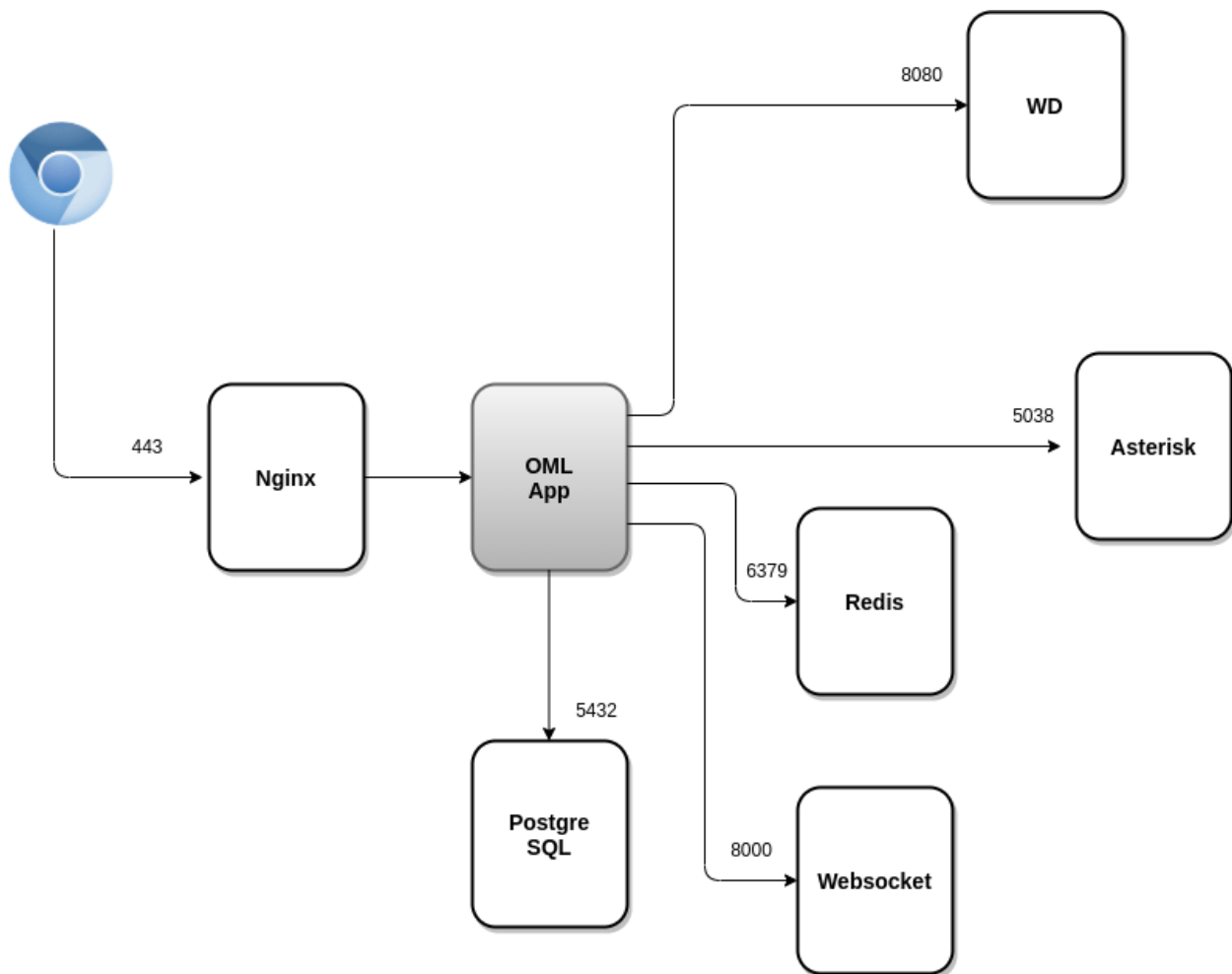
6.1.1 Description of each component

Each component is described below:

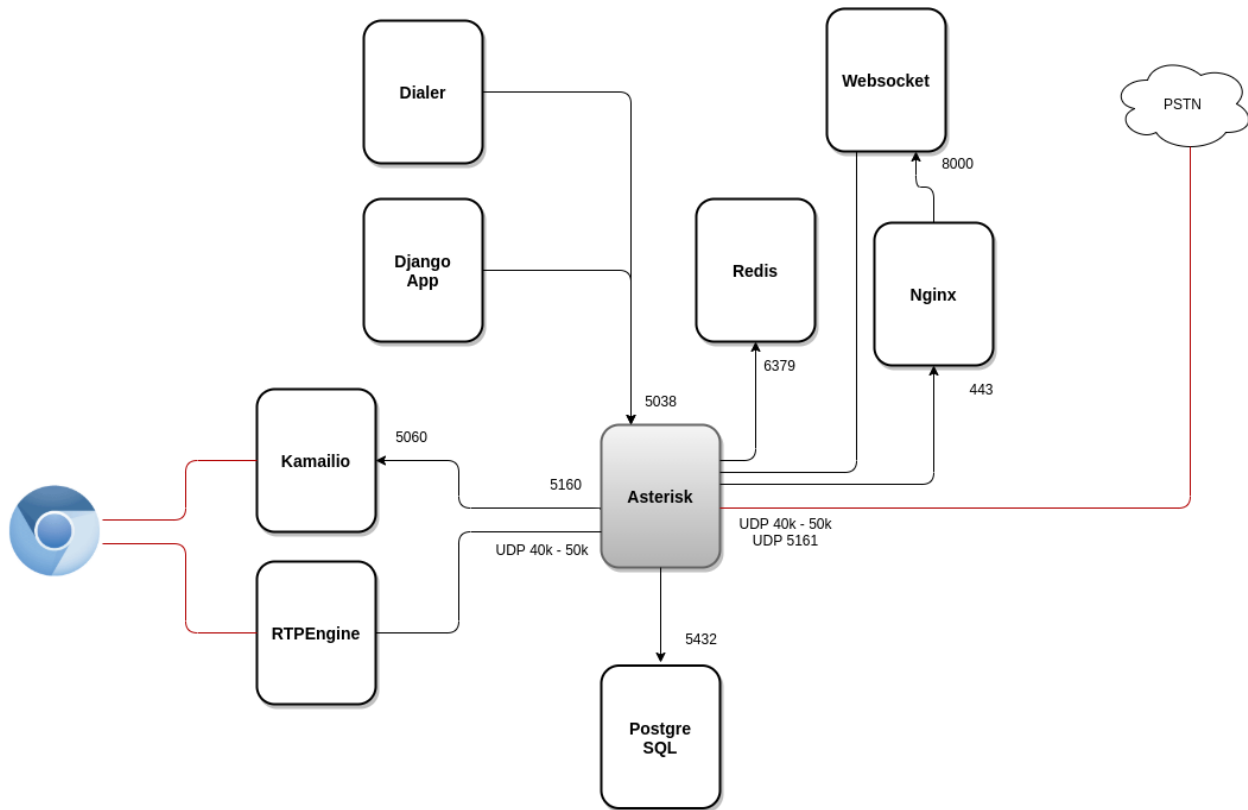
- **OMLApp** (<https://gitlab.com/omnileads/ominicontacto>): The web application (Python / Django) is contained in OMLApp.

Nginx is the webserver that receives HTTPS requests and redirects such requests to OMLApp (Django / UWSGI). OMLApp interacts with various components, either to store / provision configuration as well as in the generation of calls, or at the time of returning Agent / Campaign Monitoring and Report Views.

OMLApp uses PostgreSQL as SQL engine, Redis as cache and for provision Asterisk configuration, either via files .conf as well as generating certain key / value structures that are consulted by Asterisk in real time when processing calls about campaigns. OMLApp connects to the Asterisk AMI interface for generating calls and reloading some other configuration. It also makes connections to the WombatDialer API when necessary generate campaigns with predictive dialing.

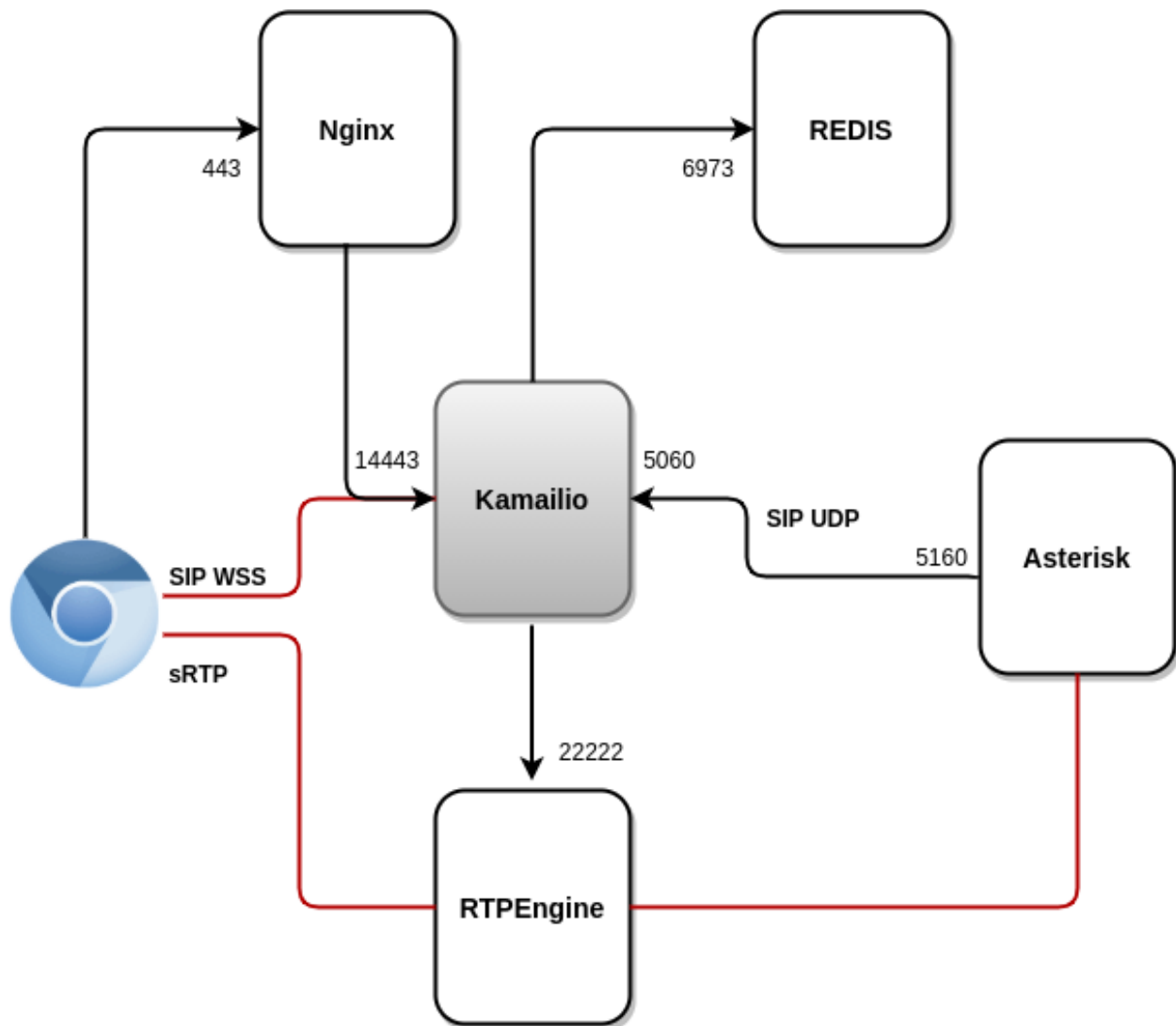


- **Asterisk** (<https://gitlab.com/omnileads/omlacd>): OMniLeads is based on the Asterisk framework as the basis of the ACD (Automatic Distributor of Calls). It is responsible for the implementation of business logic (campaigns telephone numbers, recordings, reports and channel metrics telephone). At the networking level, Asterisk receives AMI requests from OMLApp and from WombatDialer, while needs to run connections towards PostgreSQL to leave logs, towards Redis to query parameters of campaigns provisioned from OMLApp, and also to access Nginx for the establishment of the Websocket used to bring the content of configuration files contained in Asterisk (etc / asterisk) and generated from OMLApp.

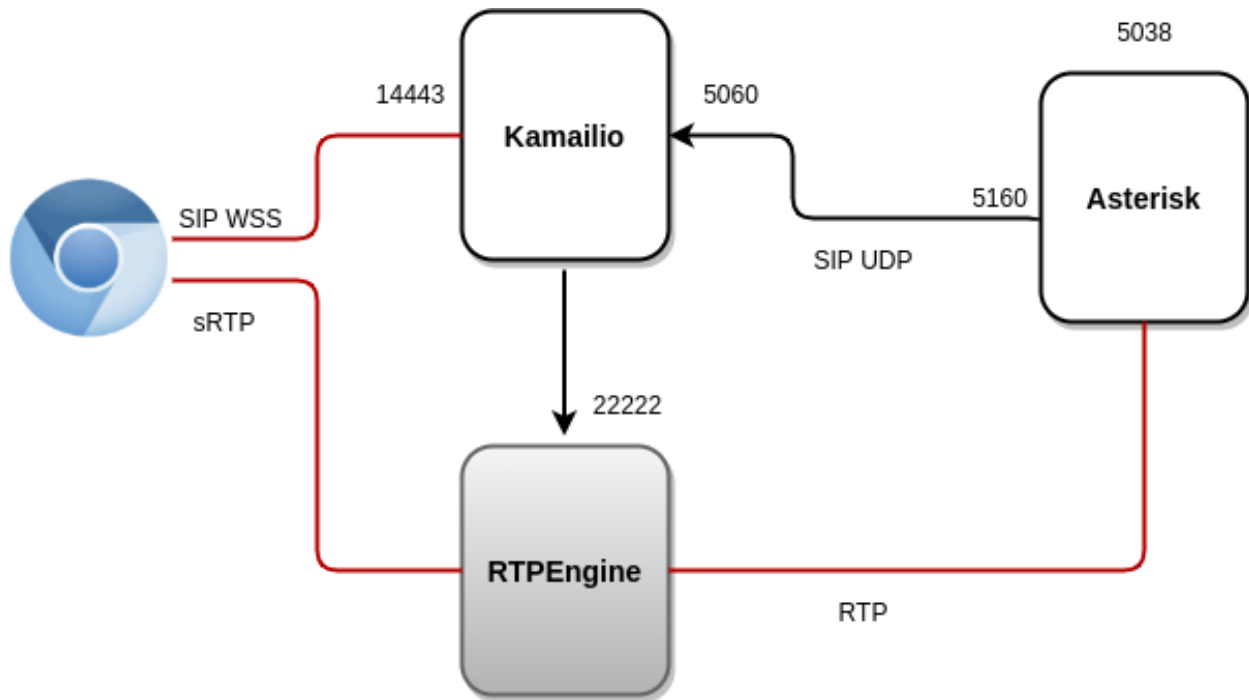


- **Kamailio** (<https://gitlab.com/omnileads/omlkamailio>): This component is used in conjunction with RTPEngine (WebRTC bridge) at the time of managing WebRTC (SIP over WSS) communications against agents, while holding sessions (SIP over UDP) against Asterisk. Kamailio receives the REGISTERs generated by the web-phone (JSSIP) from the Agents, therefore takes care of the work of * registration and locating * users using Redis to store the address of network of each user.

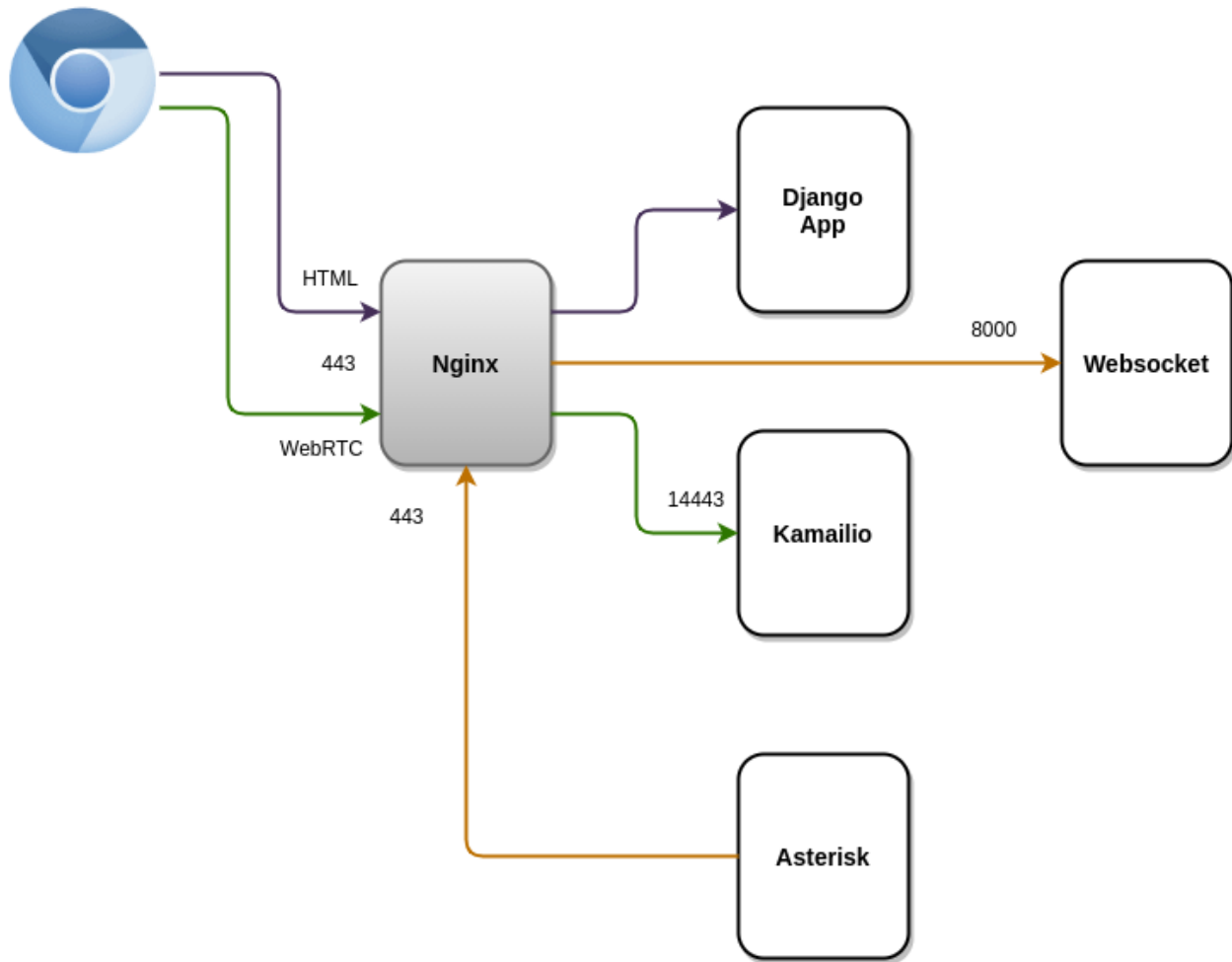
For Asterisk, all agents are available at the URI of Kamailio, so Kamailio receives INVITEs (UDP 5060) from Asterisk when it requires *locate* some agent to connect a call. Finally, it is worth mentioning the fact that Kamailio generates connections towards RTPEngine (TCP 22222) requesting an *SDP* at the time of establishing SIP sessions between Asterisk VoIP and WebRTC agents.



- **RTPEngine** (<https://gitlab.com/omnileads/oml RTPengine>): OMniLeads supports RTPEngine at the time of *transcoding and bridge* between the WebRTC technology and VoIP technology from the *audio* point of view of the. Component maintains audio channels *sRTP-WebRTC* with Agent users on the one hand, while establishing channels on the other *RTP-VoIP* against Asterisk. RTPEngine receives connections from Kamailio to port 22222.

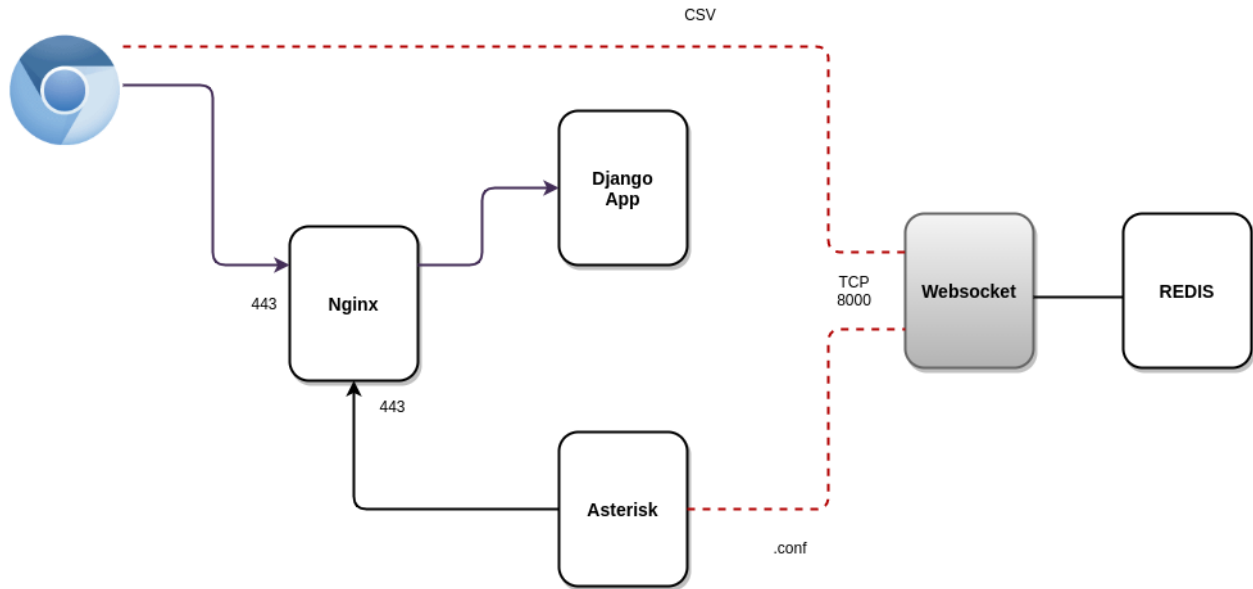


- **Nginx** (<https://gitlab.com/omnileads/omlNginx>): The web server of the project is Nginx, and its task is to receive the requests *TCP 443* by part of the users, as well as from some components like Asterisk. From one hand, Nginx is invoked every time a user accesses the URL of the deployed environment. If user requests have as destination render some view of the web application *Django*, then *Nginx* redirects the request to *UWSGI*, while if the user requests are destined to the REGISTER of their webphone JSSIP, then *Nginx* redirects the request to Kamailio (for setting a websocket SIP). Also, *Nginx* is invoked by Asterisk when setting websocket against *Websocket-Python* from OMniLeads, provisioning the provided configuration from *OMLApp*.

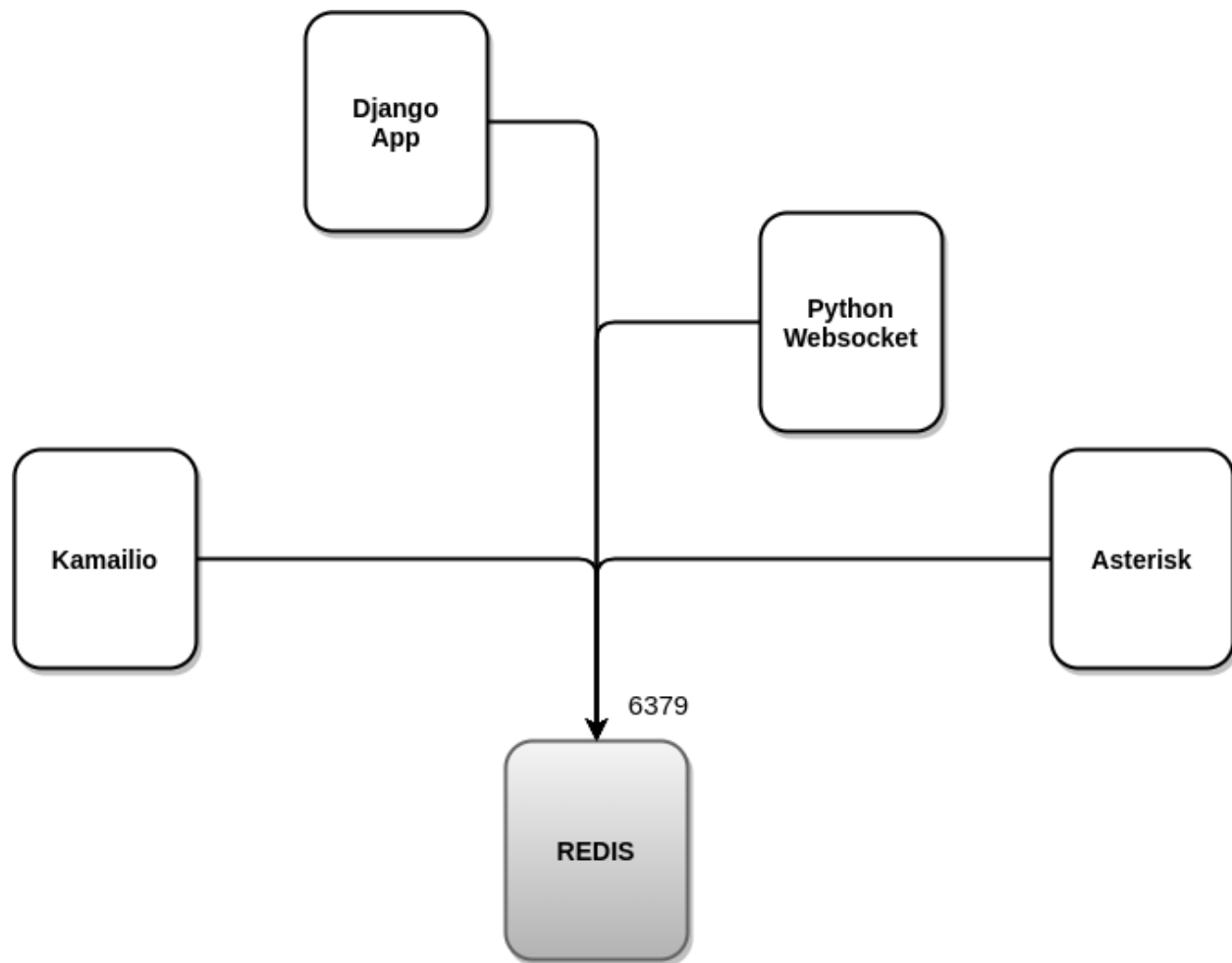


- **Python websocket** (<https://gitlab.com/omnileads/omnileads-websockets>): OMniLeads relies on a websockets server (based on Python), used to leave background tasks running (reports and generation of CSVs) and receive an asynchronous notification when the task has been completed, which optimizes the performance of the application. It is also used as a *bridge* between *OMLApp* and *Asterisk* in provisioning the configuration of *.conf* files (etc / asterisk).

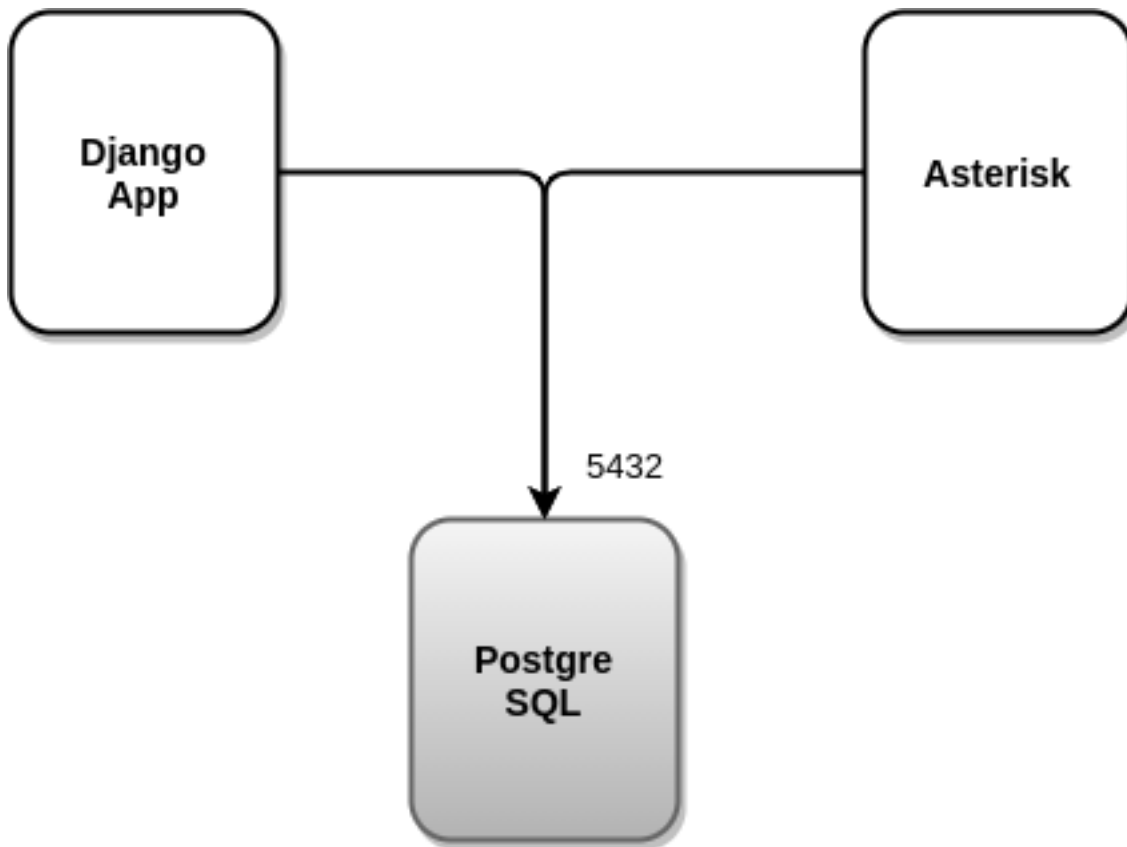
When starting Asterisk, a process connects the *websocket* against component, and from there, it receives notifications every time configuration changes are provided. In your default configuration, it raises the port *TCP 8000*, and the connections received are always redirected from *Nginx*.



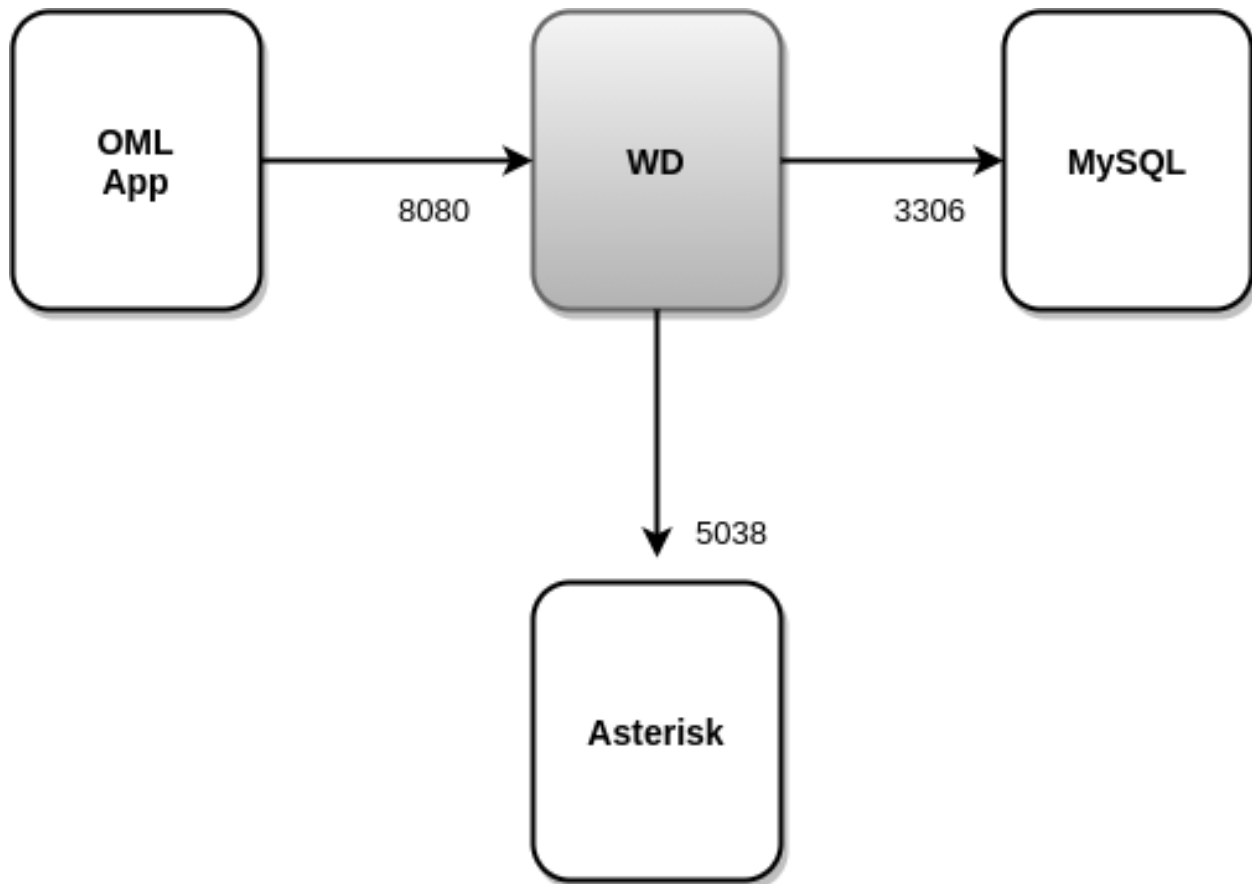
- **Redis** (<https://gitlab.com/omnileads/omlredis>): *Redis* is used with 3 very specific purposes. On the one hand, as a cache to store recurring Query Results Involved in Supervisory Views of campaigns and agents; on the other hand it is used as *DB* for the presence and location of users; and finally for the * Asterisk * configuration storage (etc / asterisk /) as well as also of the configuration parameters involved in each module (campaigns, trunks, routes, IVR, etc.), replacing the native alternative of * Asterisk * (* AstDB *).



- **PostgreSQL** (<https://gitlab.com/omnileads/omlpgsql>): *PGSQL* is the DB SQL engine used by OMniLeads. From there, it materializes all the reports and metrics of the system. Also, it stores all the configuration information that should persist in time. Receives connections on its *TCP port 5432* from components *OMLApp* (read / write) and from *Asterisk* (write logs).



- **WombatDialer** ([https://manuals.loway.ch/WD_UserManual-chunked / ch07.html](https://manuals.loway.ch/WD_UserManual-chunked/ch07.html)): To work with predictive dialing campaigns, OMniLeads uses this third party software called WombatDialer. This dialer, has an API *TCP 8080* on which *OMLApp* generates connections to provision campaigns and contacts, while WombatDialer falls back to the *AMI* interface of the *Asterisk* component at the Time to generate automatic outgoing calls, and check the status of The agents of each campaign. This component uses its own engine **MySQL** to operate.

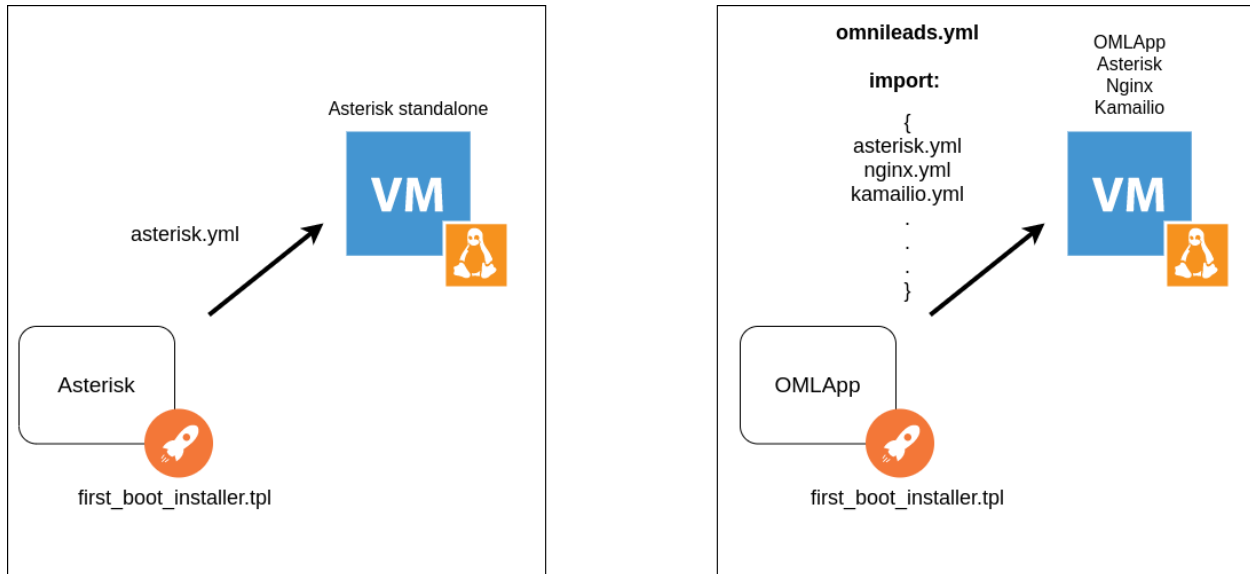


6.1.2 Deploy and environment variables

Having processed the previous exposition on the function of each component and its interactions in terms of networking, we move on to address the * deploy * process.

Each component has a *bash script* and *Ansible playbook* that allows the materialization of the component, either on a Linux-Host dedicated or living with others on the same host.

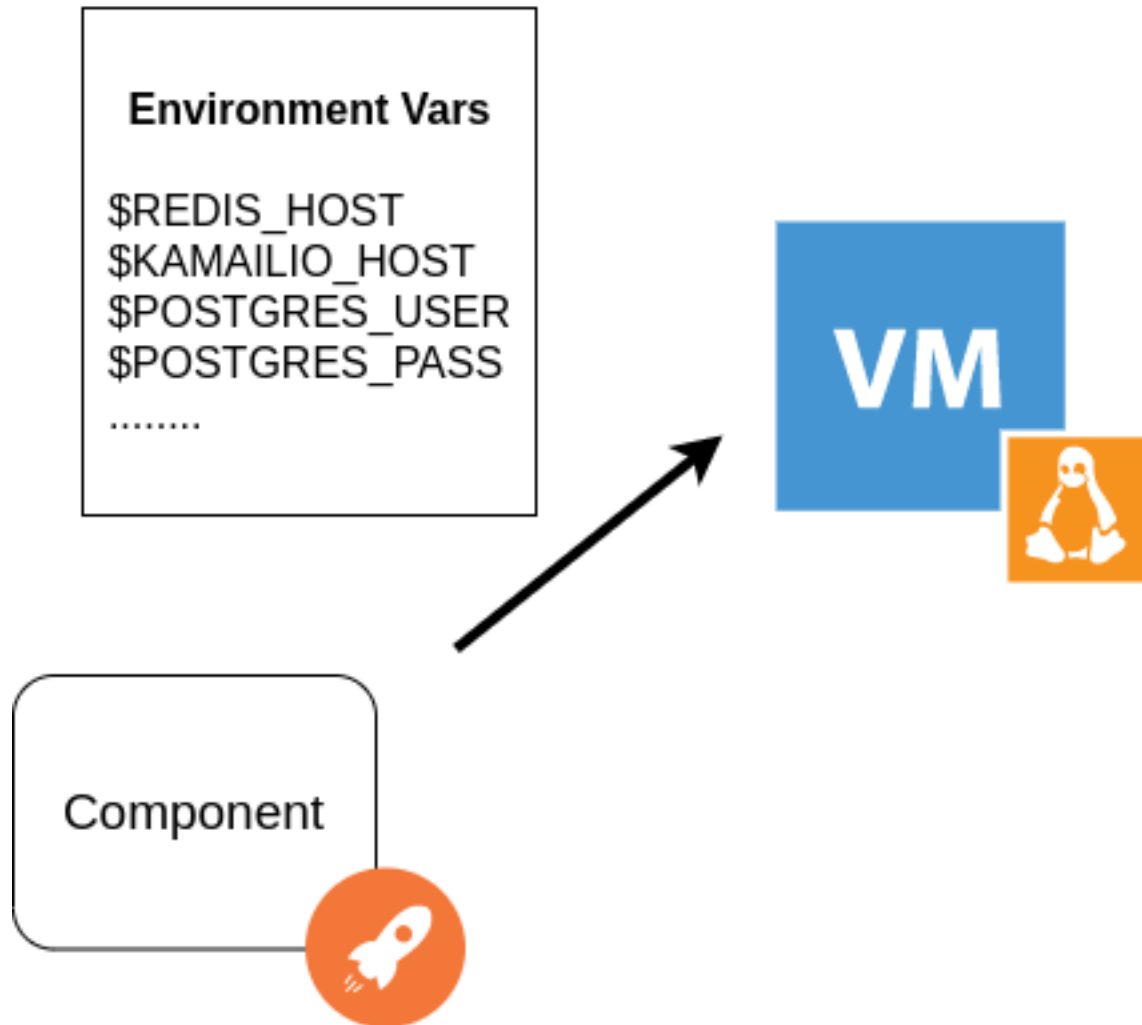
This is thanks to the fact that the *Ansible playbook* can be invoked from the *bash script* called *first_boot_installer.tpl*, in the case of going to the same as *provisioner* of a Linux-Host dedicated to host the component within the framework of a *cluster*, as well as imported by the *Ansible playbook* of the *OMLApp* component at the time of deploying several components on the same host where the application runs *OMLApp*.



Therefore, we conclude that each component can either exist in a standalone host, or also coexist with OMLApp in the same host. These possibilities are covered by the installation method.

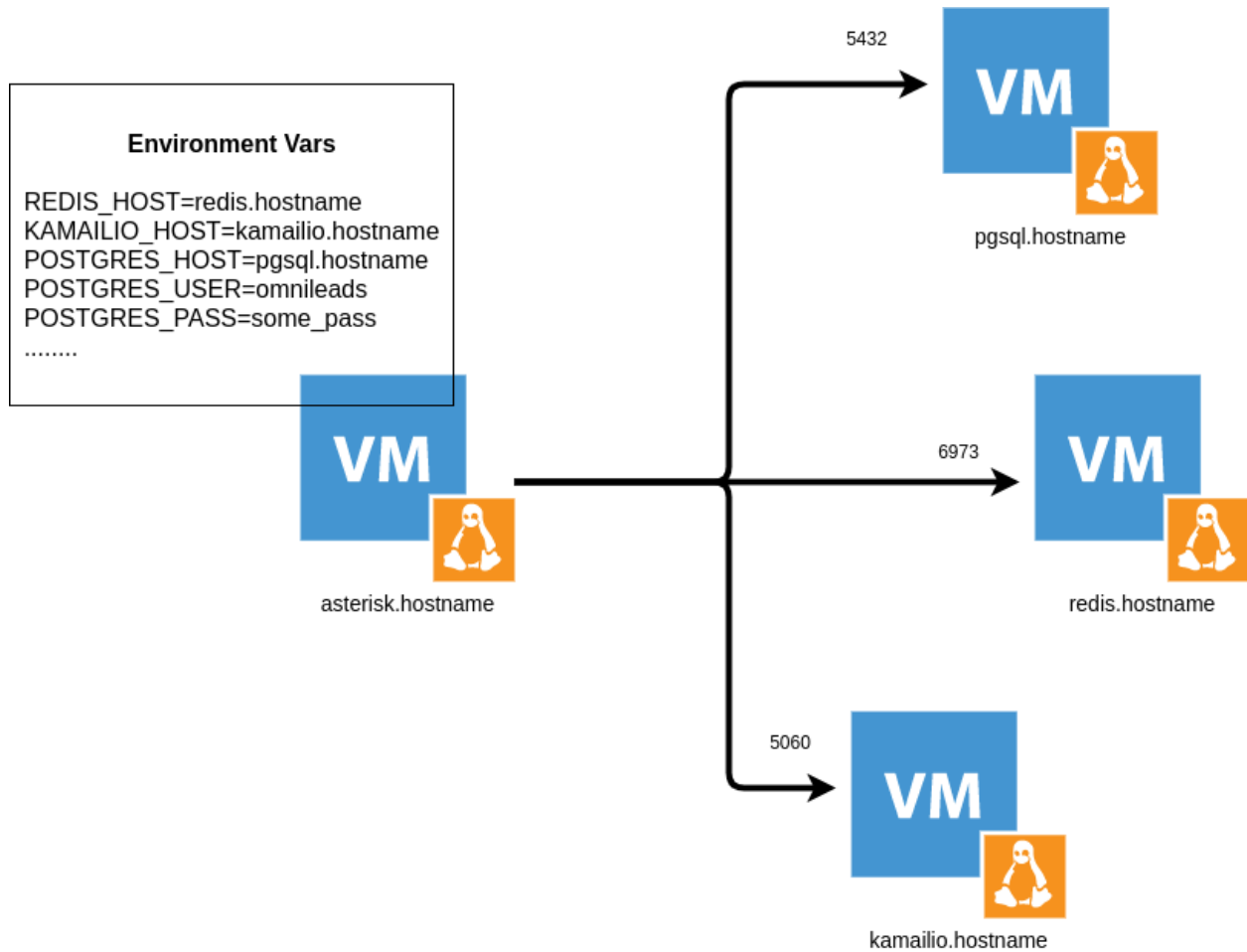
Above installation method is completely based on **variables from environment** that are generated in the deploy and are intended among other stuff, containing the network and port addresses of each component required to achieve interaction. That is, all files from configuration of each OMniLeads component, *search* for its peer by invoking OS environment variables. For example, the Asterisk component points your AGIs to the envvar **\$REDIST_HOST** and **\$REDIS_PORT** at the time of trying to generate a connection to *Redis*.

Thanks to the environment variables, a compatibility between the bare-metal approaches and docker containers is reached, that is, we can deploy OMniLeads by installing all components on one host, distributing them in several hosts or directly on Docker containers.



The fact of provisioning the configuration parameters via variables of environment, and also considering the possibility of always deploying the application protecting the data that must persist (recordings of calls and DB PostgreSQL) on resources mounted on the Linux system files that host each component, we can then raise the fact of working with immutable infrastructure as an option if we would like to. We can easily destroy and recreate each component, without losing important data when resizing the Components or raising updates. We can just discard the host where a version runs and deploy a new one with the latest upgrade.

We have the potential of the approach posed by the paradigm of infrastructure as code or immutable infrastructure, raised from the perspective of the new IT generations that operate within the DevOps culture. This approach is somewhat optional, as it can manage updates from the most traditional perspective without having to destroy the instance that hosts the component.



6.1.3 The potential of turning to cloud-init as a provisioner

Cloud-init is a software package that automates the initialization of Instances from the cloud during system startup. You can configure *cloud-init* to perform a variety of tasks. Some Examples of tasks that cloud-init can perform are:

- Configure a hostname.
- Installing packages on an instance.
- Running provisioning scripts.
- Suppress the default behavior of the virtual machine.

As of OMniLeads 1.16, each component contains a script called *fisrt_boot_installer.tpl*. Such a script can be precisely invoked at the cloud-init level, so that on the first boot of the operating system A fresh install of the component can be released.

As we have mentioned, it is possible to invoke the script at cloud VM creation time:

- [DigitalOcean Example](#)
- [VULTR Example](#)
- [AWS Example](#)

Another option is to render as [Terraform template](#) to be launched as provisioning of every instance created from Terraform.

Beyond the component in question, the *first_boot_installer.tpl* has as a purpose:

- Install some packages.
- Adjust some other configurations of the virtual machine.
- Determine network parameters of the new Linux instance.
- Run the *Ansible playbook* that installs the component on the operating system.

The first 3 steps are skipped when the component is installed from OMLApp, thus sharing the host.

In this chapter is covered all the types of installation of the App

7.1 OMniLeads deploy

Before proceeding with the installation, it is recommended to read and understand (*Architecture and Components*).

As described in the architecture section, each component has a potentially usable install script like *provisioner* at the creation time of the Linux virtual machine, that will host the component in matter. Also, it is feasible to run it in *self-hosted* mode when deploying the component on a Linux host, i.e. pouring the content of **first_boot_installer.tpl** over bash-script located in the virtual machine to be deployed, adjusting the variables and finally launching its execution.

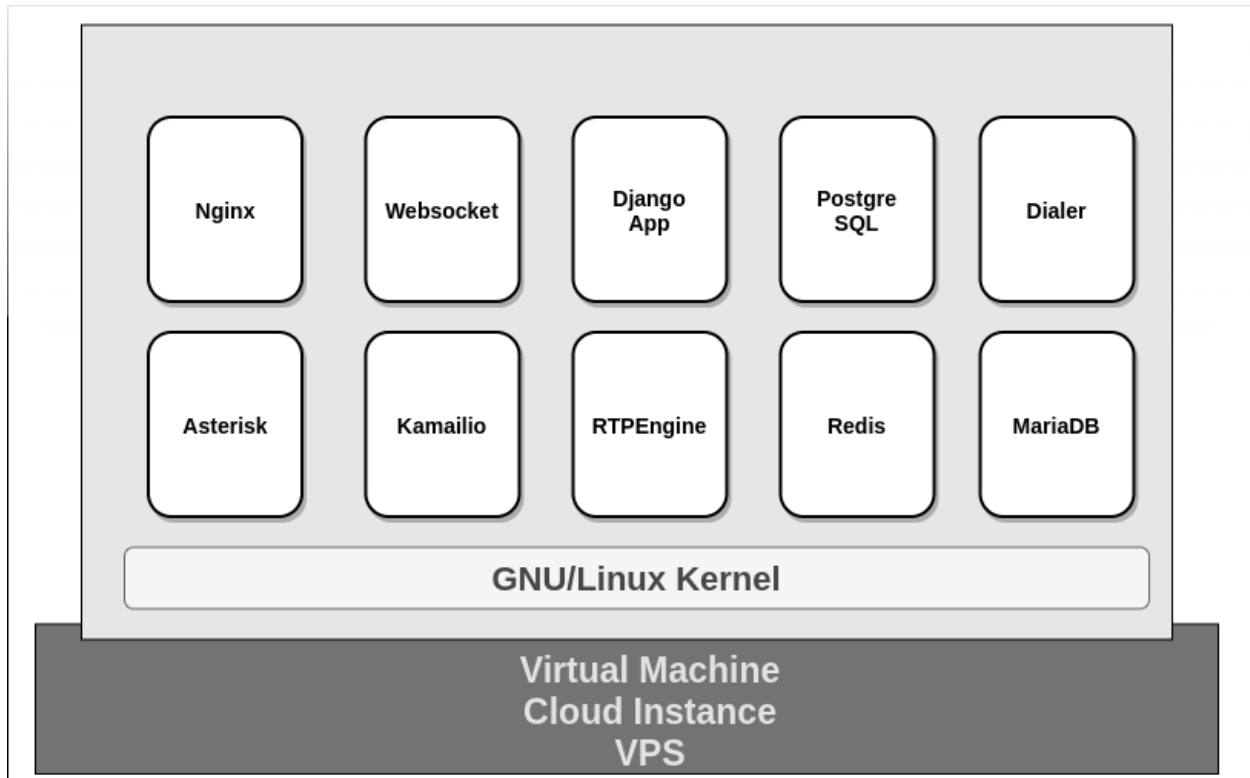
Therefore, we have different ways to reach the goal of installing OMniLeads, some fully automated, and others within a traditional scheme based on running an installation script manually.

Note: Linux distributions for which compatibility is guaranteed are [CentOS7-Minimal](#) and [Amazon Linux 2](#).

Next, we consider different scenarios corresponding to available formats.

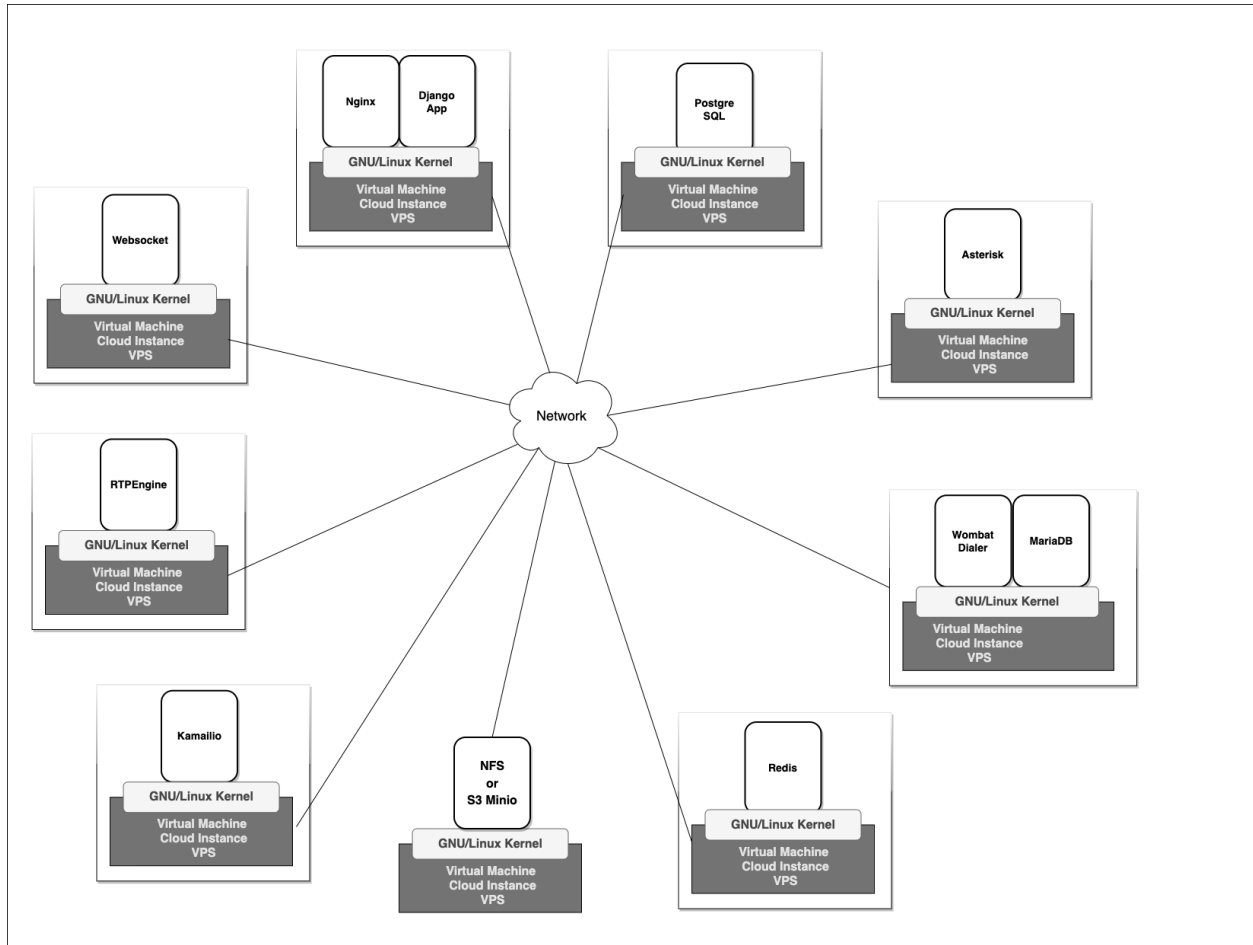
7.1.1 Deploy de OMniLeads Onpremise

OMniLeads can run like a traditional application, deploying one installation of all components on a single host. To this type of Installation, we call it **OMniLeads AIO (All In One)**.



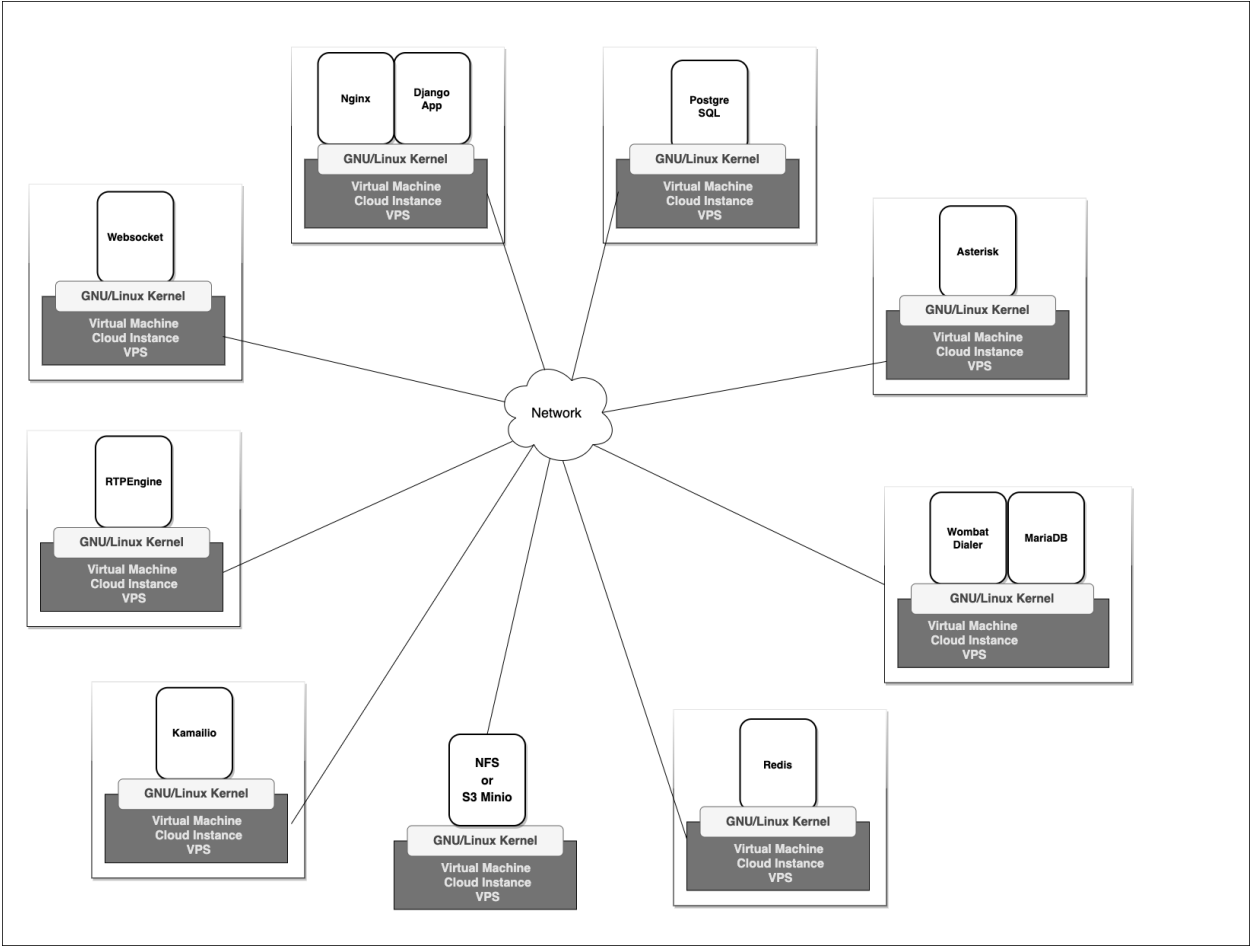
También, podemos separar los componentes de manera tal que puedan ser ejecutados como servicios aislados corriendo sobre hosts dedicados. Ésto significa que se puede aislar componentes sobre hosts dedicados con el fin de plantear una operación más robusta y resiliente, ya que entre otras ventajas, permite aislar/resguardar datos persistentes como grabaciones de llamadas, base de datos, y demas utilizando S3 object storage o bien NFS.

<<<<<<< HEAD Aprovechando el hecho de que cualquier infraestructura onpremise cuenta con la posibilidad de virtualizar instancias de linux, asi como tambien la sencillez que implica obtener una instancia cloud linux (vía web o API) y aprovisionar la misma en el instante posterior al boot, se debe considerar el despliegue por defecto en **cluster de componentes OMniLeads**.



En esta sección se cubre el despliegue onpremise: [about_install_linux](#).

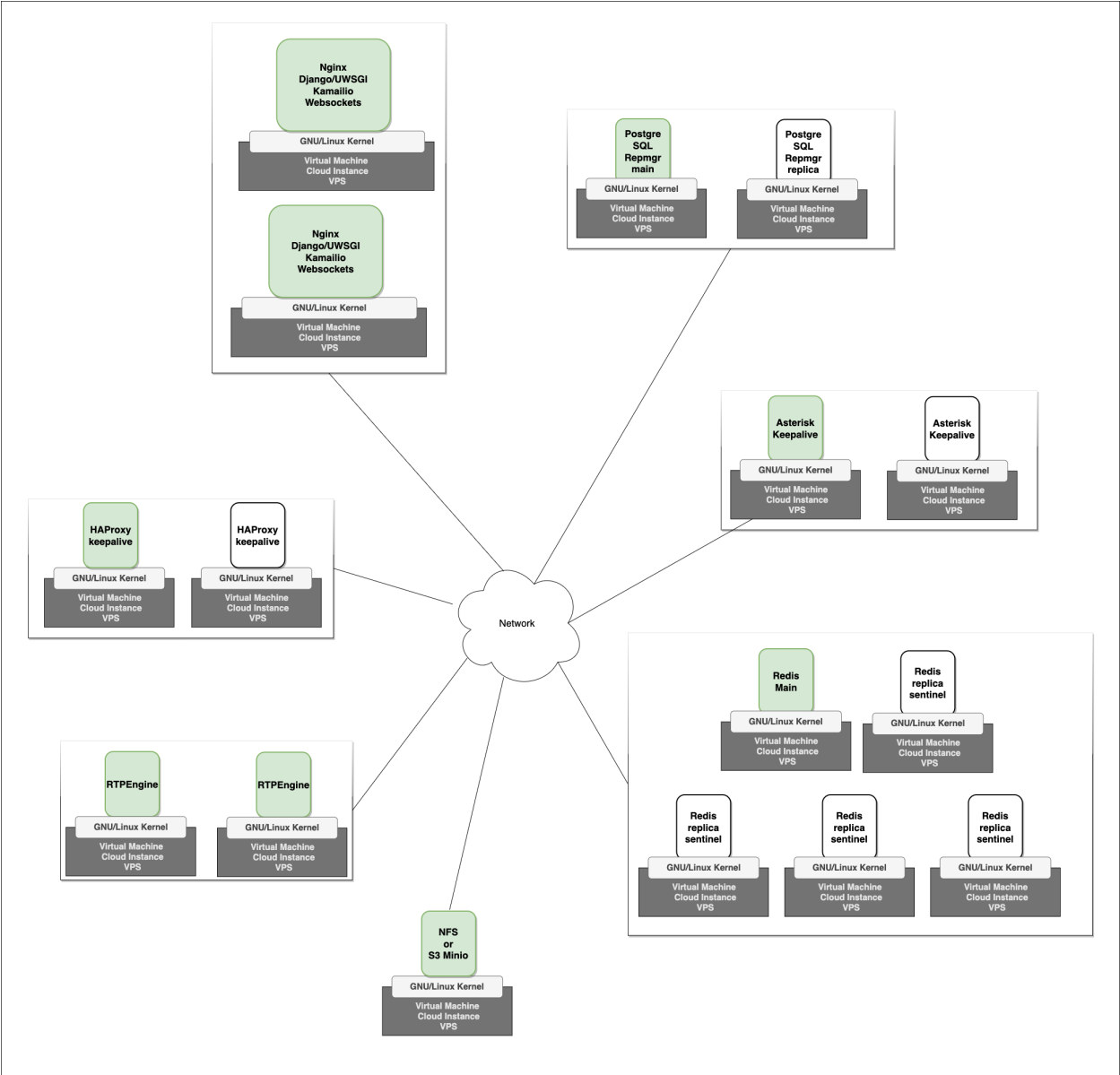
Aprovechando el hecho de que cualquier infraestructura onpremise cuenta con la posibilidad de virtualizar instancias de linux, así como también la sencillez que implica obtener una instancia cloud linux (vía web o API) y aprovisionar la misma en el instante posterior al boot, se debe considerar el despliegue por defecto en **cluster de componentes OMniLeads**.



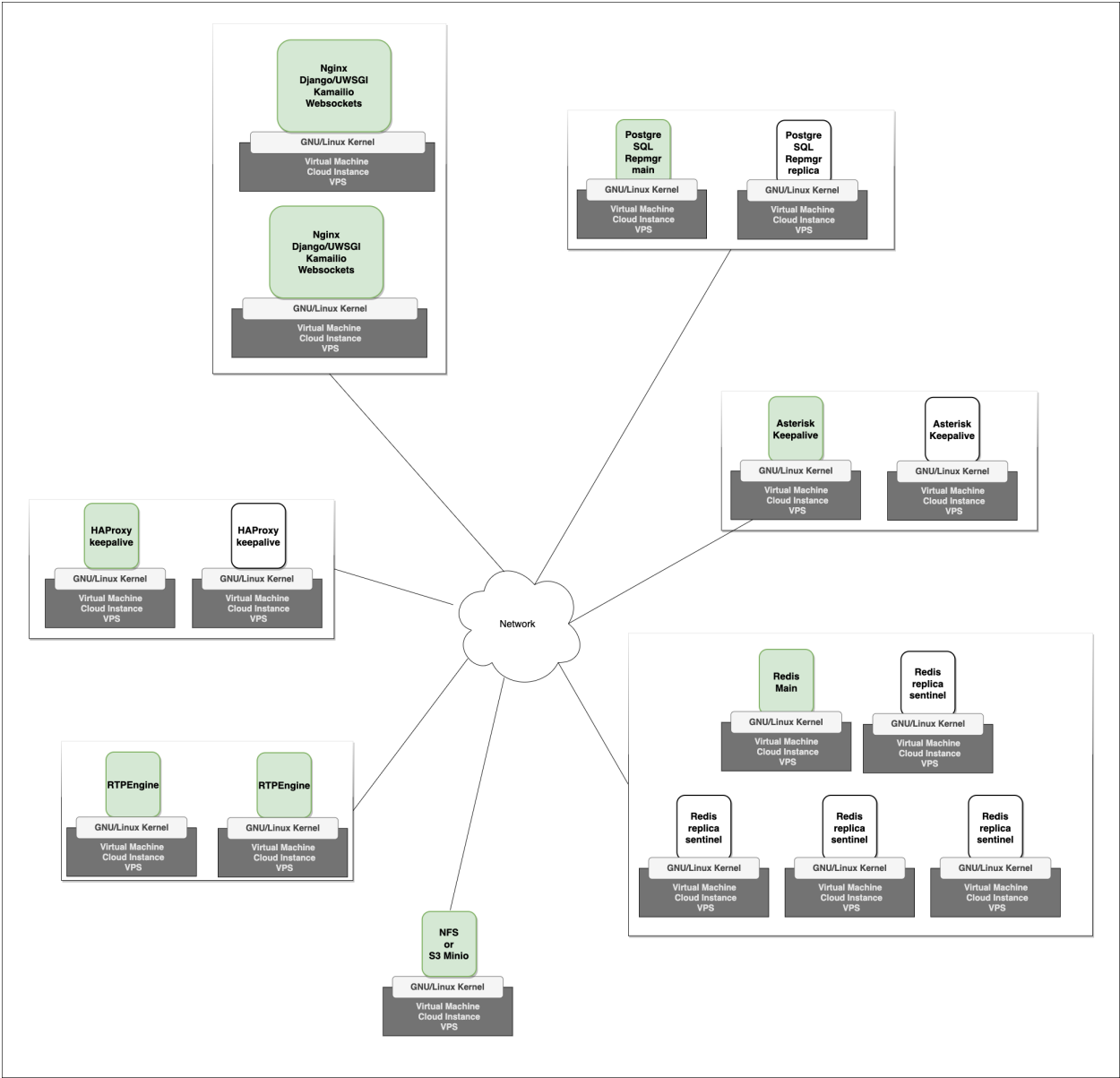
En esta sección se cubre el despliegue sobre instancias linux onpremise: `about_install_onpremise`. >>>>>> develop

7.1.2 Deploy de OMniLeads Onpremise Alta Disponibilidad

OMniLeads puede ser desplegado bajo un esquema de alta disponibilidad con redundancia en todos sus componentes.
<<<<<<< HEAD



En esta sección se cubre el despliegue en cluster de Alta Disponibilidad: [about_install_linux_ha](#).



En esta sección se cubre el despliegue en cluster de Alta Disponibilidad: [about_install_linux_ha](#).

7.1.3 Deploy de OMniLeads sobre Digital Ocean

En esta seccion se cubre la instalacion de la aplicacion aprovechando las ventajas de la nube de Digital Ocean.

- [about_install_digitalocean](#)

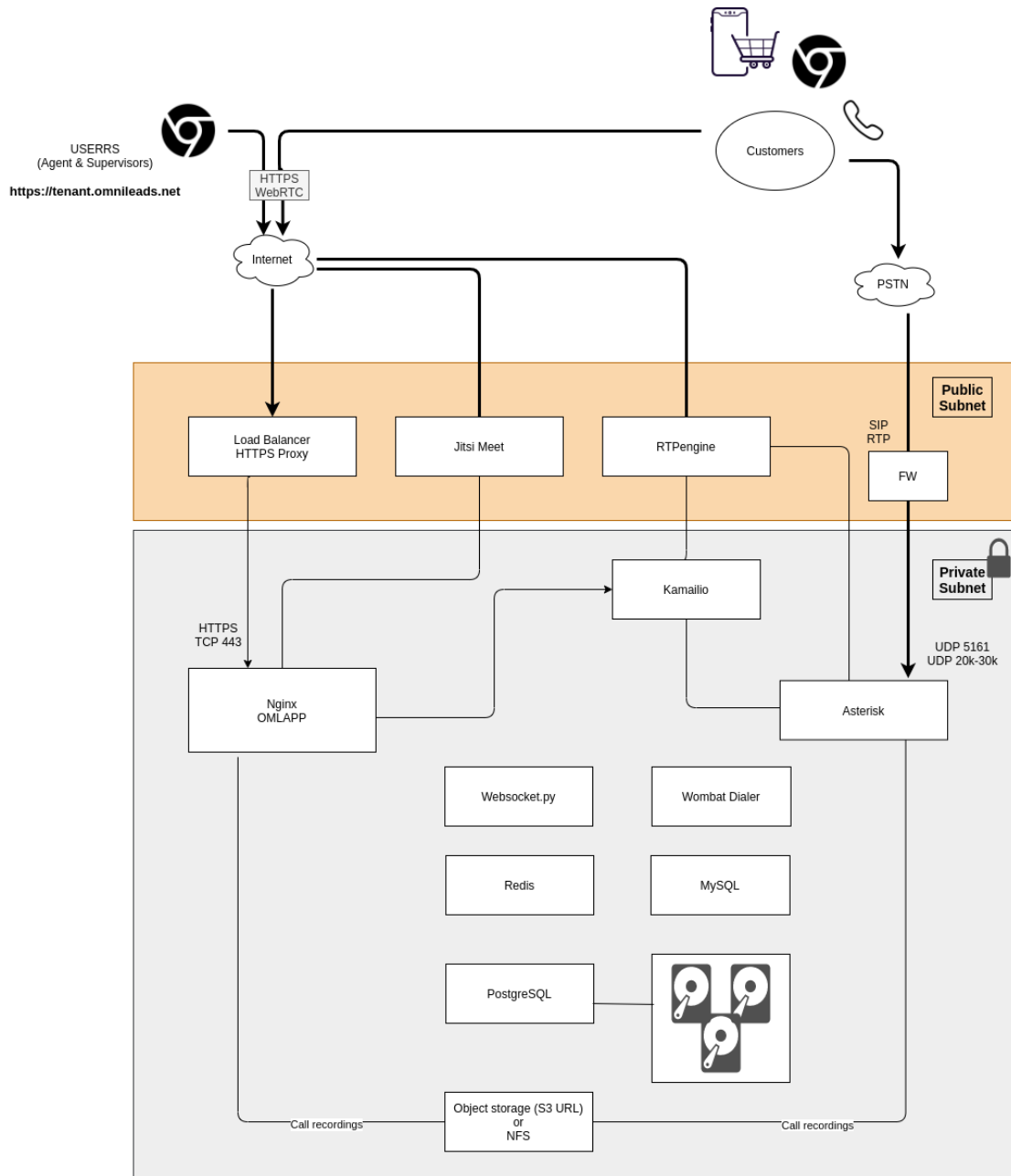
>>>>>>> develop

7.1.4 OMniLeads Deployment based on Terraform

As of OMniLeads version 1.16, it is possible to use Terraform to deploy OMniLeads on some specific cloud infrastructures. This list of providers will increase in the future thanks to the community contribution.

For those of you unfamiliar with Terraform and interested in this topic, we leave a link to [documentation](#). Terraform allows us to *code* a deployment. In terms of Terraform, we can define a deployment of OMniLeads, such as the following set of actions:

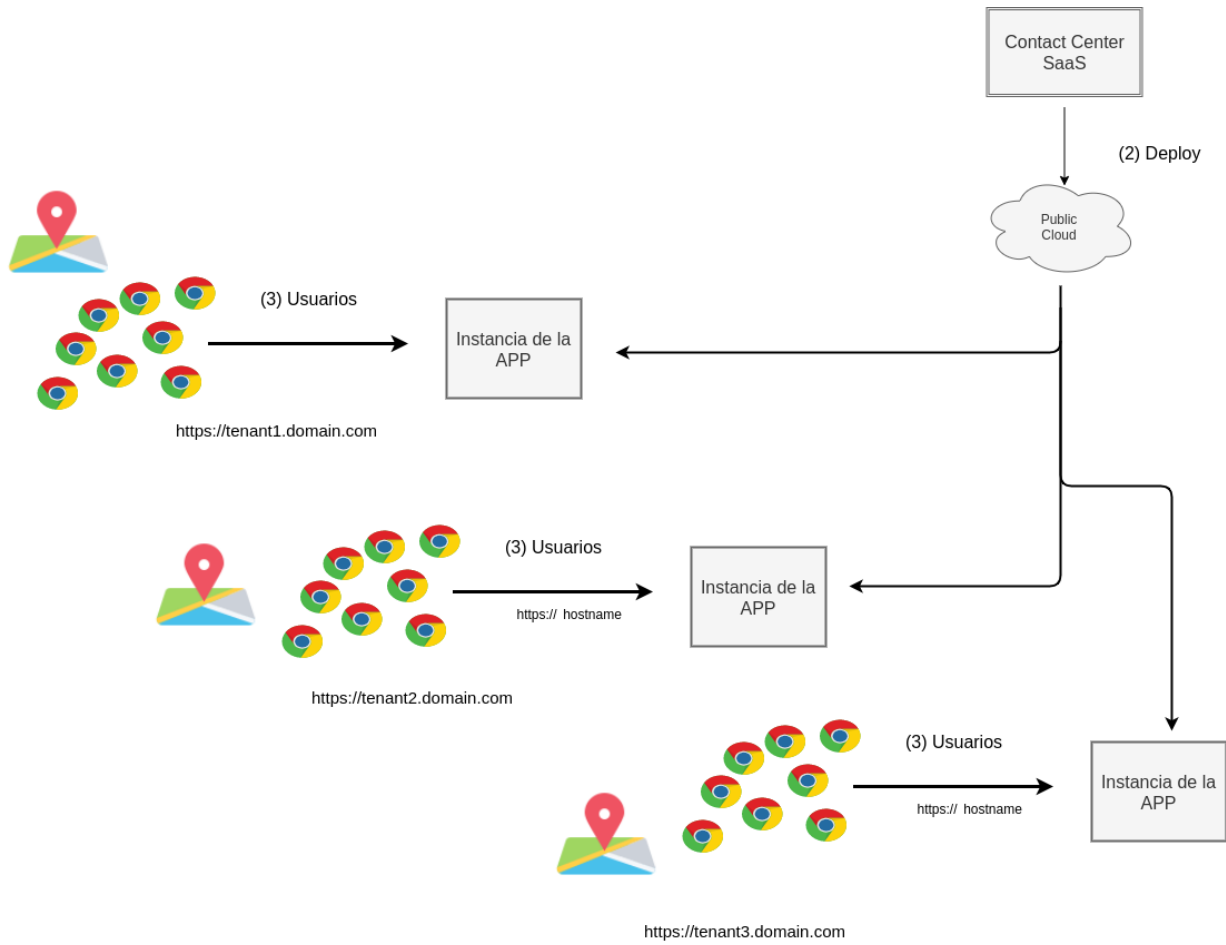
- Infrastructure networking configuration.
- Creation of virtual machines, DB clusters, load balancers.
- OMnileads Deployment and its components on virtual machines
- Application of security settings.
- Generation of SSL certificates and URL access for users.



So you can just set a variable file and then launch a command (terraform apply), and in a matter of minutes you can have an OMniLeads “as a Service” deployment ready to use. Users can start operating from anywhere.

Note: The deployment of the Jitsi Meet, WombatDialer and MySQL components is configurable according to the needs of the campaign configuration regarding predictive and/or video call features.

The approach of deploying infrastructure cloud, allows us to fully automate the generation of a subscriber to the Contact Center Service in the cloud.



Below the list of current supported public clouds:

- `about_install_terraform_digitalocean`
- `about_install_terraform_aws`

In this section the essential configuration are shown once an instance of OMniLeads is installed

8.1 Security Considerations

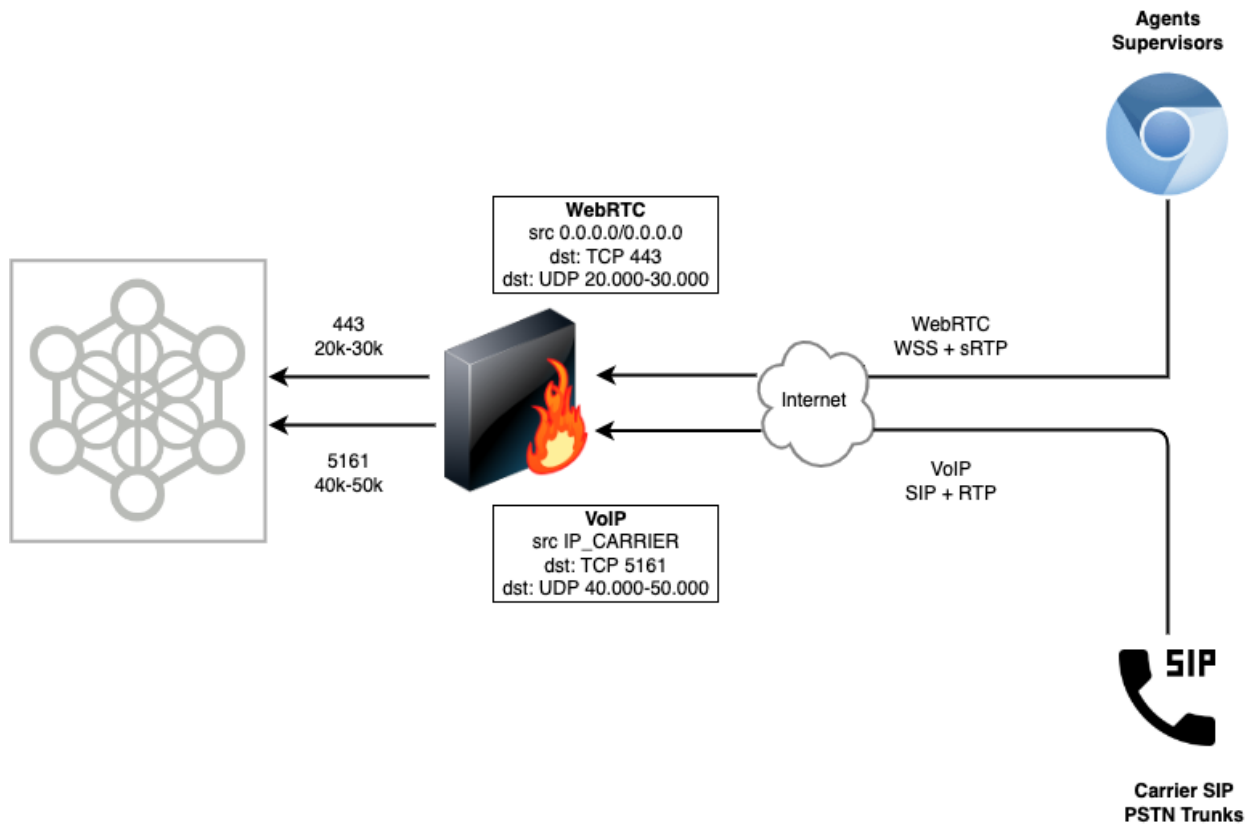
OMniLeads is a web application, designed to operate under the protection of at least one perimeter firewall or cloud firewall in an environment of cloud computing.

Ideally it is recommended to deploy OMniLeads together with an *HTTP Proxy or Edge Cloud Load Balancer* for HTTPS requests and a Session Border Controller for VoIP edge management. This makes the security deployment more robust.

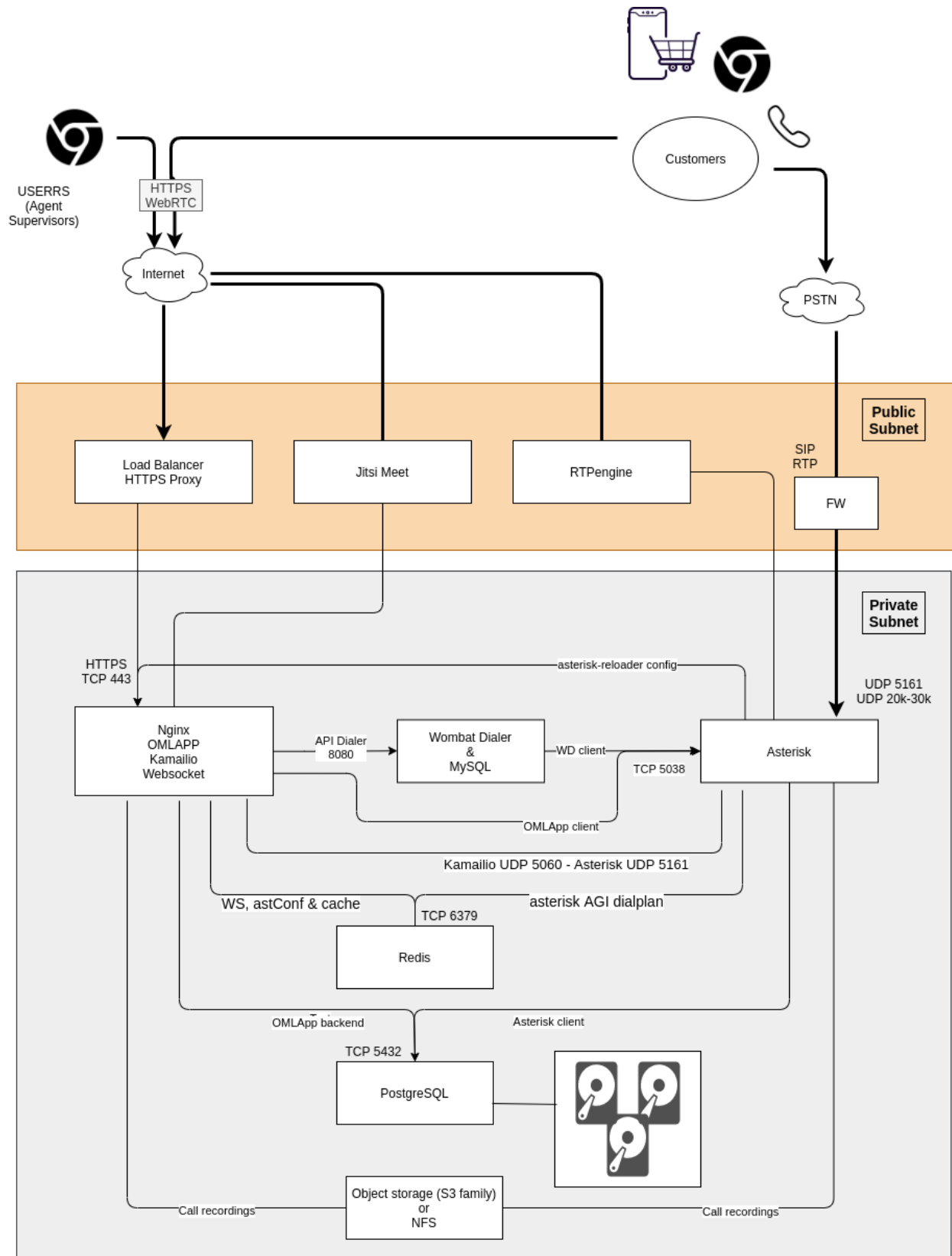
Considering the scenario where users are going to access the application both from the local network and also from the Internet, the following list of ports should be exposed:

- **UDP 5161:** Tráfico SIP proveniente de la PSTN, se debe validar por IP de origen. Esta puerto es el indicado si la instancia que aloja OML posee una dirección IPV4 pública a nivel tarjeta de red.
- **UDP 5162:** SIP traffic coming from the PSTN, it should be validated by source IP. This port is the main one if the instance that hosts OML owns a public IPV4 address behind NAT.
- **UDP 40000 a 50000:** Tráfico RTP proveniente de la PSTN. se debe validar por IP de origen. Esta puerto es el indicado si la instancia que aloja OML posee una dirección IPV4 pública detrás de NAT.
- **UDP 20000 a 30000:** Tráfico WebRTC proveniente de los usuarios. En este caso los usuarios se suponen en modalidad home-office, por lo que se deja abierto a Internet.
- **HTTPS 443:** Web traffic and WebRTC coming from users. In this case the users are supposed to be in home-office scenario, so they are set to open.

All In One Deployment:



Cluster deployment on a cloud-computing scheme:



Important: In case you need to expose the VoIP ports to ALL internet, we strongly recommend managing VoIP

security using a Session Border Controller or an *Asterisk* or *Freeswitch* configured as a SIP component edge so that OMniLeads is not exposed to all IP addresses. As minimum, it will start to receive *garbage* from SIP of multiple origins.

INITIAL CONFIGURATIONS

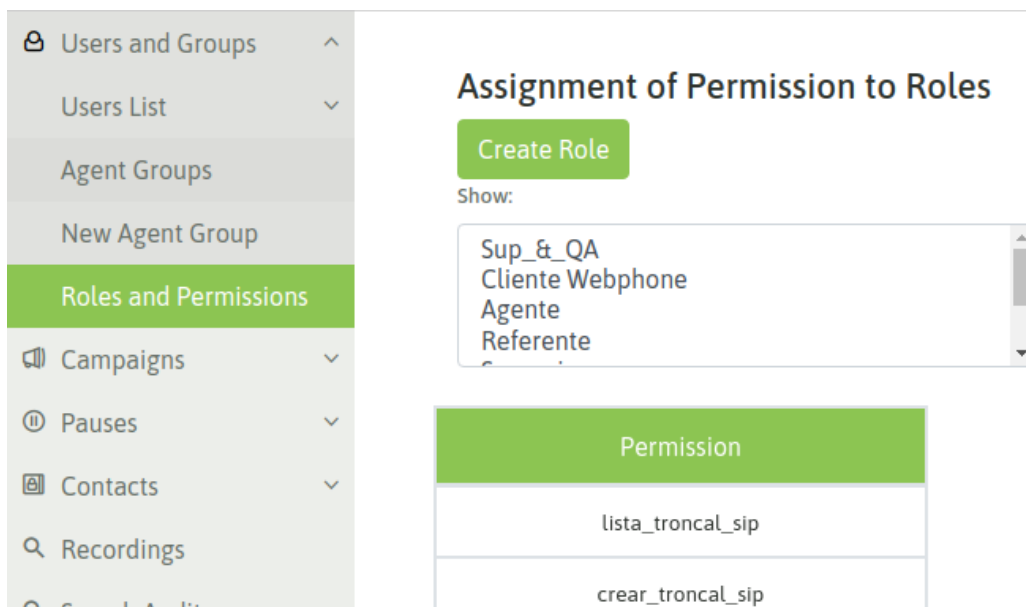
In this section the essential configuration are shown once an instance of OMniLeads is installed

9.1 Initial configuration

The application is already installed. Therefore, the post-installation configuration steps are discussed in this section.

9.1.1 Roles and permissions

In this section, you can list the roles predefined in the system and generate new custom roles, for monitor and/or manage the application, with certain privileges or limitations.



Assignment of Permission to Roles

Create Role

Show:

- Sup_&_QA
- Cliente Webphone
- Agente
- Referente

Permission
lista_troncal_sip
crear_troncal_sip

Below, the predefined roles are listed:

- **Basic supervisor:** the users generated from this role will only be able to list all the campaigns, consult general reports of agents and calls, and access the supervision of agents and campaigns.
- **Supervisor:** the users generated from this role will be able to work with all the campaigns (create, modify, delete and list), access supervision and general reports of calls and agents. They will also be able to create users and agent groups, but can only edit or erase agents that are assigned to a campaign they are assigned, find recordings of their campaigns, list and upload contacts, and access the telephony module where they will be able to work with some sections.
- **Manager:** the users generated from this role will be able to perform all the actions of a supervisor and also view the Audit module.
- **Administrator:** users generated from this role will have access to the entire system. Only administrators can edit or erase with Administrator profile

To create a *custom Role*, you should access the *Users and groups* module, *Roles and permissions* section, then indicate Create role. The new role is assigned a name and then the permissions it will have are marked. In order not to start from scratch, the user can start from a permissions base of a certain profile and then continue personalizing (adding or limiting permissions) until the role is ready.

The creation of a new role is exemplified below:

- Create a new role: create a new role and assign a name to it.

Assignment of Permission to Roles

Create Role

Show:

Sup_&_QA

Cliente Webphone

Agente

Referente

Permission	Sup_&_QA
lista_troncal_sip	<input type="checkbox"/>
crear_troncal_sip	<input type="checkbox"/>
editar_troncal_sip	<input type="checkbox"/>
eliminar_troncal_sip	<input type="checkbox"/>
lista_rutas_salientes	<input type="checkbox"/>
ordenar_rutas_salientes	<input type="checkbox"/>
crear_ruta_saliente	<input type="checkbox"/>
editar_ruta_saliente	<input type="checkbox"/>
eliminar_ruta_saliente	<input type="checkbox"/>
lista_rutas_entrantes	<input type="checkbox"/>
crear_ruta_entrante	<input type="checkbox"/>
editar_ruta_entrante	<input type="checkbox"/>
eliminar_ruta_entrante	<input type="checkbox"/>

Sup_&_QA

▼

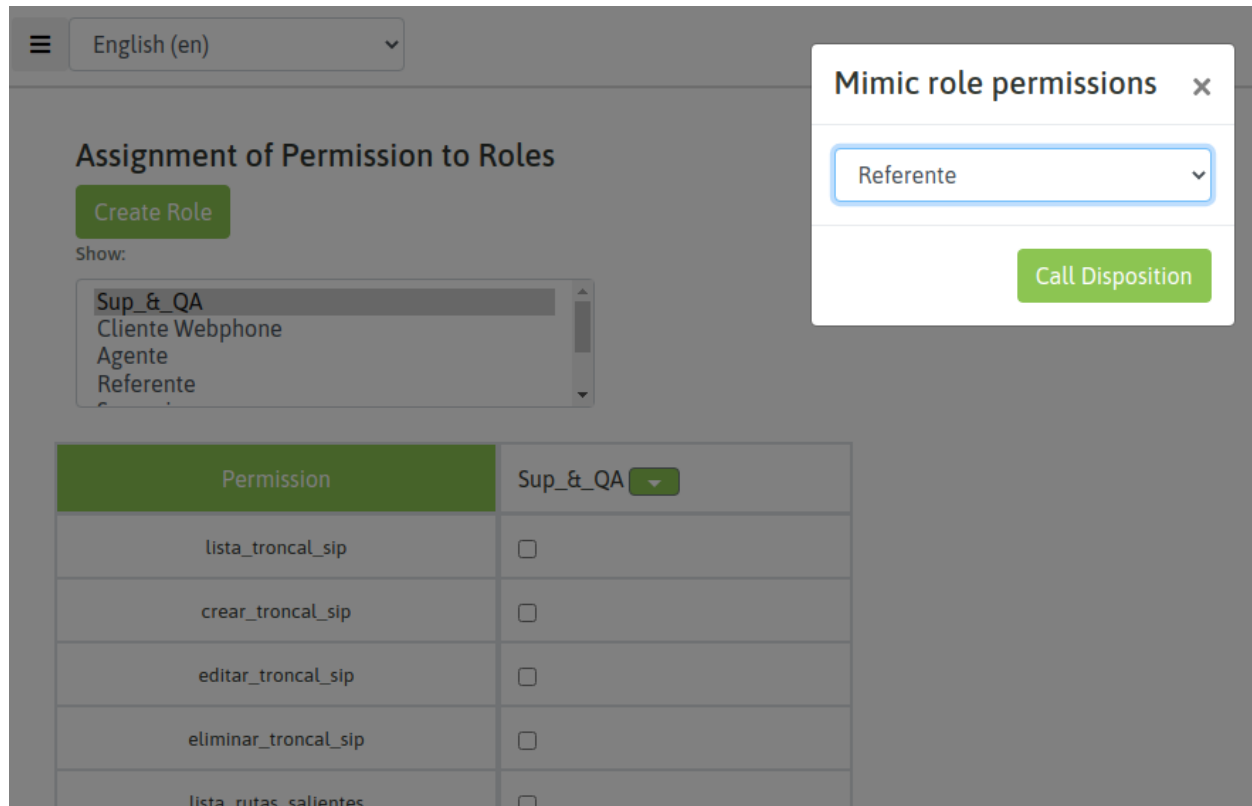
Guardar

Imitar permisos de..

Aplicar permisos de..

Eliminar

One can *imitate permissions from a group* or *apply permissions from an existing group*. The difference is that the first option uses the same permissions, while the second option adds the permissions to those that have already been checked in the new role.



- Save the new role: finally the new role created is saved.

Assignment of Permission to Roles

Create Role

Show:

Sup_ & QA
 Cliente Webphone
 Agente
 Referente

Permission	Sup_ & QA
lista_troncal_sip	<input type="checkbox"/>
crear_troncal_sip	<input type="checkbox"/>
editar_troncal_sip	<input type="checkbox"/>

Guardar
 Imitar permisos de..
 Aplicar permisos de..
 Eliminar

9.1.2 Users

There are two types of users we must differentiate: Agents and Administrative. Agent users are those who manage communications, while Administrative users manage the application.

Important: Before creating an Agent user there must be at least one Group of Agents.

To create a user one must access the section *Users* → *New Users*.

NEW USER: BASIC INFORMATION

Basic Information

Profile

Username:

Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.

First name:

Last name:

Email address:

User role:

Password:

- Your password can't be too similar to your other personal information.
- Your password must contain at least 8 characters.
- Your password can't be a commonly used password.
- Your password can't be entirely numeric.

Password confirmation:

Enter the same password as before, for verification.

Finish

A form will be displayed to be completed with the data of the new user.

9.1.3 Groups of Agents

In this section the agent groups are managed. These groups will be invoked in different modules of the system, such as when assigning users to campaigns, in the extraction of reports or in the supervision module.

Create agents group

To create a user one must access the section *Users* → *New Users*.

NEW AGENT GROUP

Name:

agent

Auto Unpause:

120

In seconds

Auto attend inbound:

☐

Auto attend Dialer:

☒

Force Call Disposition:

☐

Out of campaign Call:

☒

Allow access to recordings:

☒

Allow access to dashboard:

☒

Permitir la activación de On-Hold:

☒

Save

The fields displayed are:

- **Name:** is the name desired to assign to the agents group.
- **Auto Unpause:** To understand this parameter, we should explain that in OMniLeads, after each received call (of any nature), agent is forced to enter into an ACW pause (After Call Work), in which it remains inactive for assigned campaigns, in order for the agent to complete the disposition of the current call. To get out of such status induced by the system, there are 2 possibilities: if we leave the value at «0» the agent should explicitly exit the pause to continue operating. But if instead we set a number (for example, 5 seconds), this implies

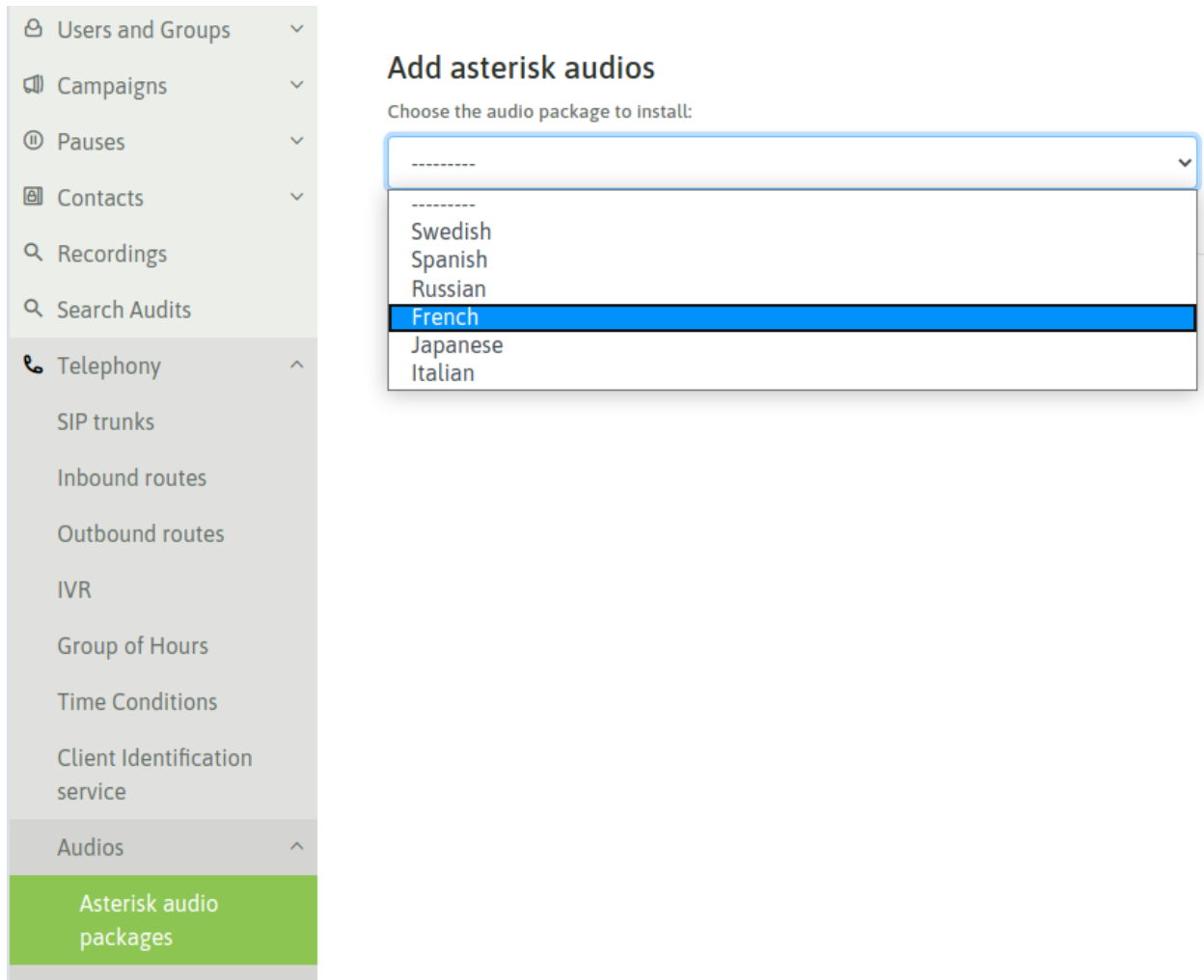
that the system will return the agent to online status after those configured seconds (as indicated in “Unpause Automatically”). This parameter can be managed in the Agent Group Configuration, or from a campaign setup.

- **Auto_attend_inbound:** if this value is checked, the calls coming from inbound campaigns will be connected to the agent without providing the possibility of notification (Ring) nor the option to attend by the agent.
- **Auto_attend_dialer:** if this value is checked, the calls from campaigns with predictive dialer will be connected to the agent without giving the possibility of notification (Ring) nor the option to attend by the agent.
- **Force Call Disposition:** If this value is set, then all calls made or received by the agent will have to be dispositioned.
- **Out of campaign Call:** If this value is set, it will allow the agent to make calls that do not fit into any campaign.
- **Allow access to recordings:** If this value is set, the agent will have access to search, download and listen to their own recordings.
- **Allow access to dashboard:** If this value is set, the agent will have access to his/her own dashboard.
- **Allow activation of On-Hold:** If this value is set, this will allow agents to put a call on hold.

9.1.4 Add audio packages in other languages

The generic audios that the external agents or telephones will listen to come by default in English, being configurable in the incoming or outgoing routes, so that if the telephone channel meets any indication through generic audio within the flow of a call, it could be reproduced according to the indicated language.

If the instance needs to use other languages, they can be installed through the Telephony Module in the Audio packages section, where new languages can be added.



9.1.5 Music on hold

Within the *Telephony* module, the *Music on Hold* section allows one to manage playlists with files in the Wav 16 bit format. The lists generated can be used in incoming campaigns when putting callers in a queue. Playlists that are being used in a campaign or have assigned files cannot be removed.

Telephony ^

Audios ^

Asterisk audio packages

Custom audios

Music on Hold playlists

Help v

MUSIC ON HOLD PLAYLIST

ID	Name	Actions
1	Playlist_1	
2	Playlist_2	

Add

Page 1 of 1.

Once a new list is created, the desired music must be added through .wav format files to be loaded from your computer. Only playlists that have at least one music loaded will be available for use in incoming campaigns.

CREATE MUSIC ON HOLD PLAYLIST: PLAYLIST_1

Music on Hold name	Music file	Actions
Music 1	<div> <div>▶ 0:00 / 0:00</div> <div></div> <div>🔊</div> </div>	

Add Music on Hold

Name:

Audio original:

Choose File

No file chosen

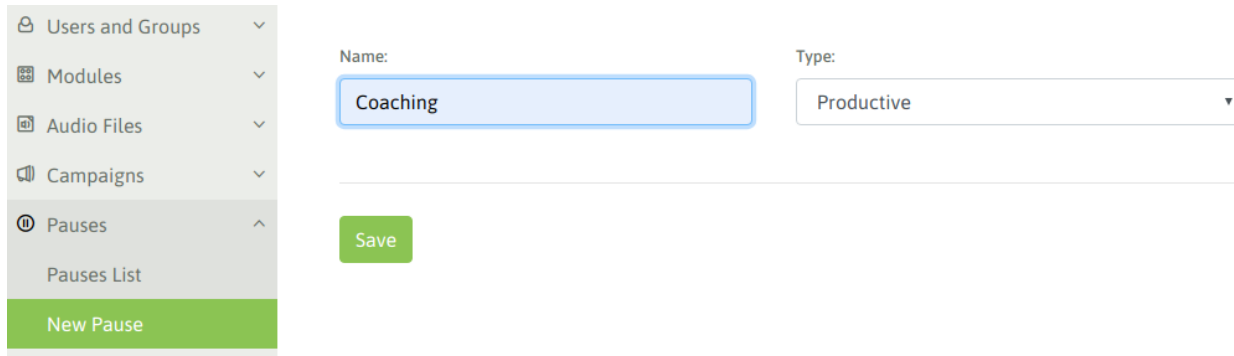
Select the .wav file you want for the Music on Hold

Add

9.1.6 Create agent pauses

The agents can enter a pause whenever they want to be unavailable for the call processing, this prevents that an incoming or dialed campaign assigns a new call. Also pause states are useful for recording productivity and measuring agent session times.

Pauses can be generated by supervisors and are classified as Recreational and Productive pauses.



Users and Groups ▾

Modules ▾

Audio Files ▾

Campaigns ▾

Pauses ▴

Pauses List

New Pause

Name:

Type:

At the time of presenting the agent sesión reports, the totalized pauses are divided in recreative pauses and productive pauses. This allows measure the productivity of our agents in a more exact way.

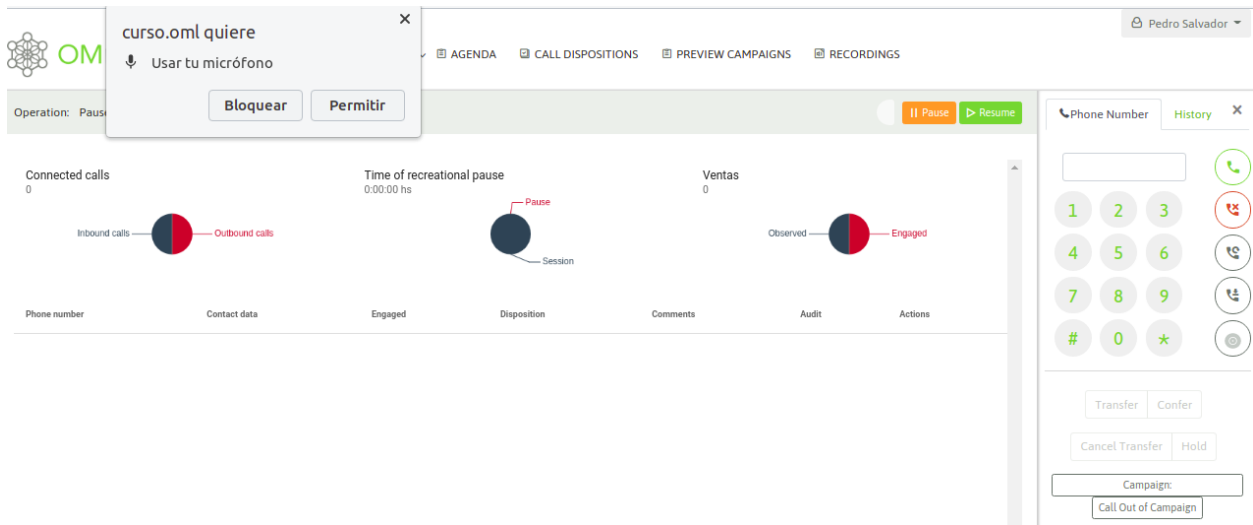
PAUSES

id	Name	Actions
1	Coaching	<input type="button" value="Options ▾"/>
2	Lunch	<input type="button" value="Options ▾"/>
3	Break	<input type="button" value="Options ▾"/>

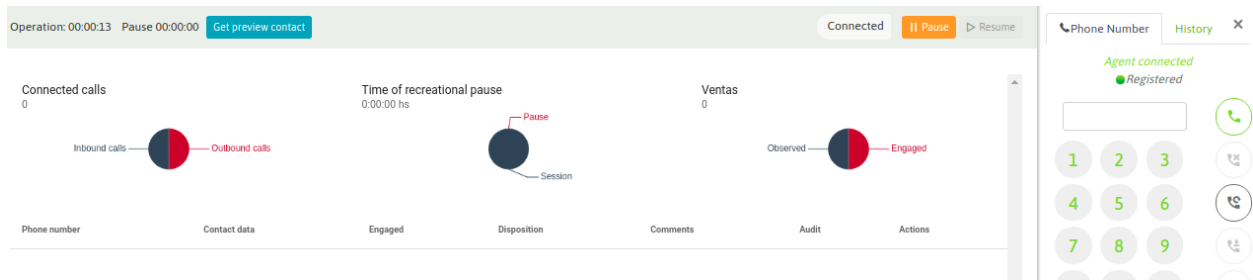
9.1.7 First agent login

Important: Have in mind that to obtain successful login, you must have an **available MICROPHONE:** in the workstation where the agent is going to work. Otherwise the login could be flawed.

Once we access with our agent, if everything goes well we should run into a popup which request the permission to take control of the microphone, ass illustrated in Figure.



When allowing the permission, we should listen to an audio that the system reproduces indicating the successful login and also the agent screen should look like figure.



9.1.8 Instance registry

This step is not mandatory since the system can work perfectly without registering. However, it is necessary to have the registered instance when acquiring an Addon or when subscribing the platform to the manufacturer support.

Finally, for those certified integrators (those who have approved the official OMniLeads certification program), after registering the instance, the installation can be signed with the certification code. Thus, visibilizing that the platform has been deployed and configured by an IT admin certified by the manufacturer.

Users and Groups

Modules

Audio Files

Campaigns

Pauses

Contacts

Recordings

Telephony

Reports

Help

Register

About

Insert your personal name or company name:

Fretech Solutions

Insert your access password:

Insert your email:

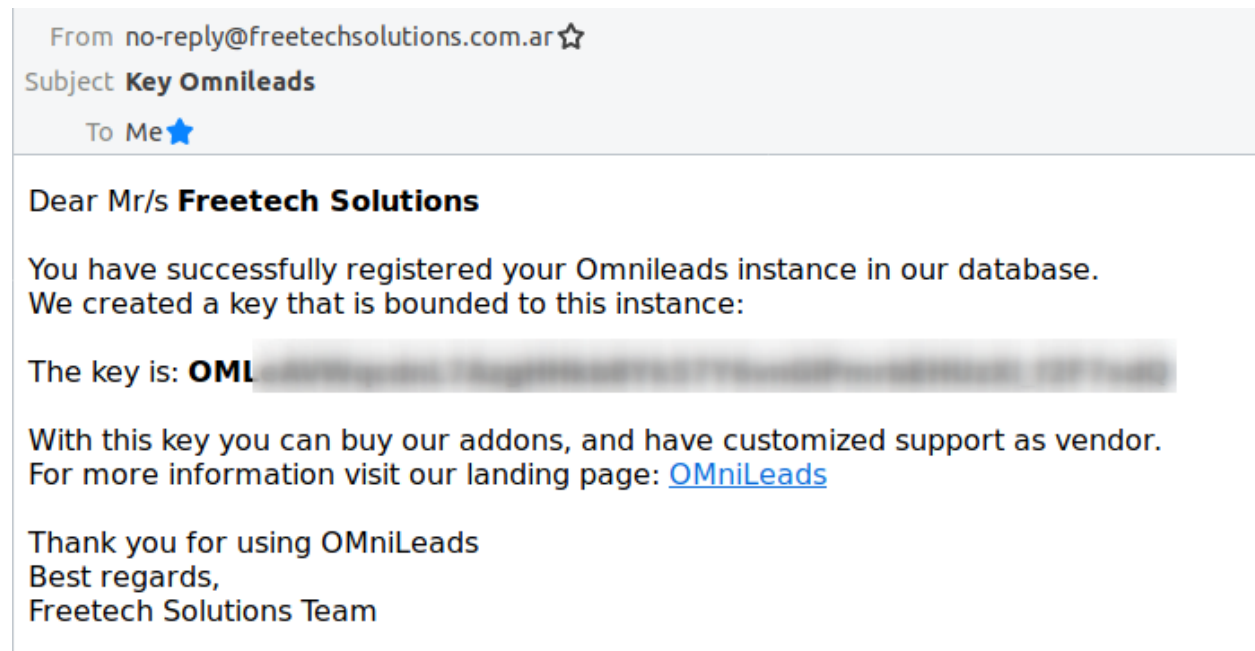
tmckenna@fretechsolutions.com.ar

Insert your phone number:

87636312312

Register

The fields here must be filled after that you will receive an email with the instance key



After that, every time you go to Register section you will see this.

Completed ×
Successful registration, you have been sent an email with your registration key.

Insert your personal name or company name:	Insert your access password:
<input type="text" value="Freetech Solutions"/>	<input type="password"/>
Insert your email:	Insert your phone number:
<input type="text" value="tmckenna@freetechsolutions.com.ar"/>	<input type="text" value="87636312312"/>

Already registered

The registration process asks for required values like username, email address and password, making optional the phone field.

Once the OML instance is successfully registered, an email is sent to the address entered. In case you want to get the email again you can use the “Resend” button.

Is important to note that if you want register several OML instances with the same email address you must enter the same password. In other cases you should use a different email address.

9.1.9 Commercial addons available

The system shows information about the commercial addons available using this page:

Users and Groups

Campaigns

Pauses

Contacts

Recordings

Search Audits

Telephony

Reports

Premium Reports

Supervision


Help

Register

About

Addons

COMMERCIAL ADDONS AVAILABLE




Premium Reports

PREMIUM REPORTS ADD-ON FOR OMNILEADS

1.2.0

Freetech Solutions

<https://www.freetechsolutions.com.ar/>



Webphone and Video Client

WEBPHONE AND VIDEO ADD-ON PARA OMNILEADS

4.0.0

Freetech Solutions

<https://www.freetechsolutions.com.ar/>

From here making click in the addons names you can access to its sites and for adquire them and installing in your current instance.

PSTN ACCESS CONFIGURATION

OMniLeads through web configuration, ease the possibility of maintain SIP trunks to access the PSTN. This trunks are invoked by the routing rules of outbound calls, where it can be specified which tye of calls are processed by each SIP trunk. Also, the trunks can be configured in *failover* mode

To go deeper it's recommended to read the rest of the chapter.

10.1 Telephony module configuration

In this section the necessary configurations to be carried out so that our App can interact with the PSTN (public telephone network) are exposed, so that the agents can take calls from abroad as well as dial calls to external telephones.

Important: OMniLeads only supports SIP as interconnection technology with others telephony switches. Therefore, the integrator will be able to configure SIP trunks from ITSP providers, SIP trunks against PBX systems and/or SIP Trunks against FXO/E1/T1 Gateways.

10.1.1 SIP Trunk Configuration

To access the trunks configuration you should click (*Telephony -> SIP Trunks*) and there add a new PJSIP trunk. A form similar to the one on figure will be displayed.

The screenshot shows a web form for configuring a SIP trunk. It is divided into two columns. The left column contains three fields: 'Trunk Name:' (a text input), 'Caller id:' (a text input), and 'SIP details:' (a large text area). The right column contains four fields: 'Maximum number of channels:' (a text input with '1000' pre-filled), 'Register String:' (a text input), 'Tecnologia:' (a dropdown menu with 'pjsip' selected), and 'Cargar template:' (a group of six green buttons: 'Internet SIP Trunk', 'Dedicated SIP Trunk', 'PBX OmniLeads (LAN)', 'PBX OmniLeads (NAT)', 'OMniLeads inside PBX', and 'Custom').

The form blanks are:

- **Trunk Name:** the name of the trunk. Must be alphanumeric without spaces or special characters (eg Trunk_provider_a).
- **Number of channels:** is the number of channels allowed by the link.
- **Caller id:** the number with which the calls will go through the trunk.
- **SIP details:** SIP parameters are provided in this text field using Asterisk PJSIP configuration Wizard syntax.

Here are some suggested templates for the types of scenarios presented as typical OMniLeads use cases.

General SIP Trunk parameters

As well anticipated PJSIP is the module that implements SIP for this kind of trunks. But also the syntax chosen to generate the configuration at the Asterisk conf is [pjsip wizard](#).

Además del link citado de la documentación oficial de Asterisk, compartimos el [siguiente link](#) que resulta de mucho interés para profundizar respecto a los parámetros disponibles dentro de pjsip wizard.

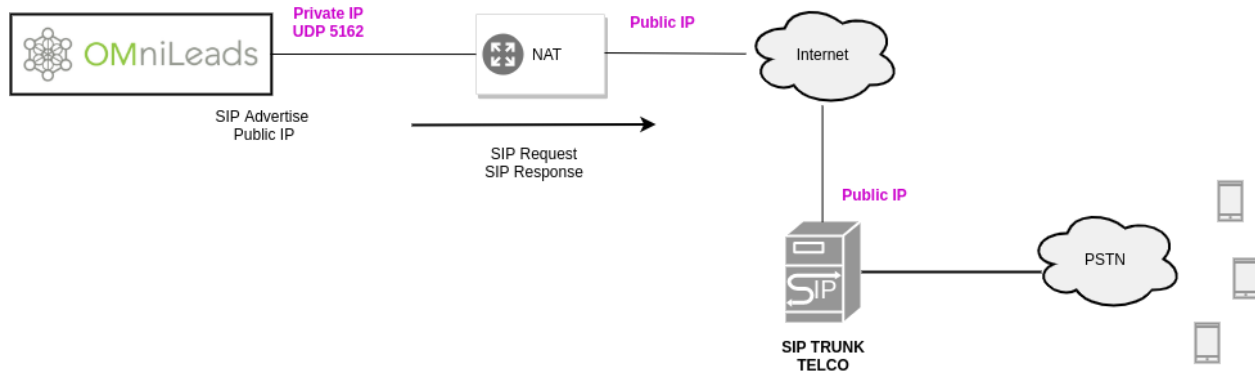
Important: We must remember very well that OMniLeads does NOT use port 5060 as the port for SIP trunks. Port 5060 is used by Kamailio in its WebRTC work in sessions against agents. When generating a SIP trunk between OML and a SIP or PBX provider, we must know that the ports to be used are and vary according to the scenarios described below.

- **Port 5161** for trunks where we must NOT warn the public IP. That is, where there is no NAT involved.
 - **Port 5162** for trunks where we MUST warn the public IP, since they will affect with NAT the SIP packets that leave OML.
 - **Port 5163** for trunks used in OML architecture over Docker containers.
-

The following scenarios are proposed for OMniLeads instances and their connection to a SIP provider with termination to the PSTN.

OMniLeads behind NAT

Under this scenario we have the possibility of the App deployed on a Cloud provider going out to the Internet through a gateway. The NAT also affects the deployment of an on-premise instance in a corporate LAN in which the Internet is routed through the company's router using a Public IP to perform the NAT on the packets generated from OMniLeads to the SIP provider.



Here, Asterisk uses the UDP port 5162 since it is the port that implements the fact of warning in the REQUEST or RESPONSE SIP the Public IP with which the packets will go abroad and reach the other end of the SIP trunk. This adaptation of the mentioned packets is done at the SIP (signaling) and SDP (media negotiation) levels. Therefore, the external recipient of the packets does not realize that our Asterisk is behind NAT. The packets assembled arrive with the public IP of the network device that performs NAT on the sent packets.

For this scheme we analyze the [PJSIP configuration Wizard](#) template that is proposed to complete with your data:

```

type=wizard
transport=trunk-nat-transport
accepts_registrations=no
accepts_auth=no
sends_registrations=yes
sends_auth=yes
endpoint/rtp_symmetric=yes
endpoint/force_rport=yes
endpoint/rewrite_contact=yes
endpoint/timers=yes
aor/qualify_frequency=60
endpoint/allow=alaw,ulaw
endpoint/dtmf_mode=rfc4733
endpoint/context=from-pstn
remote_hosts=****IPADDR-or-FQDN:PORT****
outbound_auth/username=****YOUR SIP_USERNAME****
outbound_auth/password=****YOUR SIP_PASSWORD****

```

The last three parameters have to do with the data that the provider provides us when contracting the service. That is, the address or FQDN and corresponding port of its SIP Server where to shoot our REGISTER to register the trunk or INVITE when sending outgoing calls. In addition, the username and password values with which the provider authenticates those REQUEST are available.

Regarding the rest of the parameters, we will emphasize:

```
transport=trunk-nat-transport
```

This parameter indicates the Asterisk PJSIP stack that it must warn the public IP and public port with which the SIP packets will leave when reaching the provider's SIP-Server.

The next 4 parameters allude to the fact that typically under this scheme OMniLeads does not request authentication from the SIP provider in case of incoming calls, but must authenticate when sending calls to the provider and that it must also send a recurring register to be able to be reached by the SIP provider when connecting incoming calls. We are specifically talking about the parameters and their values:

```
accepts_registrations=no
accepts_auth=no
sends_auth=yes
sends_registrations=yes
```

The following three parameters have to do with the codecs to use, the protocol used for the DTMF exchange. Finally the entry point (dialplan context) of the calls that arrive through the trunk.

```
endpoint/allow=alaw, ulaw
endpoint/dtmf_mode=rfc4733
endpoint/context=from-pstn
```

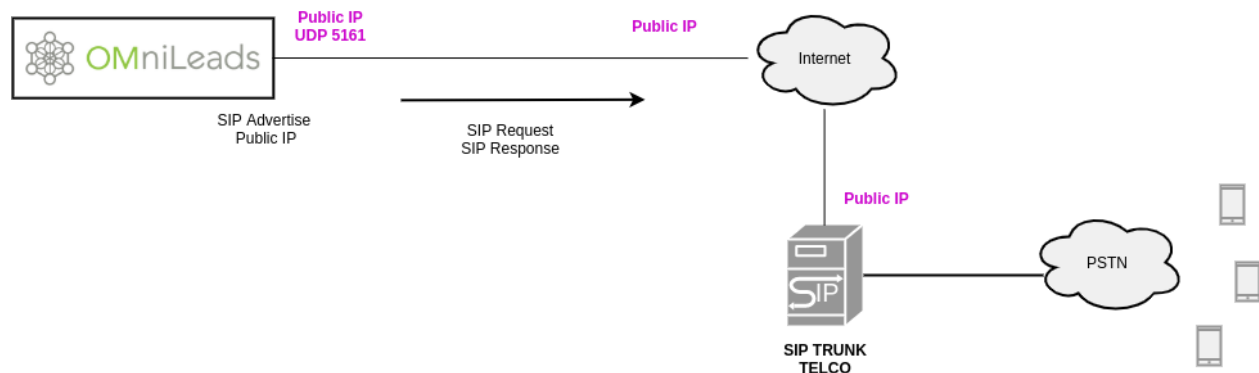
Important: When declaring the SIP trunk at the other end, keep in mind that OMniLeads will use the SIP UDP port 5162 in these NAT environments.

OMniLeads in environments without NAT

Under this scenario, we have the possibility of the App deployed on a VPS with a public IP whose SIP provider is also on a public IP. Therefore, there is no NAT. This scenario includes a deployment performed on a corporate LAN (on premise) connecting to the Internet through the company's Router or using a SBC or PSTN-GW, which is in charge (among other things) of the NAT issue.

SIP trunk on the Internet

As in the previous item, a SIP provider available on the Internet is proposed. Its public IP is now reached without the affection of the NAT, since OMniLeads is available with a public IP. Just as before, the provider facilitates us with the IP or FQDN of the SIP Server to which we must send all the REQUEST, on the one hand, and a username and password to authenticate them, on the other.



For this scheme we analyze the [PJSIP configuration Wizard](#) template that is proposed to complete with your data:

```
type=wizard
transport=trunk-transport
accepts_registrations=no
```

(continues on next page)

(continued from previous page)

```

accepts_auth=no
sends_registrations=yes
sends_auth=yes
endpoint/rtp_symmetric=yes
endpoint/force_rport=yes
endpoint/rewrite_contact=yes
endpoint/timers=yes
aor/qualify_frequency=60
endpoint/allow=alaw,ulaw
endpoint/dtmf_mode=rfc4733
endpoint/context=from-pstn
remote_hosts=****IPADDR-or-FQDN:PORT****
outbound_auth/username=****YOUR SIP_USERNAME****
outbound_auth/password=****YOUR SIP_PASSWORD****

```

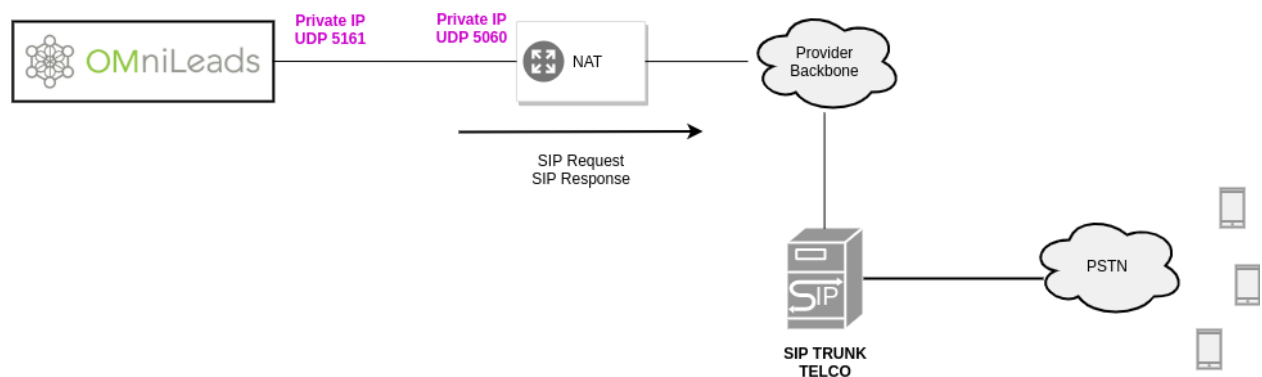
The only parameter that changes with in regard to the examples with NAT is:

```
transport=trunk-transport
```

Where the use of a PJSIP transport is indicated, where the packets simply flow through the UDP 5161 port without any NAT treatment.

Corporate SIP Trunk

Under this classification we have SIP link providers that arrive with their own connectivity backbone to the physical location where the data center is located. In this scenario it is frequent that the provider does not request authentication or registration. Besides, when making calls on the provider's private backbone the NAT issue is no longer a problem to be solved.



For this scheme we analyze the [PJSIP configuration Wizard](#) template that is proposed to complete with your data:

```

type=wizard
transport=trunk-transport
accepts_registrations=no
accepts_auth=no
sends_registrations=no
sends_auth=no
endpoint/rtp_symmetric=no
endpoint/force_rport=no
endpoint/rewrite_contact=no
aor/qualify_frequency=60
endpoint/allow=alaw,ulaw

```

(continues on next page)

(continued from previous page)

```

endpoint/dtmf_mode=rfc4733
endpoint/timers=yes
endpoint/language=es
endpoint/context=from-pstn
remote_hosts=****IPADDR-or-FQDN:PORT****

```

The last two parameters have to do with the data that the provider facilitates us. That is, the IP / FQDN address and corresponding port where we should fire our REQUEST. Keep in mind that under this scheme we assume that the SIP provider does not authenticate us via SIP, therefore we do not use username or password.

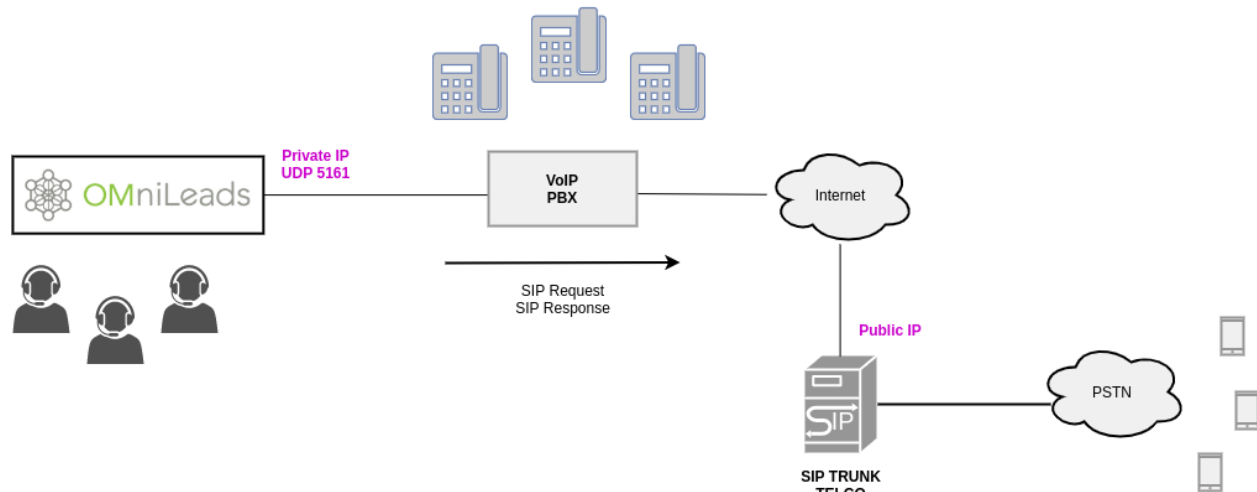
Once again the transport = trunk-transport is used, implying that NAT is not affected.

The rest of the parameters were already discussed in the previous case.

OML SIP trunk with PBX on LAN

A highly implemented scheme has to do with the SIP trunk connection between OMniLeads and the company's PBX. Under this modality, access to the PSTN is provided by the central PBX. The outgoing calls to the PSTN are made through the SIP trunk to the PBX and then the latter is in charge of routing the calls to the specific destinations through their links to the PSTN.

Under this configuration, a company can deploy a Contact Center App fully integrated with its central PBX.



The template [PJSIP configuration Wizard](#) that is proposed to be completed according to the configuration generated on the IP-PBX side is:

```

type=wizard
transport=trunk-transport
accepts_registrations=no
sends_auth=yes
sends_registrations=no
accepts_auth=yes
endpoint/rtp_symmetric=no
endpoint/force_rport=no
endpoint/rewrite_contact=no
endpoint/timers=yes
aor/qualify_frequency=60
endpoint/allow=alaw,ulaw

```

(continues on next page)

(continued from previous page)

```

endpoint/dtmf_mode=rfc4733
endpoint/context=from-pbx
remote_hosts=****IPADDR-or-FQDN:PORT****
inbound_auth/username=****SIP_USER PBX -> OML****
inbound_auth/password=****SIP_PASS PBX -> OML****
outbound_auth/username=****SIP_USER OML -> PBX****
outbound_auth/password=****SIP_PASS OML -> PBX****
endpoint/from_user=****SIP_USER OML -> PBX****

```

Outgoing calls (from OMniLeads to the PBX) and incoming calls (from the IPPBX to OMniLeads) are proposed to be authenticated via SIP. That explains the following parameters and their values:

- **sends_auth=yes**
- **accepts_auth=yes**
- **remote_hosts=****IPADDR-or-FQDN:PORT******
- **inbound_auth/username=****SIP_USER PBX -> OML******
- **inbound_auth/password=****SIP_PASS PBX -> OML******
- **outbound_auth/username=****SIP_USER OML -> PBX******
- **outbound_auth/password=****SIP_PASS OML -> PBX******
- **endpoint/from_user=****SIP_USER OML -> PBX******

We take for granted the interpretation of the parameters from their suggestive names. We also highlight the fact of not involving any SIP registration -either from OMniLeads to the PBX or the other way around- since both systems are on a LAN network and with an IP address or assigned FQDN.

On the other hand, the parameters **transport=trunk-transport** and **endpoint/force_rport=no** not tell us that any type of NAT treatment is applied to the SIP packets generated from OMniLeads.

Finally the parameter related to context **endpoint/context=from-pbx** indicates that the calls from the PBX have an access point different that calls from PSTN, because using the context from-pbx you can call and transfer between OMniLeads agents and PBX extensions.

Important: When declaring the SIP trunk at the other end, keep in mind that OMniLeads will use SIP UDP 5161 port in these NAT-free environments.

PJSIP custom trunk

Here the Administrator will be able to customize his or her own PJSIP wizard configuration. Beyond the templates provided, the Administrator always has the possibility of adjusting the configuration according to the specific scenario and to the particularities of each case. It is therefore highly recommended to fully comprehend the parameters of the Asterisk PJSIP stack due to their high level of customization.

10.1.2 Outbound calls routing configuration

OMniLeads allows to manage the outbound call routing on several SIP trunks (created previously), so using criteria like the *length or number prefix* to determine which SIP link use to route the call. It is also possible to maintain a *failover* logic in terms of SIP trunk.

To access the ABM of outbound routes enter the menú point (*Telephony -> Outbound routes*)

Outbound route creation

Name: Ring time:
In seconds

Dial options:

Dial patterns

Prepend:	Prefix:	Dial pattern:	
<input type="text"/>	<input type="text" value="99"/>	<input type="text" value="1XXX"/>	<input type="button" value="Remove"/>
<input type="text"/>	<input type="text"/>	<input type="text" value="0[1-3]XXXXXXXX"/>	<input type="button" value="Remove"/>
<input type="text"/>	<input type="text"/>	<input type="text" value="15[2-9]XXXXXXX"/>	<input type="button" value="Remove"/>
<input type="text"/>	<input type="text"/>	<input type="text" value="0[1-3]XXXXXXXXXXXX"/>	<input type="button" value="Remove"/>

Trunks sequence

Figure 2: Outbound route

- **Nombre:** Es el nombre de la ruta (alfanumérico, sin espacios ni caracteres especiales).
- **Ring time:** is the time (in seconds) that calls made of this route, will attempt to stablish a conection with the destination, before continuing to try the next trunk or discard the call.
- **Dial options:** are the “dial” application options used by Asterisk(r) in a low level.
- **Dialing patterns:** By patterns, you can represent the *types of calls* that will be accepted by the route, and thus placed over the SIP trunk to finally reach the desired destination.

To understand how the digits with patterns are represented, it is recommended to read this link: <https://www.voip-info.org/asterisk-extension-matching/>.

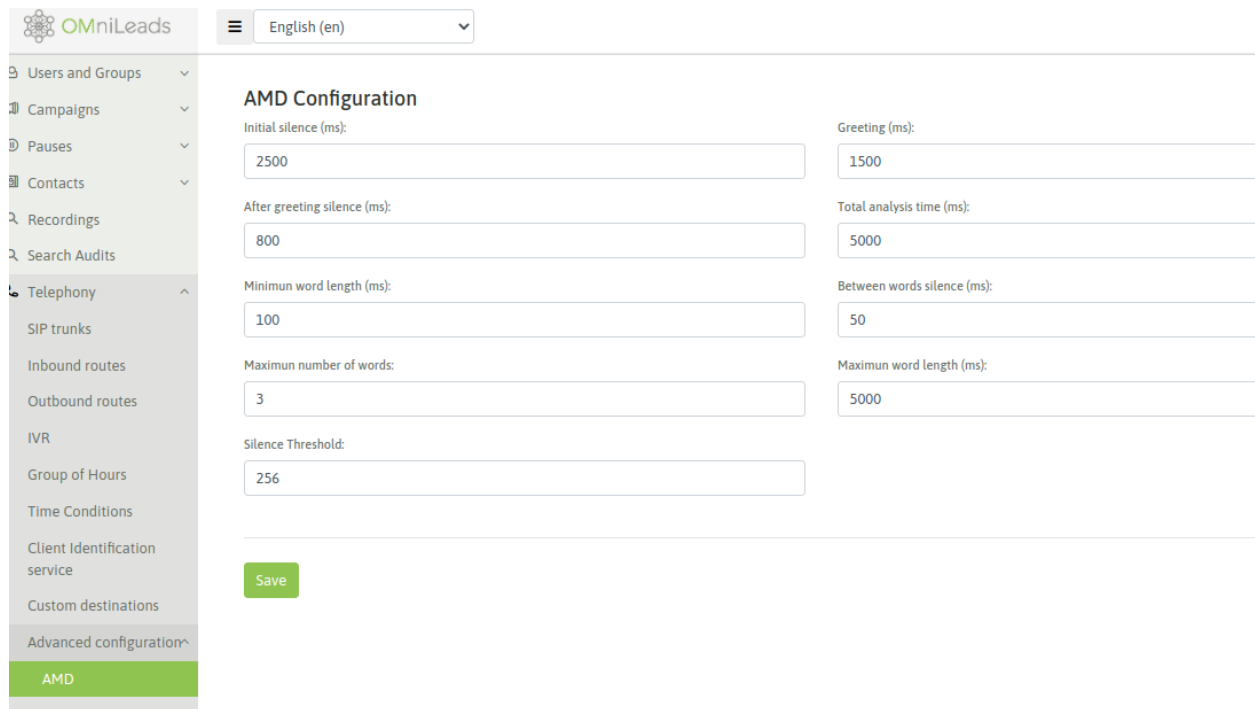
<https://www.voip-info.org/asterisk-extension-matching/>

Dentro de cada patrón ingresado, hay 3 campos:

- **Prepend:** are the digits sent to the SIP trunk as additional to the dialed number. In other words, they arrive at the Trunk positioned in front of the dialed number.
 - **Prefix:** are the digits that can come as a “prefix” of a dialed call and will be removed at the time of sending them by the SIP Trunk.
 - **Dial pattern:** this field seeks to represent the patern of authorized digits that the route will proccess for and send to a SIP trunk to take the call out.
- **Trunks sequence:** are the SIP trunks on which the outbound route will attemptr to stablish the dialed call by OML. If the call fails in a trunk, will attempt again in the next one.

10.1.3 AMD Module configuration

The Asterisk AMD module, can be configured in OMniLeads using the following interface:



OMniLeads English (en) ▼

- Users and Groups ▼
- Campaigns ▼
- Pauses ▼
- Contacts ▼
- Recordings
- Search Audits
- Telephony ▲
 - SIP trunks
 - Inbound routes
 - Outbound routes
 - IVR
 - Group of Hours
 - Time Conditions
 - Client Identification service
 - Custom destinations
 - Advanced configuration^
 - AMD**

AMD Configuration

Initial silence (ms): <input type="text" value="2500"/>	Greeting (ms): <input type="text" value="1500"/>
After greeting silence (ms): <input type="text" value="800"/>	Total analysis time (ms): <input type="text" value="5000"/>
Minimum word length (ms): <input type="text" value="100"/>	Between words silence (ms): <input type="text" value="50"/>
Maximum number of words: <input type="text" value="3"/>	Maximum word length (ms): <input type="text" value="5000"/>
Silence Threshold: <input type="text" value="256"/>	

Figura 2: Página de configuración de AMD

For more information about this Asterisk module, refer to next link:

https://wiki.asterisk.org/wiki/display/AST/Application_AMD

10.1.4 Configuration scheme for call recordings filenames

The following section allowd configure the format that will have the names of recordings files on the system

The selection of some of the options shown in the page will impact directly on the name of recording files

Important: Note: notice that the Agent ID as a variable is not available for inbound campaigns or dialer campaigns

The screenshot shows the Omnileads web interface. On the left is a sidebar with a menu including 'Users and Groups', 'Campaigns', 'Pauses', 'Contacts', 'Recordings', 'Search Audits', 'Telephony' (expanded), and 'Advanced configuration'. Under 'Telephony', 'Recordings scheme' is highlighted. The main content area is titled 'EDIT RECORDINGS FILES NAME SCHEME'. It contains a form with the following fields and checkboxes:

- Id contact: ☐
- Date: ☒
- Phone contact: ☐
- Id Campaign: ☒
- Id external contact: ☒
- Id external campaign: ☐
- Id agent: ☒

A green 'Save' button is located below the 'Id agent' checkbox.

Figure 2: Recordings scheme page

10.1.5 Inbound calls routing configuration

The operation of incoming calls will be covered in the Incoming Campaigns section of this documentation since in order to operate with said module we should at least have created some object (IVR, time conditional, incoming campaign, etc.) to where to route each generated DID.

Inbound routes.

The processing of communications between the “world” and an OMniLeads’s agent is encapsulated in a campaign. In this chapter all about the management of Inbound and Outbound (manual, preview and dialer) campaigns is covered.

11.1 Telephone Campaigns

A campaign represents a Contact Center operation that groups:

- An agent group processing phone calls in one direction (outbound or inbound)
- A contact base associated with the campaign.
- A dispositions list that are displayed when classifying the call processed by the agent.
- One or more forms to be launched in case the agent assigns a *engage disposition* associated with any of the forms on the call in progress the form is displayed by said engage disposition and the agent can complete it according to the contact details of the ongoing call.

Figure 1 shows a campaign and its components.

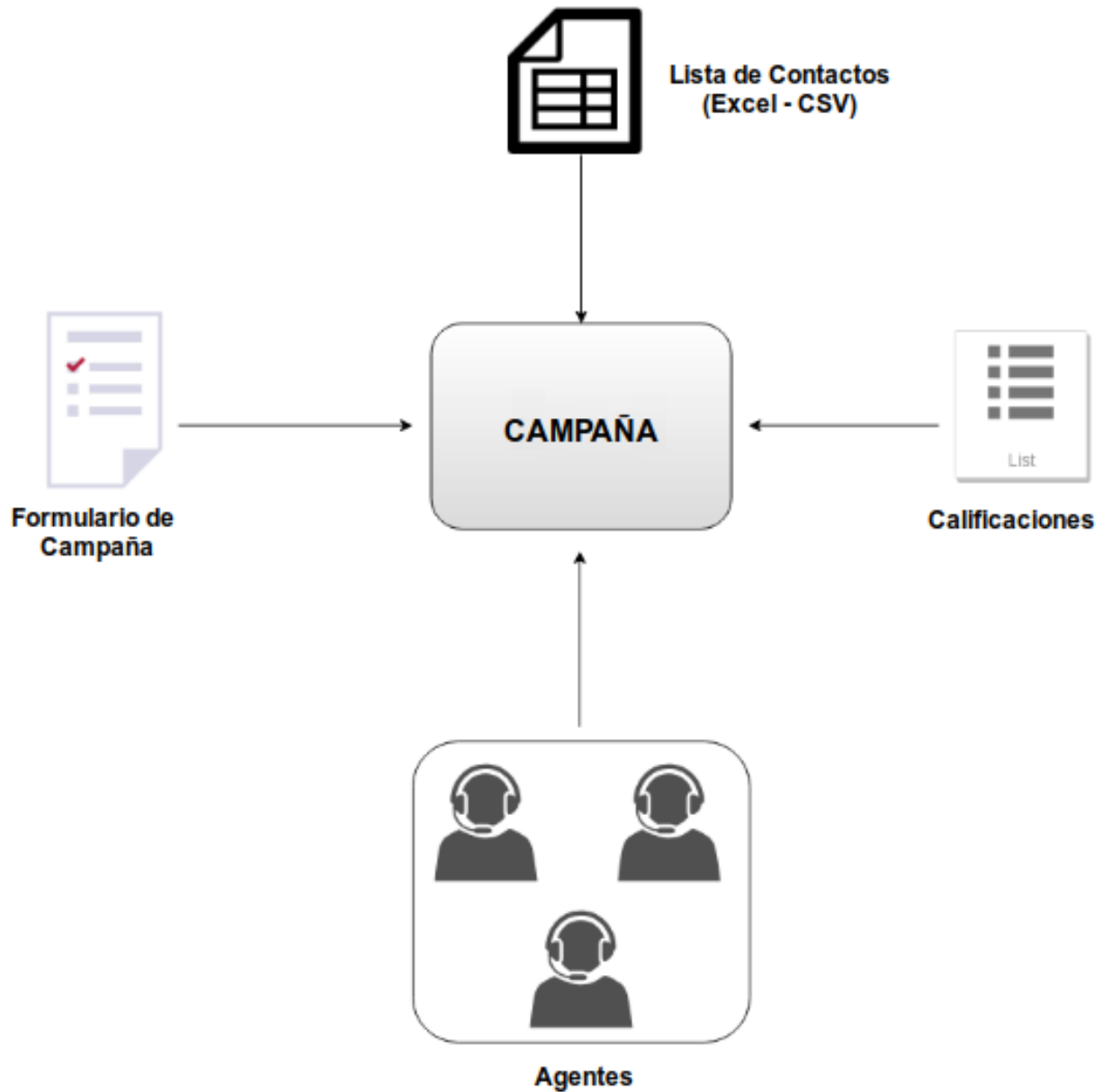


Figure 1: Campaign elements

The fact of working with campaigns allows *Supervisor or Administrator* user profiles to focus metrics and reports, as well as real-time monitoring or search for recordings, among other actions, using campaigns as a filtering criteria. We can have different campaigns of different nature (predictive, preview, incoming or manual), living simultaneously in OMniLeads and leaving records about the calls transacted by the agents.

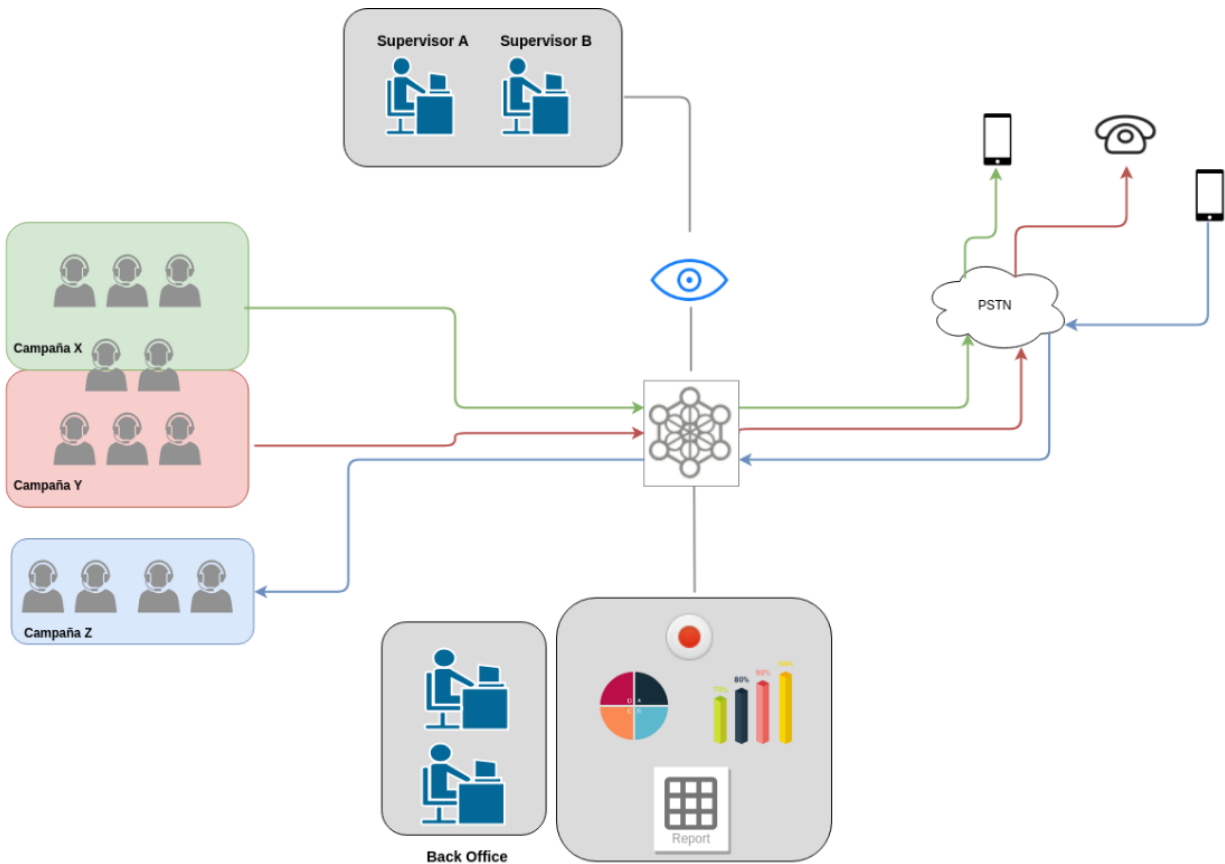


Figure 2: Campaigns, agents, supervisors and backoffice

11.1.1 Dispositions

Dispositions form a list of *tags* available to be assigned to any campaign, so that agents can classify each call within a campaign, with any of these dispositions.

Dispositions are defined by the supervisor or administrator and can be related to several aspects, for example:

- The telephone call result (busy, no answer, wrong number, voicemail, etc.)
- The result of an interaction in a call answered (sale, survey completed, schedule call, etc.)

Dispositions can be totally arbitrary and to generate them you must enter the screen *Campaigns* → *Call Dispositions* → *New Call Dispositions**

We can list the ratings generated within *Campaigns* → *Call Dispositions* → *Call Dispositions*

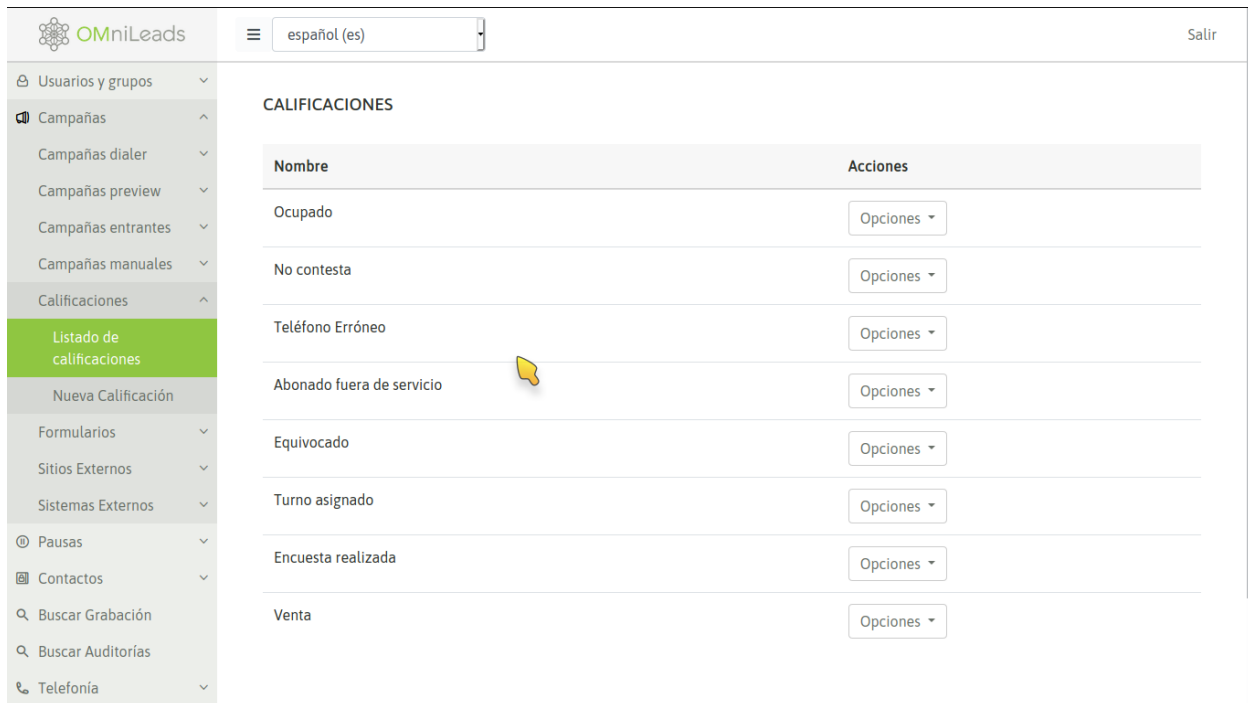


Figure 3: Dispositions

11.1.2 Campaign contact database

Contact databases are a requirement for outbound campaigns although they can also be used by incoming campaigns. In the framework of outgoing campaigns, the contact database attached to the campaign, provides the data required by the predictive dialer or preview; while, in the incoming campaigns, it provides the full information available on the calling number that will be displayed in the agent's screen when receiving a call.

These databases are provided in files with CSV format with the fields separated with commas, and also generated in UTF-8 encoding (exclusive requirement). There should be at least one column that contains a phone number for each contact (record) in the file, while the rest of the columns can contain any data (generally each registry has complementary data to the main telephone number). These data is exposed on the agent screen when establishing a communication between both (agent and database contact).

```
tel1,tel2,tel3,name,surname,id
4149090,5149095,6149097, Terence, McKenna, 1000001
4149014,5149015,6149017, Timothy, Leary, 1000002
4149020,5149025,6149027, Albert, Hoffman, 1000003
4149032,5149095,6149037, Aldous, Huxley, 1000004
4149040,5149092,6149047, Herman, Hesse, 1000005
4149052,5149092,6149055, Jorge Luis, Borges, 1000006
4149050,5149092,6149067, Jiddu, Krishnamurti, 1000007
4149050,5149090,6149077, Aleister, Crowley, 1000008
4149050,5149090,6149087, Facundo, Cabral, 1000009
4149052,5149090,6149090, Atahualpa, Yupanqui, 1000010
4149055,5149095,6149000, Carl Gustav, Jung, 1000011
4149050,5149090,6149010, Isaac, Luria, 1000012
4149057,5149095,6149026, Robert Anton, Wilson, 1000013
```

Figure 4: CSV Contact file - Text editor View

	A	B	C	D	E	F
1	tel1	tel2	tel3	name	surname	id
2	4149090	5149095	6149097	Terence	McKenna	1000001
3	4149014	5149015	6149017	Timothy	Leary	1000002
4	4149020	5149025	6149027	Albert	Hoffman	1000003
5	4149032	5149095	6149037	Aldous	Huxley	1000004
6	4149040	5149092	6149047	Herman	Hesse	1000005
7	4149052	5149092	6149055	Jorge Luis	Borges	1000006
8	4149050	5149092	6149067	Jiddu	Krishnamurti	1000007
9	4149050	5149090	6149077	Aleister	Crowley	1000008
10	4149050	5149090	6149087	Facundo	Cabral	1000009
11	4149052	5149090	6149090	Atahualpa	Yupanqui	1000010
12	4149055	5149095	6149000	Carl Gustav	Jung	1000011
13	4149050	5149090	6149010	Isaac	Luria	1000012
14	4149057	5149095	6149026	Robert Anton	Wilson	1000013

Figure 5: CSV Contact File - View in libreoffice excel

A contact base (csv) is available, to proceed with the the loading of the file into the system by accessing the menu item; *Contacts* → *New contacts database*

NUEVA BASE DE DATOS DE CONTACTO

Nombre:

new_contacts1

Archivo de importación:

Examinar...

No se ha seleccionado ningún archivo.

Definir Base Datos

bajar base de datos de ejemplo

Carga de archivos

Recientes

Carpeta personal

Escritorio

Descargas

miguel Contactos-BD

Nombre

oml-example-db.csv

Figure 6: New contact database

It should be indicated with a check, which columns are the ones that store phones, as indicated in figure 7.

telefono	name	surname	id_contact	tel2	tel3	tel4
4553101	Terence	McKenna	30000001	55001101	66001102	77001103
4553102	Timothy	Leary	30000002	55001102	66001103	77001104

Campos de teléfono:

- ☒ telefono
- ☐ name
- ☐ surname
- ☐ id_contact
- ☒ tel2
- ☒ tel3
- ☒ tel4

Id externo:

id_contact

Volver a nueva base de datos

Guardar Base Datos

Figure 7: Telephone Selection

Finally, the form is saved and the CSV file is now available as a system contact database available for any campaign.

11.1.3 Restriction configuration for Contacts fields

Sometimes Agents should not be able to see or edit certain fields of the database contacts. For this cases, restrictions over this fields can be set entering the option “Restrict contacts fields” from the campaign’s list.

The screenshot shows the OMniLeads web interface. On the left is a sidebar menu with options like 'Usuarios y grupos', 'Campañas', 'Campañas dialer', 'Campañas preview', 'Listado de campañas', 'Nueva campaña', 'Templates', 'Nuevo template', 'Campañas entrantes', 'Campañas manuales', and 'Calificaciones'. The 'Listado de campañas' option is selected. The main area displays 'CAMPANAS PREVIEW' with two sections: 'Campañas Activas' and 'Campañas Finalizadas'. The 'Campañas Activas' section shows a table with columns 'Id', 'Nombre', and 'Estado'. There is one active campaign with Id 7, named 'Encuesta', and in 'Activa' state. The 'Campañas Finalizadas' section shows a table with columns 'Id', 'Nombre', 'Estado', and 'Acciones', and a message 'No existen campañas'. On the right side, a dropdown menu is open, showing various actions: 'Modificar', 'Agregar agentes', 'Dar permiso para ver', 'Calificaciones', 'Resultados de Base', 'Reportes', 'Ver Contactos Asignados', 'Ordenar entrega contactos', 'Agregar Contacto', 'Agregar Contactos', 'Restringir campos de contacto' (highlighted with a green box), 'Ver Agendas', 'Configuraciones para agentes', and 'Eliminar'. The top right corner has a 'Salir' button.

Restriction configuration for Contacts fields

In this screen the fields to block or hide to the agents must be selected. Only the fields selected as blocked can be selected as hidden. The main phone field can’t be set as hidden.

Restringir campos de contactos para Agentes

Bloquear telefono: ☐

Bloquear name: ☐ Ocultar name: ☐

Bloquear surname: ☒ Ocultar surname: ☐

Bloquear id_contact: ☒ Ocultar id_contact: ☒

Bloquear tel2: ☐ Ocultar tel2: ☐

Bloquear tel3: ☐ Ocultar tel3: ☐

Bloquear tel4: ☐ Ocultar tel4: ☐

Figure 9: Selecting contact fields to block or hide

When an Agent is to edit a contact the hidden fields will not be present and the fields set as blocked (and not hidden) will be disabled for edition. When creating a contact, the fields set only as blocked will be enabled for only this first time.

OMniLeads DASHBOARD CONTACTOS AGENDAS CALIFICACIONES CAMPAÑAS PREVIEW GF

Operación: 00:36:18 Pausa: 00:00:00 Obtener contacto preview Conectado Pausar Reaundar Teléf

Encuesta-12/7

telefono: 6553103 6553103

name: Isaac

surname: Luria

tel2: 55001113 55001113

tel3:

Calificación: -----

Observaciones:

Speech: Esta es una prueba

1 4 7 #

Figure 10: Restricted and hidden contact fields

11.1.4 Forms

Campaign forms constitute the default method to collect information (relevant to the campaign) in the interaction with the person behind an established communication. They are designed by the user combining in a static view different types of fields (text, date, multiple selection and text area), being able to order them according to the need of the campaign.

To create forms you must access the menu item; *Campaigns* → *New form*

Campaign forms may contain field types like

- **Text**
- **Date**
- **List**
- **Text Area Box**

Figure 8 exemplifies a campaign form creation.

Nuevo Formulario **Campos del formulario** Vista previa del formulario

Nuevo campo

Nombre de campo: Tipo: Campo requerido: ☐

[Agregar campo](#)

[Agregar item](#) [Eliminar seleccionados](#)

Muy insatisfecho

Insatisfecho

Neutral

Satisfecho

Campos del formulario

Orden	Nombre de campo	Tipo	Campo requerido	Valores de la lista
No se encontraron campos creados.				

Figure 11: New campaign form

We can generate a sample satisfaction survey form with the appearance of figure 9.

[Nuevo Formulario](#) [Campos del formulario](#) [Vista previa del formulario](#)

Calidad_de_Servicio:

Muy insatisfecho

Calificación_del_representante_de_servicio_al_cliente:

Muy insatisfecho

Conocimientos_del_representante_de_servicio_al_cliente:

Muy insatisfecho

Tiempo_de_espera_hasta_que_respondieron_mi_consulta:

Muy insatisfecho

Tiempo_que_le_tomó_al_representante_en_resolver_mi_problema:

Muy insatisfecho

Comentarios:

Finalizar

Figure 12: Survey Form

11.1.5 Campaigns, Dispositions & Forms

To explain the relationship between these components, we must remember that multiple forms can be assigned to a campaign. The idea is that different dispositions of a campaign can trigger different forms, thus allowing the ability to collect through previously designed forms, information associated with the interaction between the OMniLeads agent and the person at the other end of the communication within the campaign.

It is important to explain conceptually how campaign forms are used in OMniLeads. We must clarify that in the context of a campaign when assigning ratings, it will be possible to define normal and *engaged disposition*. The latter are the ones that trigger the campaign forms.

<div>Equivocado</div>	<div>Sin acc...</div>	<div>-----</div>	<div>Ocultar</div> <div><input type="checkbox"/></div>	<div>Remover</div>
<div>Ocupado</div>	<div>Sin Acc...</div>	<div>-----</div>	<div>Ocultar</div> <div><input type="checkbox"/></div>	<div>Remover</div>
<div>No contesta</div>	<div>Sin Acc...</div>	<div>-----</div>	<div>Ocultar</div> <div><input type="checkbox"/></div>	<div>Remover</div>
<div>Buzón de voz</div>	<div>Sin Acc...</div>	<div>-----</div>	<div>Ocultar</div> <div><input type="checkbox"/></div>	<div>Remover</div>
<div>Encuesta realizada</div>	<div>Gestión</div>	<div>Calidad de Servicio</div>	<div>Ocultar</div> <div><input type="checkbox"/></div>	<div>Remover</div>
<div>Encuesta</div>	<div>Gestión</div>	<div>Encuesta cliente</div>	<div>Ocultar</div> <div><input type="checkbox"/></div>	<div>Remover</div>

Agregar campo

Figure 13: Dispositions within the campaign

In the the previous example figure, we have 2 call dispositions of type “Management”, on the one hand the call disposition “Completed Survey” that is associated with the form ” Quality of Service ” and on the other the disposition “Survey” that triggers the form “Customer survey”.

Whenever there is an active call between an OMniLeads agent and a contact at thecampaign contact database, the agent has the complementary data to the contact’s phone on their screen next to the disposition selection combo for the current contact. If the agent selects and saves a *engaged disposition* then the form associated with the engaged disposition within the campaign is triggered on the agent screen.

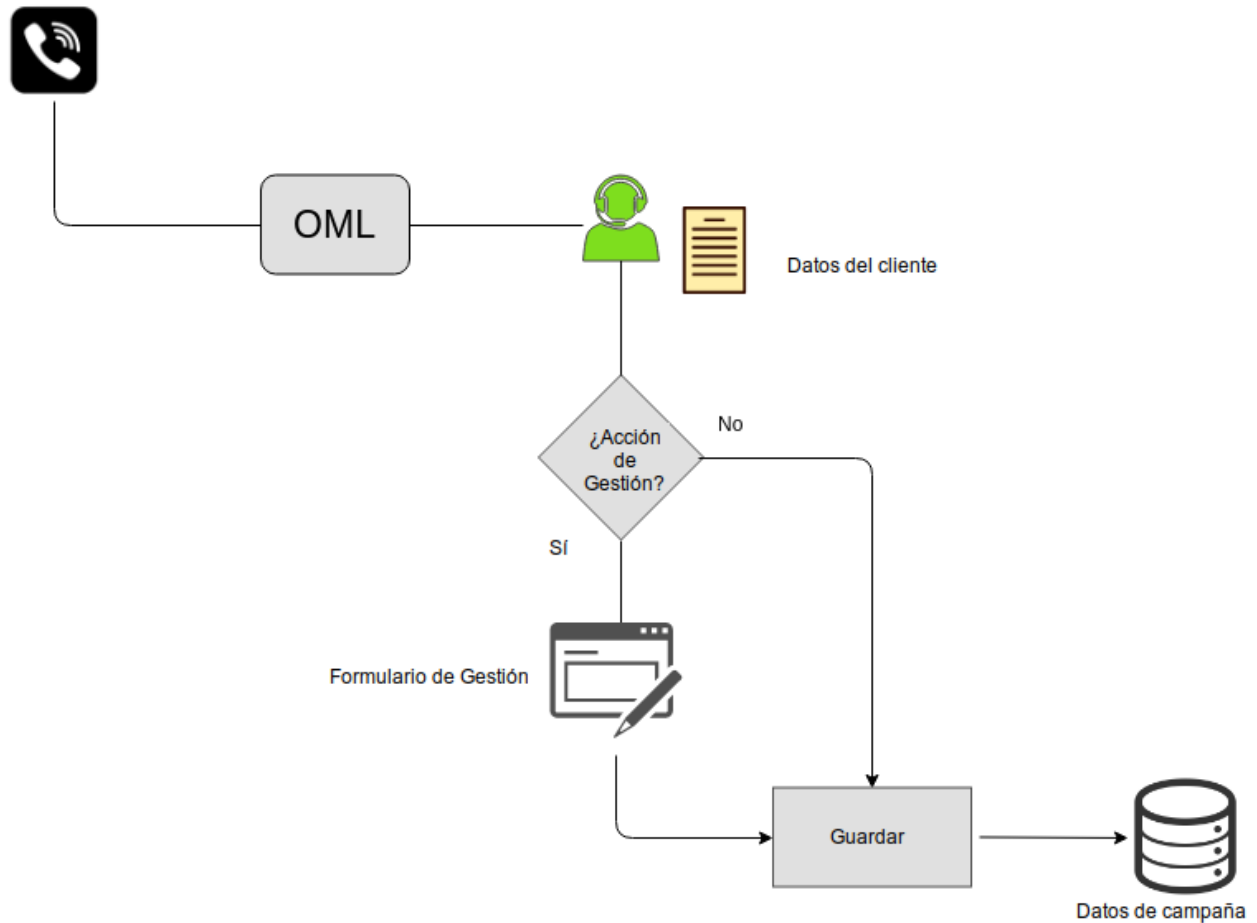


Figure 14: Management Actions and Forms

11.1.6 Manual Campaigns

Within this subsection the step-by-step example of how to manage Manual Campaigns is exemplified.

Manual Campaigns creation

To create a new manual campaign you must enter the menu item Manual Campaigns -> New Campaign.

The first stage invites us to indicate a series of campaign parameters, as indicated in Figure 1.

CAMPAÑA MANUAL: DATOS BÁSICOS

[Campaña](#)
[Opciones de calificación](#)
[Dar permiso a supervisores](#)
[Adicionar agentes](#)

Nombre:

Base de Datos de Contactos:

Los espacios, tildes, ñ-es y otros caracteres no ASCII no están permitidos.

Tipo de Interacción

☒ Formulario
 ☐ Url externa

Url Externa

Grabar llamados:

☐

Objetivo:

Sistema externo:

ID en sistema externo:

Ruta Saliente:

CID Ruta Saliente

Speech:

Paso siguiente

Figure 1: Campaign Parameters

- **Name:** Campaign name
- **Contact database:** it is used to display extra data to the phone number, when executing a call to a campaign contact.
- **Type of interaction:** configures whether the campaign will use OMniLeads's forms or launch a request towards a CRM for every connected call.
- **External URL:** in case of selecting an external URL request launch for each call, this parameter sets which of the defined CRMs shall the campaign use.
- **Enable recordings:** enable the recording of all calls that are placed through the campaign.
- **Scope:** it is defined as the amount of *engaged dispositions* expected for the campaign. In the campaign monitoring, the percentage of the campaign's arrival with respect to the defined objective is shown in real time.
- **External system:** Selects the external CRM that will execute the requests against OML (External System)
- **ID on external system:** the campaign ID in the external system from where the *click to call* or *disposition request* will be sent.
- **Outbound Routes:** An existing outbound route is assigned to a campaign.
- **CID on Outbound Routes:** This field must contain the CID assigned for an existing outbound route to a campaign.

- **Speech:** The campaign speech to be displayed on the agent console in the campaign calls”

Note: Regarding the contact base in manual (and also incoming) campaigns, it poses a flexible scenario, since it is optional to assign a contact database to this kind of campaign. In this case, the contact database is used if we want that each time an agent dials a telephone that corresponds to a contact in the database, the data (extra columns to the telephone) can be retrieved from it. In addition to working with a contact base in a manual campaign allows you to rate each contact called.

In the second screen you should assign the dispositions that are required so that agents can classify each call made. As can be seen in the following figure, in our example we handle 2 dispositions that trigger 2 different forms:

Equivocado	Sin acc...	-----	Ocultar <input type="checkbox"/>	Remover
Ocupado	Sin Acc...	-----	Ocultar <input type="checkbox"/>	Remover
No contesta	Sin Acc...	-----	Ocultar <input type="checkbox"/>	Remover
Buzón de voz	Sin Acc...	-----	Ocultar <input type="checkbox"/>	Remover
Encuesta realizada	Gestión	Calidad de Servicio	Ocultar <input type="checkbox"/>	Remover
Encuesta	Gestión	Encuesta cliente	Ocultar <input type="checkbox"/>	Remover

Agregar campo

Figure 2: Campaign Dispositions

In the following steps you can add supervisors and agents to our campaign

CAMPAÑA MANUAL: ADICIONAR AGENTES

Campaña Opciones de calificación Dar permiso a supervisores Adicionar agentes

Agregar grupo de agentes

Grupo:
 agentespordefault ▾ Agregar grupo

Agente Penalty

Perico de los Palotes 0 Remover

Fulanito De Tal 0 Remover

Agregar agente

primer paso paso anterior Finalizar

Figure 3: Agents Assignment

Agent-campaign interaction

Finally, we have our new manual campaign, so when an agent assigned to it makes a login to the platform and starts dialing calls from your webphone, the system will allow you to select the campaign on which you will assign each generated manual call from the webphone, as shown in figure 4.

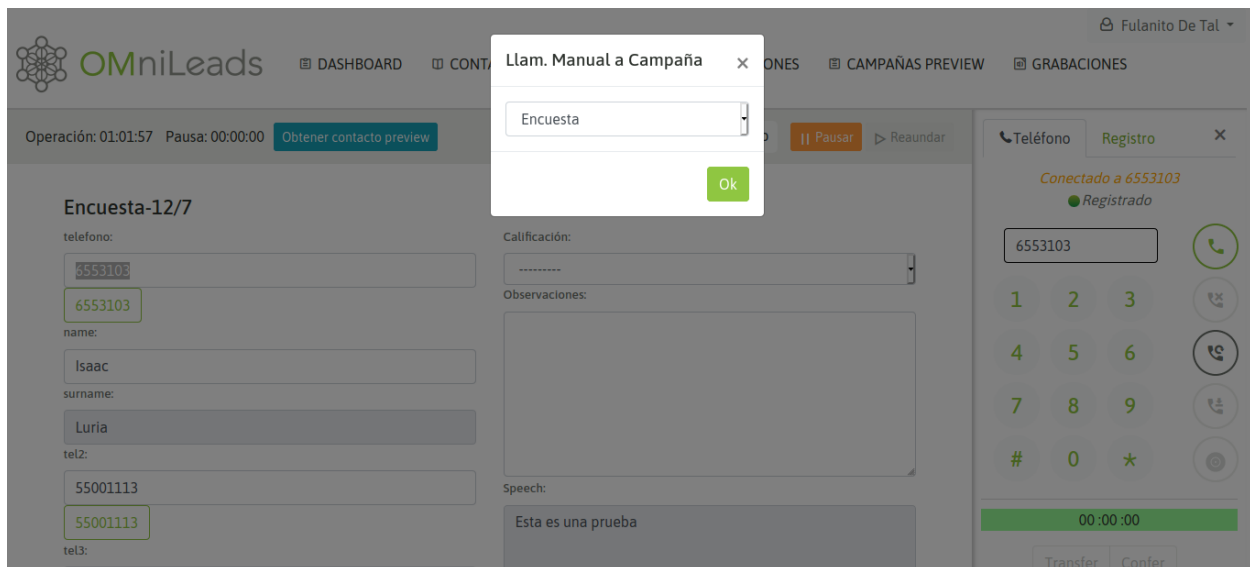


Figure 4: Manual Call to Campaign Selection

When an agent is online and dials a phone corresponding to a contact at the campaign database, the contact whose phone matches the phone dialed by the agent is displayed as a contact to select and thus display their data on the agent screen, it is also allowed to generate a new contact. Then the agent can either confirm that the call is triggered towards

the listed contact or also create a new contact and dial it. The contact information (extra phone) is displayed on the agent screen.

Operación: 01:03:15 Pausa: 00:00:00 [Obtener contacto preview](#) Conectado Pausar Reaundar

Identificar Contacto a llamar

Contactos Sugeridos

6553103 [Buscar](#)

Teléfono	name	surname	tel2	Acciones
6553103	Isaac	Luria	55001113	📞

[Es contacto nuevo](#) [Llamar sin identificar](#)

Teléfono Registro ✕

Conectado a 6553103
● Registrado

6553103 [📞](#)

1 2 3 [📞](#)
4 5 6 [📞](#)
7 8 9 [📞](#)
0 * [📞](#)

00:00:00

Figure 5: Contact Selection

If you select to call the contact listed, then the data is displayed on the screen, as shown in Figure 6.

OMniLeads DASHBOARD CONTACTOS AGENDAS CALIFICACIONES CAMPAÑAS PREVIEW GRABACIONES Fulanito De Tal

Operación: 00:02:35 Pausa: 00:02:16 [Obtener contacto preview](#) En llamado Pausar Reaundar

Encuesta

telefono: 6553103 [📞](#)

Calificación: [-----](#)

Observaciones: [📝](#)

Id_contact: 30000013

name: Isaac

surname: Luria

tel2: 55001113

Teléfono Registro ✕

Conectado a 6553103
● Registrado

6553103 [📞](#)

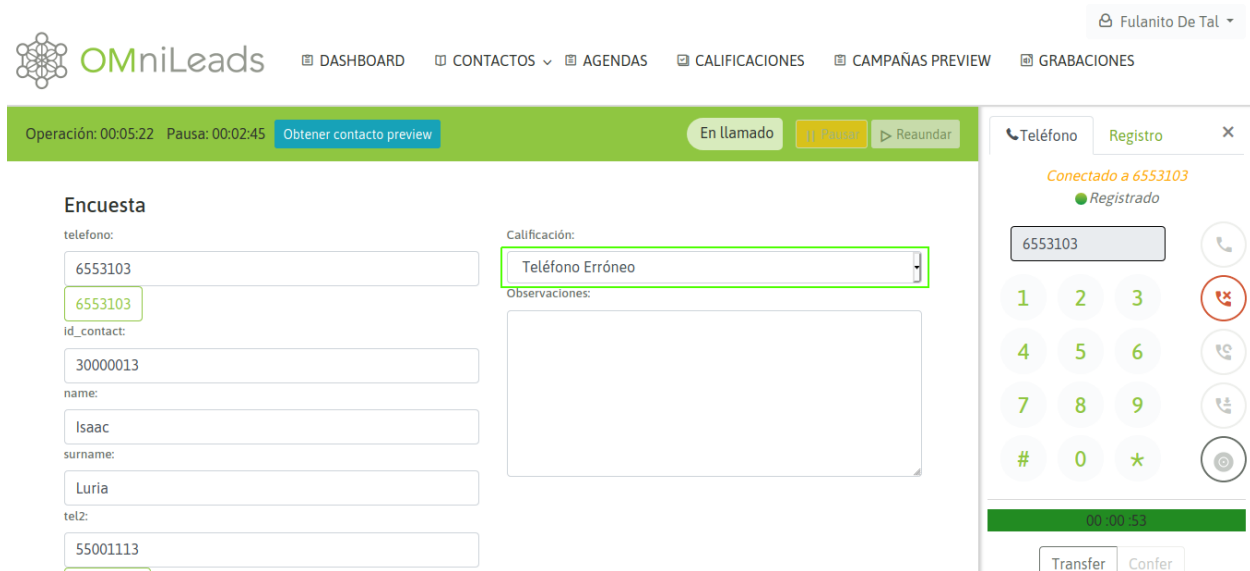
1 2 3 [📞](#)
4 5 6 [📞](#)
7 8 9 [📞](#)
0 * [📞](#)

00:00:24

[Transfer](#) [Confer](#)

Figure 6: Contact Data

In this way the agent can assign a rating on the called contact; figure 7.



OMniLeads

DASHBOARD CONTACTOS AGENDAS CALIFICACIONES CAMPAÑAS PREVIEW GRABACIONES

Operación: 00:05:22 Pausa: 00:02:45 Obtener contacto preview En llamado Pausa Reaudar

Encuesta

telefono: 6553103 6553103

id_contact: 30000013

name: Isaac

surname: Luria

telZ: 55001113

Calificación: Teléfono Erróneo

Observaciones:

Conectado a 6553103 Registrado

6553103

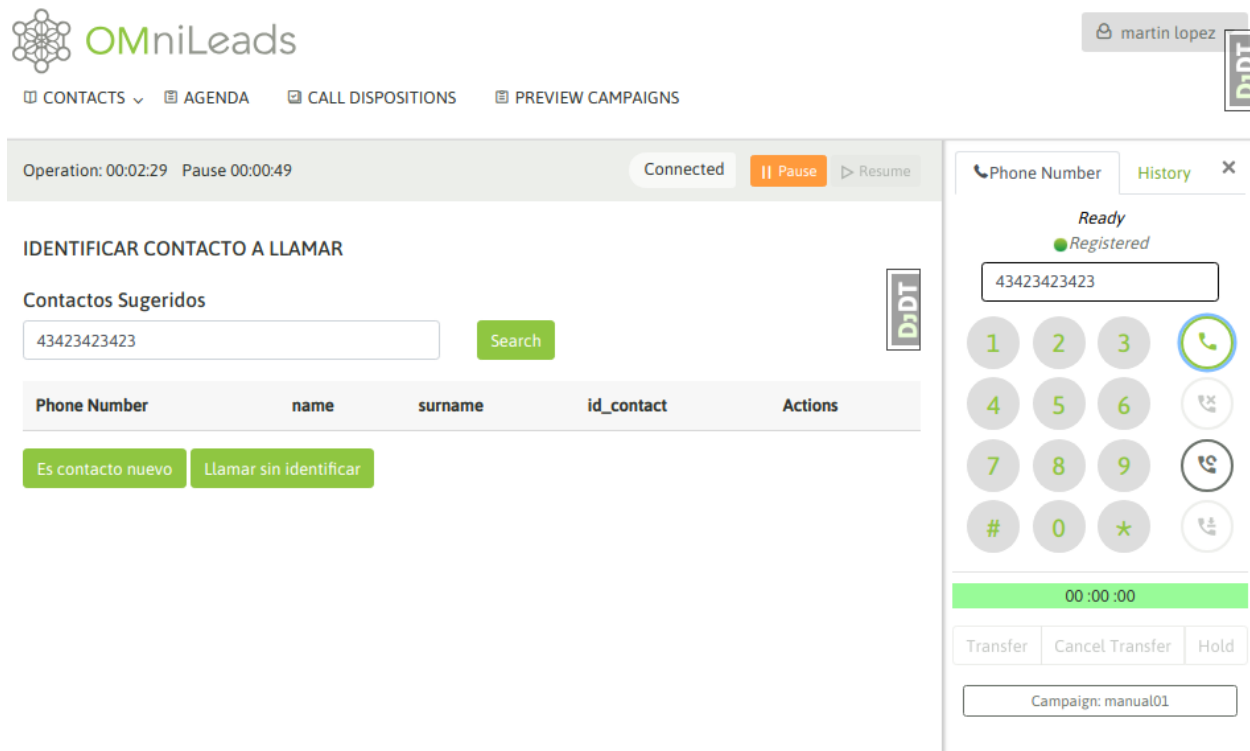
1 2 3 4 5 6 7 8 9 # 0 *

00:00:53

Transfer Confer

Figure 7: Contact Disposition

On the other hand, if the dialed telephone does not correspond to any contact of the database, then the system allows the agent to search for the contact in the database or generate a new contact. In the case of a campaign without a contact database, then each call made by an agent implies that the contact associated to the dialed call is generated (figures 5 and 6).



OMniLeads

CONTACTS AGENDA CALL DISPOSITIONS PREVIEW CAMPAIGNS

Operation: 00:02:29 Pause 00:00:49 Connected Pause Resume

IDENTIFICAR CONTACTO A LLAMAR

Contactos Sugeridos

43423423423 Search

Phone Number	name	surname	id_contact	Actions
Es contacto nuevo	Llamar sin identificar			

Ready Registered

43423423423

1 2 3 4 5 6 7 8 9 # 0 *

00:00:00

Transfer Cancel Transfer Hold

Campaign: manual01

Figure 8: Add a new contact to database of campaign 1

OMniLeads

martin lopez

CONTACTS AGENDA CALL DISPOSITIONS PREVIEW CAMPAIGNS

Operation: 00:03:22 Pause 00:00:49 Connected Pause Resume

telefono: 43423423423

name: Abraham

surname: Abulafia

id_contact: 7272727272

Guardar y llamar

Phone Number History

Ready
Registered

43423423423

1 2 3 4 5 6 7 8 9 # 0 *

00:00:00

Transfer Cancel Transfer Hold

Campaign: manual01

Figure 9: Add a new contact to database of campaign 2

Then at the time of dialing a number that does not return a contact, the agent will go through a view in which the agent must first add the contact as a record of the campaign base and then dial. Finally, the new contact and the option to classify the call with some qualification are displayed (figure 10).

OMniLeads

martin lopez

CONTACTS AGENDA CALL DISPOSITIONS PREVIEW CAMPAIGNS

Operation: 00:03:35 Pause 00:00:49 On Call Pause Resume

Completed
Contacto creado satisfactoriamente. Efectuando llamada.

MANUAL01

telefono: 43423423423

name: Abraham

surname: Abulafia

id_contact: 7272727272

Call Disposition: -----

Comments:

Phone Number History

Connected to 43423423423
Registered

43423423423

1 2 3 4 5 6 7 8 9 # 0 *

00:00:07

Transfer Cancel Transfer Hold

Campaign: manual01

Figure 10: new contact called

Multinum contact database

As we know, OMniLeads admits that each contact of a database has n telephone numbers, so that if the contact is not found in its main number (the first of our database CSV file), it can be contacted at the other phone numbers. In this case, each telephone number (which we indicate in the database load) is generated as a link within the contact data presented on the agent screen. Clicking on that link triggers a call to the extra telephone number of the contact. Figure 11 shows this scenario.

The screenshot displays the Omnileads agent interface. On the left, a contact form contains the following fields: a main number '6553103' (highlighted with a green border), a 'name' field with 'Isaac', a 'surname' field with 'Luria', an 'id_contact' field with '30000013', a 'tel2' field with '33213211' (highlighted with a green border), a 'tel3' field with '5343431' (highlighted with a green border), and a 'Call Disposition' field. A vertical 'D,DT' button is positioned between the contact form and the keypad. On the right, a numeric keypad is visible, featuring buttons for digits 1-9, 0, *, and #, along with call-related icons. Above the keypad, the status 'Agent in Pause' and 'Registered' (with a green dot) is shown. Below the keypad, a green timer displays '00:00:00'. At the bottom of the keypad area are buttons for 'Transfer', 'Cancel Transfer', and 'Hold', and a 'Campaign:' field.

Multinum contact database

Therefore, the agent can try to contact all available numbers as a *link* in the contact form, until finally selecting a disposition and moving to a new one.

11.1.7 Preview Campaigns

Within this subsection the step-by-step example of how to manage Preview Campaigns is exemplified.

Preview campaigns creation

To create a new preview campaign you must enter the menu item Preview Campaigns -> New Campaign.

The first view invites us to indicate a series of campaign parameters, as indicated in Figure 1.

Figure 1: Campaign Parameters

- **Name:** campaign name
- **Contact database:** contact database that the preview dialer will use when delivering on demand contacts to each agent.
- **Type of interaction:** sets if the campaign will work with OMniLeads forms or make a request to the CRM for each connected call.
- **External URL:** in case of selecting an external URL interaction, this field sets with CRMs must the campaign trigger.
- **Enable recordings:** Enable all campaign calls to be recorded.
- **Objective:** It is defined as the number of *positive steps* expected in the campaign. In the campaign monitoring, it is displayed in real time the percentage of campaign progress with respect to this defined objective.
- **External system:** external CRM system linked to launch *click to call actions or dispositions* on campaign contacts, through the API.
- **ID on external system:** this field must contain the ID of the campaign in the external system from where the *click to call* or *contact disposition requests* will be triggered.
- **Outbound Routes:** An existing outbound route is assigned to a campaign.
- **CID on Outbound Routes:** This field must contain the CID assigned for an existing outbound route to a campaign.
- **Disconnection time:** time that the preview dialer reserves a contact assigned to an agent, after that time the contact is released so that it can be used by another agent.

- **Speech:** The campaign speech to be displayed on the agent console in the campaign calls.

On the second screen you must assign the dispositions that are required as available to the agents when classifying each call to each contact.

CAMPAÑA PREVIEW: OPCIONES DE CALIFICACIÓN

Campaña
Opciones de calificación
Dar permiso a supervisores
Adicionar agentes
Asignación inicial de contactos a agentes

Equivocado	Sin acc...	-----	Ocultar	Remove
Ocupado	Sin acc...	-----	Ocultar	Remove
No contesta	Sin acc...	-----	Ocultar	Remove
Buzón de voz	Sin acc...	-----	Ocultar	Remove
Encuesta	Gestión	Encuesta cliente	Ocultar	Remove
Encuesta cliente	Gestión	Encuesta de Satisfacci...	Ocultar	Remove

Agregar campo

primer paso
paso anterior
Paso siguiente

Figure 2: Dispositions Call

Then it remains to assign the supervisors and agents who can work on the campaign. An assignment of agents to a campaign is exemplified in Figures 3 and 4.

CAMPAÑA PREVIEW: ADICIONAR SUPERVISORES

Campaña
Opciones de calificación
Dar permiso a supervisores
Adicionar agentes
Asignación inicial de contactos a agentes

Agregar Supervisores:

☐ Seleccionar Todos

☐ admin: ☒ mgarridoe: MigueL Garrido

primer paso
paso anterior
Paso siguiente

Figure 3: Supervisors Assignment

CAMPAÑA PREVIEW: ADICIONAR AGENTES

Campaña Opciones de calificación Dar permiso a supervisores Adicionar agentes Asignación inicial de contactos a agentes

Agregar grupo de agentes

Grupo:

agentespordefault ▾ Agregar grupo

Agente	Penalty	
Perico de los Palotes ▾	0 ▾	Remover
Fulanito De Tal ▾	0 ▾	Remover

Agregar agente

primer paso paso anterior Paso siguiente

Figure 10: Deactivation contacts field

Finally, in the final wizard step is allowed, optionally realize a pre-assignment of contacts to agents. This operation can be made using two variants

The first way is shown in the next figure:

CAMPAÑA PREVIEW: ASIGNACIÓN DE CONTACTOS

Campaña Cola Opciones de calificación Dar permiso a supervisores Adicionar agentes Asignación inicial de contactos a agentes

Proporcionalmente:

☐

primer paso paso anterior Finalizar

Figure 10: Deactivation contacts field

Contacts Assignment

If the option to allocate proportionally is selected, the system make an initial and equitable allocation of contacts to each of the agents. So when agent ask for a campaign contact, only free contacts will be given to or from the list of those assigned to it. We can see an example of distribution using the Django administrative interface in the following figure:

<input type="checkbox"/>	CONTACTO ID	AGENTE ID	CAMPANA ID	DATOS CONTACTO	ESTADO
<input type="checkbox"/>	4	2	4	{"id_contact": "30000001", "surname": "McKenna", "name": "Terence"}	INICIAL
<input type="checkbox"/>	5	2	4	{"id_contact": "30000002", "surname": "Leary", "name": "Timothy"}	INICIAL
<input type="checkbox"/>	6	2	4	{"id_contact": "30000003", "surname": "Hoffman", "name": "Albert"}	INICIAL
<input type="checkbox"/>	7	2	4	{"id_contact": "30000004", "surname": "Huxley", "name": "Aldous"}	INICIAL
<input type="checkbox"/>	8	2	4	{"id_contact": "30000005", "surname": "Hesse", "name": "Herman"}	INICIAL
<input type="checkbox"/>	9	2	4	{"id_contact": "30000006", "surname": "Borges", "name": "Jorge Luis"}	INICIAL
<input type="checkbox"/>	10	2	4	{"id_contact": "30000007", "surname": "Krishnamurti", "name": "Jiddu"}	INICIAL
<input type="checkbox"/>	11	1	4	{"id_contact": "30000008", "surname": "Hesse", "name": "Herman"}	INICIAL
<input type="checkbox"/>	12	1	4	{"id_contact": "30000009", "surname": "Crowley", "name": "Aleister"}	INICIAL
<input type="checkbox"/>	13	1	4	{"id_contact": "30000010", "surname": "Cabral", "name": "Facundo"}	INICIAL
<input type="checkbox"/>	14	1	4	{"id_contact": "30000011", "surname": "Yupanqui", "name": "Atahualpa"}	INICIAL
<input type="checkbox"/>	15	1	4	{"id_contact": "30000012", "surname": "Jung", "name": "Carl Gustav"}	INICIAL
<input type="checkbox"/>	16	1	4	{"id_contact": "30000013", "surname": "Luria", "name": "Isaac"}	INICIAL
<input type="checkbox"/>	17	1	4	{"id_contact": "30000014", "surname": "Wilson", "name": "Robert Anton"}	INICIAL
<input type="checkbox"/>	14	2	2	{"id_contact": "30000011", "surname": "Yupanqui", "name": "Atahualpa"}	INICIAL

Figure 10: Deactivation contacts field

At the moment of select the proportionally pre-assignment the system shows the random selection

CAMPAÑA PREVIEW: ASIGNACIÓN DE CONTACTOS

Campaña
Cola
Opciones de calificación
Dar permiso a supervisores
Adicionar agentes
Asignación inicial de contactos a agentes

Proporcionalmente:
☒

En orden aleatorio:
☐

primer paso

paso anterior

Finalizar

Figure 7: New option for random assignment

This option allows that the contacts could be assigned in random order to the agents, as is shown next:

<input type="checkbox"/>	CONTACTO ID	AGENTE ID	CAMPANA ID	DATOS CONTACTO	ESTADO
<input type="checkbox"/>	4	2	4	{"id_contact": "30000001", "surname": "McKenna", "name": "Terence"}	INICIAL
<input type="checkbox"/>	5	2	4	{"id_contact": "30000002", "surname": "Leary", "name": "Timothy"}	INICIAL
<input type="checkbox"/>	6	2	4	{"id_contact": "30000003", "surname": "Hoffman", "name": "Albert"}	INICIAL
<input type="checkbox"/>	7	2	4	{"id_contact": "30000004", "surname": "Huxley", "name": "Aldous"}	INICIAL
<input type="checkbox"/>	8	2	4	{"id_contact": "30000005", "surname": "Hesse", "name": "Herman"}	INICIAL
<input type="checkbox"/>	9	2	4	{"id_contact": "30000006", "surname": "Borges", "name": "Jorge Luis"}	INICIAL
<input type="checkbox"/>	10	2	4	{"id_contact": "30000007", "surname": "Krishnamurti", "name": "Jiddu"}	INICIAL
<input type="checkbox"/>	11	1	4	{"id_contact": "30000008", "surname": "Hesse", "name": "Herman"}	INICIAL
<input type="checkbox"/>	12	1	4	{"id_contact": "30000009", "surname": "Crowley", "name": "Aleister"}	INICIAL
<input type="checkbox"/>	13	1	4	{"id_contact": "30000010", "surname": "Cabral", "name": "Facundo"}	INICIAL
<input type="checkbox"/>	14	1	4	{"id_contact": "30000011", "surname": "Yupanqui", "name": "Atahualpa"}	INICIAL
<input type="checkbox"/>	15	1	4	{"id_contact": "30000012", "surname": "Jung", "name": "Carl Gustav"}	INICIAL
<input type="checkbox"/>	16	1	4	{"id_contact": "30000013", "surname": "Luria", "name": "Isaac"}	INICIAL
<input type="checkbox"/>	17	1	4	{"id_contact": "30000014", "surname": "Wilson", "name": "Robert Anton"}	INICIAL
<input type="checkbox"/>	14	2	2	{"id_contact": "30000011", "surname": "Yupanqui", "name": "Atahualpa"}	INICIAL

Figure 8: Proportional and Random Distribution of Contacts

Finally our campaign is available to start operating. Therefore, when the agents assigned to it perform a login to the platform, they should have the preview campaign as shown in Figure 5.

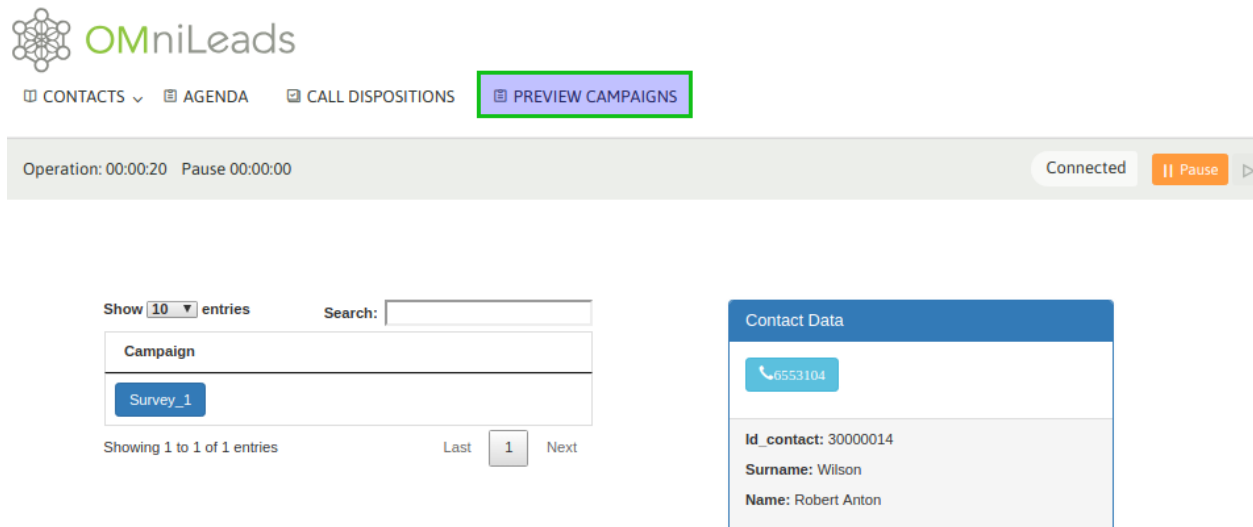


Figure 9: Preview Campaign Agent View

Agent-campaign interaction

If the agent clicks on the phone, then the call starts, and the data (extras to the phone) of the contact are displayed in the agent view, allowing the agent in turn to disposition the call with any of the call dispositions assigned to the campaign:

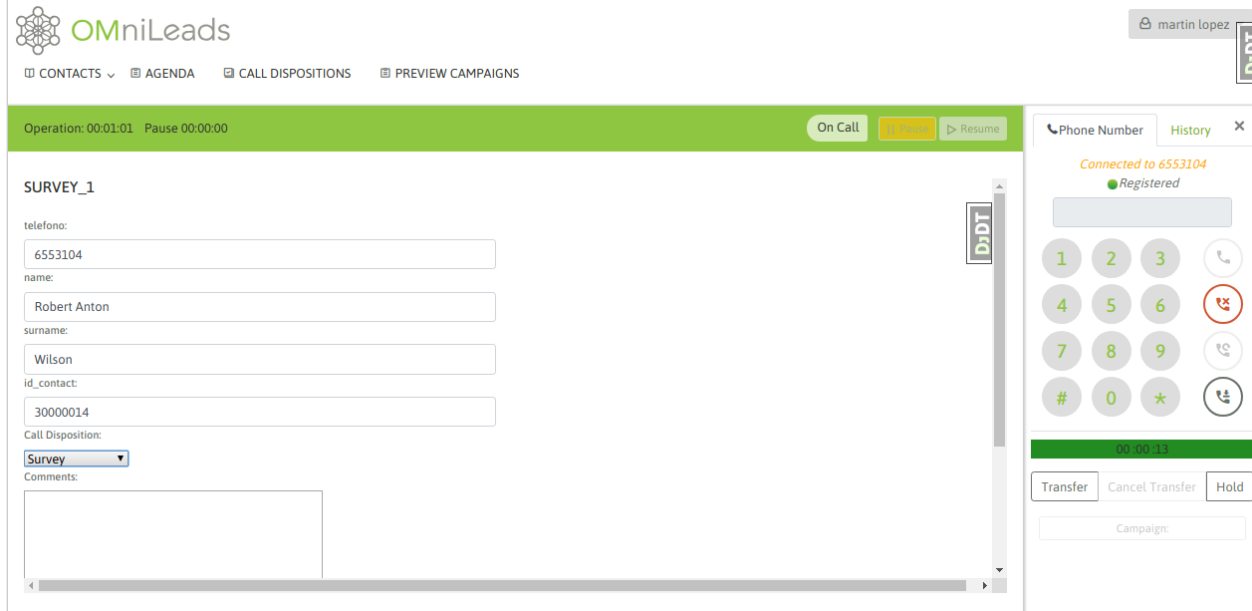
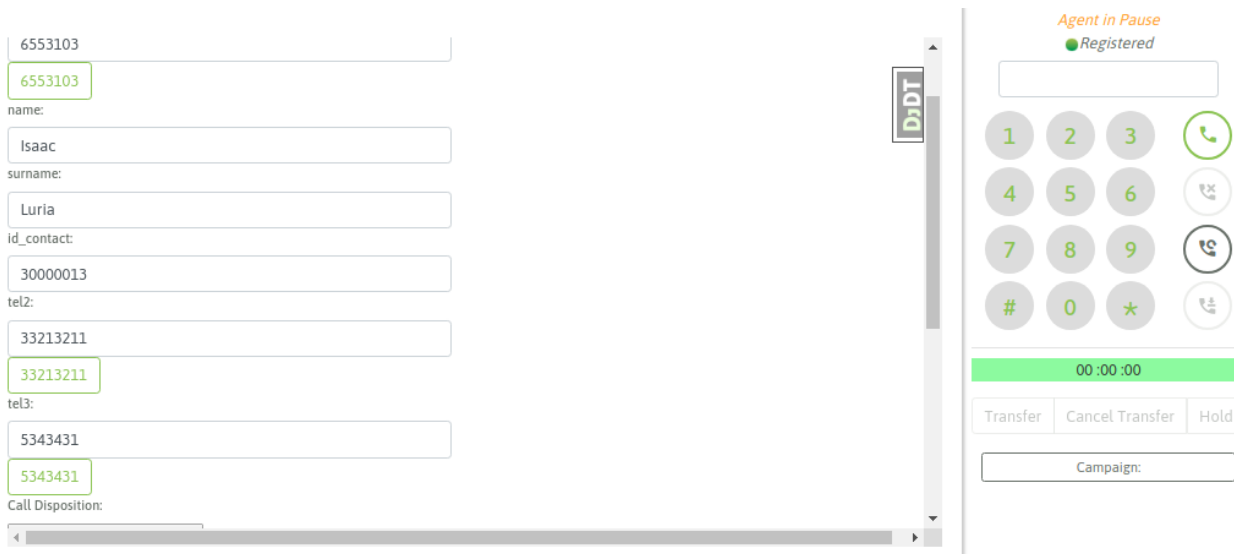


Figure 10: Deactivation contacts field

Campaign with Multinum database

As we know, OMniLeads admits that each contact of a database has n telephone numbers, so that if the contact is not found in its main number (the first of our database CSV file), it can be contacted at the other phone numbers. In this case, each telephone number (which we indicate in the database load) is generated as a link within the contact data presented on the agent screen. Clicking on that link triggers a call to the extra telephone number of the contact. Figure 11 shows this scenario.



Campaign with Multinum database

Therefore, the agent can try to contact all available numbers as a *link* in the contact form, until finally selecting a disposition and moving to a new one.

Management of contacts assignments

Once the campaign is created the available contacts for every agent will be assign following an order established on the model `AgenteEnContacto`, this feature allows edit that order using the exportation/importation of .csv files

The idea is that the administrator can download the current order of contacts to a .csv file, reorder the rows and then import the file, in which the new order impacts on the order of the assigned contacts to agents in the campaign.

This feature is accessible using the preview campaign menu

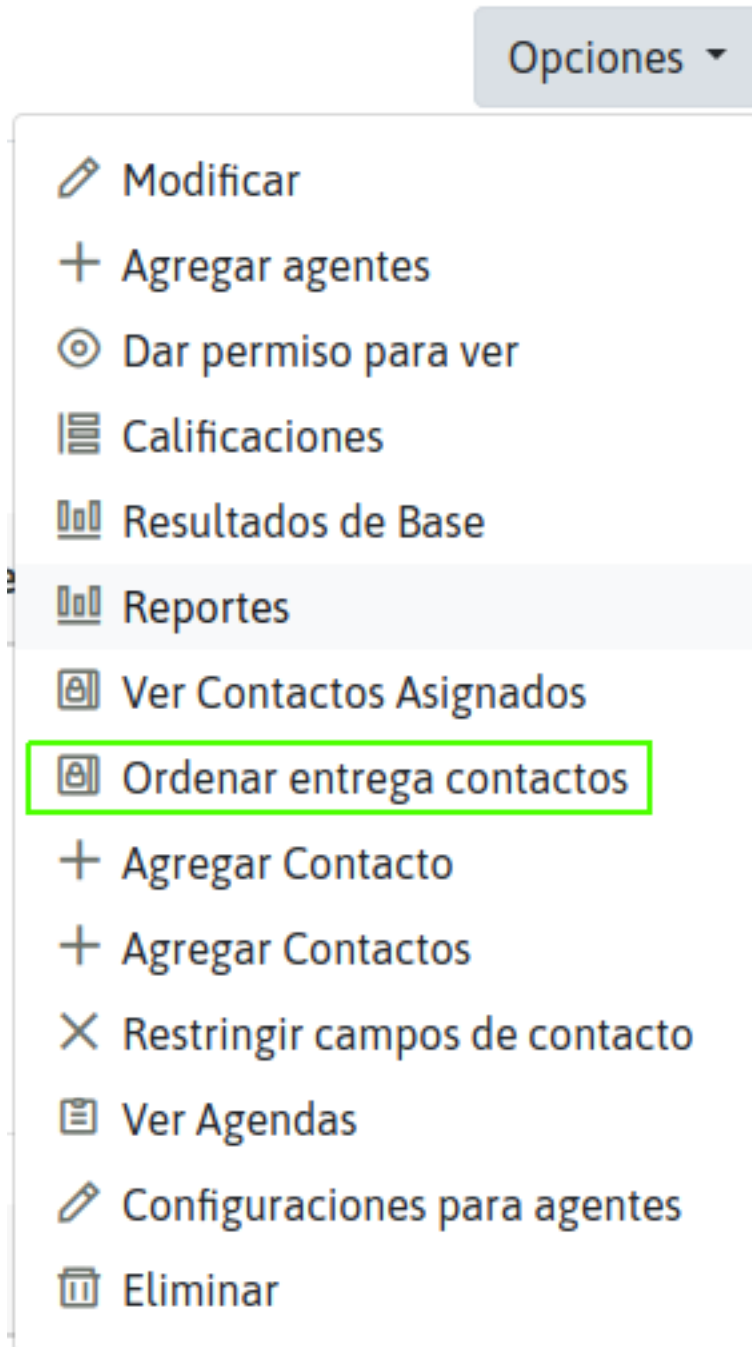


Figure 8: Access to reorder contacts assignments page

GESTIONAR CONTACTOS DE CAMPAÑA: ENCUESTA-PREVIEW-14/7

Campo desactivación:

activo

Exportar status actual contactos

Exportar y desactivar

Importar status contactos

Seleccionar archivo .csv:

Examinar... No se ha seleccionado ningún archivo.

Importar y desactivar

Figure 9: Reorder contacts assignments page

Also, is possible to tag contacts as disabled, which will not delivered to any agent

This is posible by defining a deactivation field for the campaign choosing between the data colums in contacts database

Campo desactivación:

activo

Figure 10: Deactivation contacts field

Important: The field to choose should be taken into consideration in the CSV format before importing the contact database and attaching it to the campaign. It is not allowed to add or modify the names of the columns in the CSV afterwards.

After make current contacts order exportation is possible to edit the deactivation field column with values 0 or FALSE wich, after importation of.csv file will indicates to the system to not deliver this contacts to any agent.Any other value different to 0 or FALSE makes the system can deliver the contact.

11.1.8 Predictive dialer campaigns

Within this subsection the step-by-step example of how to manage Predictive dialer Campaigns is exemplified.

Presentation

OMniLeads makes campaigns with automatic call dialing available through a predictive dialer.

Important: The automatic dialer functionality is not provided within the GPLv3 base components, to use this type of campaigns the integration with [Wombat Dialer](#). is contemplated. The use of this software is subject to the acquisition of the corresponding license with the software manufacturer.

Having clarified the theme of the engine dialer component, we proceed with the explanation of the necessary steps when generating a campaign with predictive dialing.

Dialer campaign creation

Enter the **Campaigns menu -> Dialer Campaigns -> New Campaigns** where a sequence of configuration stages are displayed.

The first stage looks like figure 1.

CAMPAÑA DIALER: DATOS BÁSICOS - CANALES EN USO: 0

Nombre:

Mostrar nombre al recibir llamado: ☒

Base de Datos de Contactos:

Los espacios, tildes, ñ-es y otros caracteres no ASCII no están permitidos.

Fecha inicio: Fecha fin:

Sistema externo: ID en sistema externo:

Tipo de Interacción: ☒ Formulario ☐ Url externa

Url Externa:

Ruta Saliente:

CID Ruta Saliente:

Objetivo:

Speech:

Paso siguiente

Figure 1: Campaign Parameters

- **Name:** Campaign Name
- **Show campaign name whe reciving call:** it can be enabled so that each call connected by the dialer to the agent implies notification of the name of the campaign associated with that call.
- **Contact database:** the contact database that the dialer uses to make contacts and generate calls.
- **Initial date:** date on which, while the campaign is active and with agents connected, it will start dialing.
- **End date:** date on which the dial will stop dialing no matter how many numbers remain unmarked.
- **External system:** external CRM system linked to launch *click to call actions or dispositions* on campaign contacts, through the API.

- **ID on external system:** ID of the campaign in the external system from where the *click to call* or *disposition requests* will be sent.
- **Interaction class:** select if the campaign will use a *campaign form* or will launch an http-request to an external *CRM system*.
- **External URL:** URL (http-request) to be triggered each time the dialer connects a call to an agent.
- **Outbound Routes:** An existing outbound route is assigned to an campaign.
- **CID on Outbound Routes:** This field must contain the CID assigned for an existing outbound route to a campaign.
- **Scope:** it is defined as the amount of *engaged dispositions* expected for the campaign. In the campaign monitoring, the percentage of the campaign's arrival with respect to the defined objective is shown in real time.
- **Speech:** Campaign speech to be displayed on the agent console on campaign calls.

Once these fields are completed, you must click on the *Next* button, to continue with the campaign configuration.

The next stage establishes more parameters that model the campaign behavior, as shown in Figure 2.

CAMPAÑA DIALER: DATOS DE COLA

[Campaña](#)
[Cola](#)
[Opciones de calificación](#)
[Actuacion](#)
[Reglas Incidencias](#)
[Dar permiso a supervisores](#)
[Adicionar agentes](#)

Sincronizar

Cantidad Max de llamadas:

Nivel de Servicio:

Importancia de de campaña:

Grabar llamados:

☒

Audio para contestadores:

Factor de boost inicial:

Anuncio de entrada para el agente:

Destino en caso de timeout

Tipo destino:

Destino:

primer paso

paso anterior

Paso siguiente

Tiempo de descanso entre llamadas:

En segundos

Estrategia de distribución:

Tiempo de espera en cola:

En segundos

Detectar contestadores:

☒

Activar predictividad:

☒

Dial timeout:

Es recomendable que este valor sea menor al dial timeout definido en la ruta saliente. En segundos

Musica de espera:

** Figure 2: Campaign Parameters

- **Max amount of calls:** number of calls that are allowed to be queued while waiting for the availability of an agent. Above that number, the call will be sent to the *failover destination*.
- **Wrapup time:** It is the rest time (in seconds) that each agent has between each call connected by the dialer.
- **Service level:** It is a parameter to measure how many of the calls were connected to an agent within that time slot (in seconds).
- **Distribution strategy:** call distribution method that the campaign on agents will use. For outbound campaigns, RRmemory is recommended.
- **Campaign priority:** it is a linear parameter (1 to 10), which implies how important are the calls of this campaign

with respect to others. Set priorities for agents working in several campaigns simultaneously. If the value is left at 0 (by default), equity is maintained with the rest of the campaigns.

- **Waiting time:** It is the time (in seconds), that the call contacted will be queued, waiting for an agent to be released to connect it.
- **Enable recordings:** Enable all campaign calls to be recorded.
- **Answer Machine Detection:** enable answering machine detection.
- **AMD audio play:** You can indicate the playback of an audio if an answering machine is detected. For audio to be available, it must be pre-loaded from the Audios menu> New audio
- **Predictive mode activation:** The dialer offers a configuration that makes it possible to review statistics on the effectiveness of the campaign during its performance. Depending on these results, the number of calls generated by available agent will be varied in such a way as to avoid dead times between each calls assigned by the dialer to each agent.
- **Initial Boost Factor:** Indicates the value by which you want to multiply the predictive behavior. For example: if the dialer detected that it can make three calls simultaneously because it is the result of the successful communications statistics, by placing 2 in the initial boost factor, the dialer is asked to double that value and then make six calls at once.
- **Dial timeout:** It is the time (in seconds) to ring before sending a CANCEL.
- **Agent Announcement:** An audio can be selected for playing at the moment a campaign call comes to an agent. For the audio to be available, it should be previously uploaded from the menu *Telephony -> Audios -> Custom Audios*, button “Add”.
- **Music on hold:** You can select a special music on hold for the campaign, which will be played until there is an agent available to answer the call, or until the waiting time in queue is reached so the call is directed to the destination in case of timeout. For music on hold to be available, it should be previously added in the menu *Telephony -> Audios -> Waiting Music lists* ,”Add” button.
- **Failover dst:** destination to where calls that have expired (timeout) go.

After completing all the fields, click *next*

On the this stage you must assign the dispositions that are required as available to the agents when classifying each call to each contact.

CAMPAÑA DIALER: OPCIONES DE CALIFICACIÓN

Campaña Cola Opciones de calificación Actuación Reglas Incidencias Dar permiso a supervisores Adicionar agentes

Sincronizar

Buzón de voz	Sin acc...	-----	Ocultar <input type="checkbox"/>	Remover
Equivocado	Sin Acc...	-----	Ocultar <input type="checkbox"/>	Remover
Encuesta	Gestión	Encuesta cliente	Ocultar <input type="checkbox"/>	Remover
Encuesta cliente	Gestión	Encuesta de Satisfacci...	Ocultar <input type="checkbox"/>	Remover

Agregar campo

primer paso paso anterior Paso siguiente

Figure 3: Campaign Parameters - Call Dispositions

We arrive at the configuration that determines what days of the week and the time of day which this campaign can place calls (always within the range of dates established in the first step of creating the campaign).

CAMPAÑA DIALER: ACTUACIÓN VIGENTE

Campaña Cola Opciones de calificación Actuación Reglas Incidencias Dar permiso a supervisores Adicionar agentes

Sincronizar

Domingo: ☐

Lunes: ☒

Martes: ☒

Miércoles: ☒

Jueves: ☒

Viernes: ☒

Sábado: ☐

Hora desde: 09:00

Hora hasta: 20:00

primer paso paso anterior Paso siguiente

Figure 4: Campaign Parameters - Days and Hours

Now the reschedule rules are configured, that is how the dialer tries to call numbers whose contact attempt was failed: Busy, Answering Machine, No Answer, temporarily out of coverage, etc.

CAMPAÑA DIALER: REGLAS INCIDENCIA

[Campaña](#) [Cola](#) [Opciones de calificación](#) [Actuacion](#) [Reglas Incidencias](#) [Dar permiso a supervisores](#) [Adicionar agentes](#)

[Sincronizar](#)

Estado	Reintentar más tarde(En segundos)	Intento max.(En cantidades)	
<div>Ocupado</div>	<div>120</div>	<div>2</div>	<div>Remover</div>
<div>No atendido</div>	<div>600</div>	<div>2</div>	<div>Remover</div>
<div>Contestador</div>	<div>1200</div>	<div>2</div>	<div>Remover</div>

Agregar regla

primer paso

paso anterior

Paso siguiente

Figure 5: Campaigns Parameters - Incidence Rules

The fields completed allow you to determine how many times it should retry to establish communication and how many seconds after an failed try, the dialer should retry to call.

Callstatus that can be retried automatically dial are:

- Busy
- Answer machine
- Phone not answer
- Call rejected (Rejected): when the call could not be carried out due to problems inherent to the external telephone network.
- Timeout: when the call was contacted, it was connected but no agent was free to handle it.

Finally we go to the last stage

CAMPAÑA DIALER: ADICIONAR SUPERVISORES

[Campaña](#) [Datos básicos de la cola](#) [Opciones de calificación](#) [Dar permiso a supervisores](#) [Adicionar agentes](#) [Sincronizar](#)

Agregar Supervisores:

☐ Seleccionar Todos

☐ admin: ☒ mgarrioe: Miguel Garrido

[primer paso](#) [paso anterior](#) [Paso siguiente](#)

Figure 6: Campaign Parameters - Supervisors

Finally we go to the last stage

CAMPAÑA DIALER: ADICIONAR AGENTES

[Campaña](#) [Datos básicos de la cola](#) [Opciones de calificación](#) [Dar permiso a supervisores](#) [Adicionar agentes](#) [Sincronizar](#)

Agregar grupo de agentes

Grupo:

agentespordefault

Agregar grupo

Agente	Penalty	
<div>Perico de los Palotes</div>	<div>0</div>	<div>Remover</div>
<div>Fulanito De Tal</div>	<div>0</div>	<div>Remover</div>

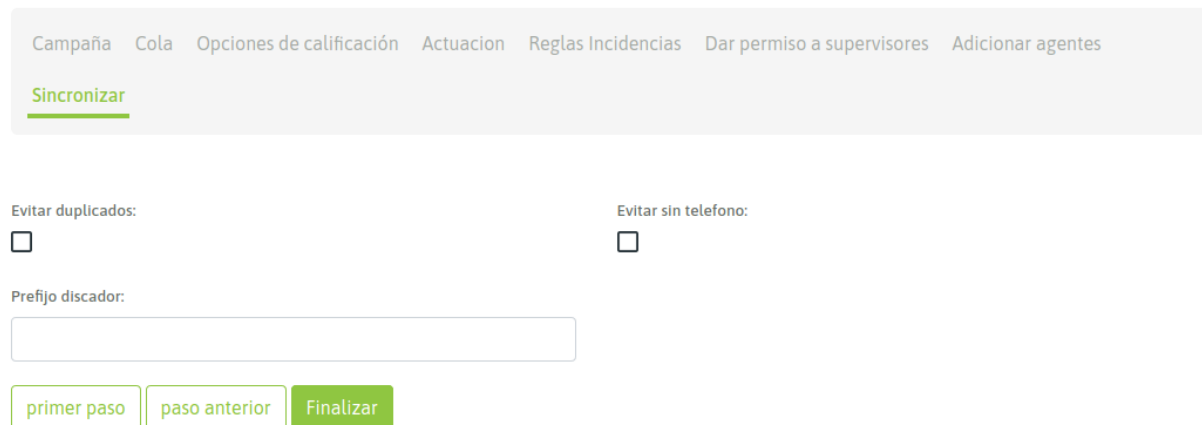
Agregar agente

[primer paso](#) [paso anterior](#) [Paso siguiente](#)

Figure 7: Campaign Parameters - Agents

Finally we go to the last stage

CAMPAÑA DIALER: SINCRONIZAR



Campaña Cola Opciones de calificación Actuacion Reglas Incidencias Dar permiso a supervisores Adicionar agentes

Sincronizar

Evitar duplicados: ☐

Evitar sin telefono: ☐

Prefijo discador:

primer paso paso anterior Finalizar

Figure 8: Campaign Parameters - Sync

In this step, three options are simply indicated:

- **Avoid Duplicated:** Select this option to avoid uploading records with the duplicate main phone to the dialer.
- **Avoid Without Numbers:** select this option to avoid uploading contact base records that do not have a primary telephone to the dialer.
- **Prefix:** This field is used to indicate to the dialer the prefix that must be placed before each number of the contact base at the time of dialing the campaign calls.

Finally we go to the last stage

Campaign Activation

The newly created campaign appears in the inactive state (figure 7), in the list of predictive campaigns.

Id	Nombre	Fecha Ini	Fecha Fin	Base de datos	Estado	Acciones
----	--------	-----------	-----------	---------------	--------	----------

No existen campañas

Campañas Pausadas

Id	Nombre	Fecha Ini	Fecha Fin	Base de datos	Estado	Acciones
----	--------	-----------	-----------	---------------	--------	----------

No existen campañas

Campañas Inactivas

Id	Nombre	Fecha Ini	Fecha Fin	Base de datos	Estado	Acciones
16	Dialer-14/7 	14/07/2021	13/08/2021	oml-example-multinum-db-activo-14/7: (47 contactos) Cambiar	Inactiva	<button>Opciones ▾</button>

Campañas Finalizadas

Id	Nombre	Fecha Ini	Fecha Fin	Base de datos	Estado	Acciones
13	pruebadiscador 	08/06/2021	08/06/2021	oml-example-db: (14 contactos) Cambiar	Finalizada	<button>Opciones ▾</button>

Figure 9: Inactive Campaign

The administrator should activate the campaign manually.

Campañas Inactivas

Id	Nombre	Fecha Ini	Fecha Fin	Base de datos	Estado	Acciones
16	Dialer-14/7 	14/07/2021	13/08/2021	oml-example-multinum-db-activo-14/7: (47 contactos) Cambiar	Inactiva	<button>Opciones ▾</button>

Campañas Finalizadas

Id	Nombre	Fecha Ini	Fecha Fin	Base de datos	Estado	Acciones
13	pruebadiscador 	08/06/2021	08/06/2021	oml-example-db: (14 contactos) Cambiar		
9	venta 	22/10/2020	31/10/2020	5-7-reciclada: (31 conta) Cambiar		
12	CAMPANA_CLONADA_12_(reciclada) 	23/12/2020	23/12/2020	9-10-reciclada: (15 cont) Cambiar		
10	pruebaAMD 	23/12/2020	10/01/2021	oml-example-multinum-db-	Finalizada	<button>Opciones ▾</button>

- [▶ Iniciar](#)
- [✎ Editar](#)
- [+ Agregar agentes](#)
- [✕ Finalizar](#)
- [👁 Dar permisos para ver](#)
- [📋 Calificaciones](#)
- [📊 Resultados de Base](#)
- [📊 Reportes](#)
- [🔄 Reglas de incidencia](#)
- [✎ Configuraciones para agentes](#)
- [🗑 Eliminar](#)

Figure 10: Activate Campaign

After activating the campaign, it should appear in active campaign list (figure 9)

Finalizado ×
 Operación Exitosa! Se llevó a cabo con éxito la activación de la Campaña.

CAMPAÑAS DIALER

Campañas Activas - **Canales en uso: 10**

Finalizar campañas sin contactos por discar

Id	Nombre	Fecha Ini	Fecha Fin	Base de datos	Estado	Acciones
16	Dialer-14/7 🔍	14/07/2021	13/08/2021	oml-example-multinum-db-activo-14/7: (47 contactos)	Activa	Opciones ▾

Figure 11: Active Campaigns

As soon as an agent assigned to our predictive campaign login within the active date and time range of the campaign, then the dialer can start generating calls and deliver these to the agents active in the campaign.

Predictive dialer campaigns completion

To determine when a predictive campaign is without records to be dial, the status of the campaign must be checked by clicking on the name of the campaign (figure 10).

Finalizado ×
 Operación Exitosa! Se llevó a cabo con éxito la activación de la Campaña.

CAMPAÑAS DIALER

Campañas Activas - **Canales en uso: 10**

Finalizar campañas sin contactos por discar

Id	Nombre	Fecha Ini	Fecha Fin	Base de datos	Estado	Acciones
16	Dialer-14/7 🔍	14/07/2021	13/08/2021	oml-example-multinum-db-activo-14/7: (47 contactos)	Activa	Opciones ▾

Dialer-14/7 ×

Nombre: Dialer-14/7

Estado: **Activa**

Llamadas efectuadas: 48

Llamadas Terminadas: 48

Llamadas restantes estimadas: 0

Reintentos abiertos: 0

Agente no disponible: 48

Cerrar

Figure 12: Campaign Pending Calls

When the value “Pending calls” is at 0, you should press the button “End campaigns without contacts by dialing” to end all campaigns without contacts to dial, as shown in next figure:

Campañas Activas - Canales en uso: 1

Finalizar campañas sin contactos por discar

Id	Nombre	Fecha Ini	Fecha Fin	Base de datos	Estado	Acciones
16	Dialer-14/7 	14/07/2021	13/08/2021	12-13-reciclada: (48 contactos)	Activa	Opciones ▾

Figure 13: End campaigns without contacts to dial

Or it can be finalized individually:



español (es)

CAMPAÑAS DIALER

Campañas Activas - Canales en uso: 1

Id	Nombre	Fecha Ini	Fecha Fin	Base de datos
16	Dialer-14/7 	14/07/2021	13/08/2021	12-13-reciclada: (48 cont

Campañas Pausadas

Id	Nombre	Fecha Ini	Fecha Fin	Base de datos
No existen campañas				

Salir

 Editar
 Agregar agentes
 Pausar
 Reciclar
 Finalizar
 Dar permiso para ver
 Calificaciones
 Resultados de Base
 Reportes
 Agregar Contacto
 Agregar Contactos
 Restringir campos de contacto
 Ver Agendas
 Reglas de incidencia
 Configuraciones para agentes
 Eliminar

Figure 14: Finalize a Campaign

The campaign goes to the list of * Campaigns finished*.

Recycling and rotation of contact databases

When a predictive campaign runs out of pending records to call its contact base, that campaign can be reused through two possibilities:

Recycle the contact base

This option allows the administrator to select contacts from the base with certain dispositions made by agents (on connected calls) as well as dispositions made by the dialer (on failed calls; busy, does not answer, voicemail, etc.), as a recycling criterion from the contact base of the current campaign, so that the dialer returns to call the contacts that fall within the dispositions indicated in the recycling.

To recycle a completed campaign, we should display the menu “Options” of the campaign, and select “Recycle”, as it is shown in the following figure:

Campañas Finalizadas

Id	Nombre	Fecha Ini	Fecha Fin	Base de datos	Estado	Acciones
13	pruebadiscador 🔍	08/06/2021	08/06/2021	oml-example-db: (14 contactos) Cambiar	Finalizada	Opciones ▾
9	venta 🔍	22/10/2020	31/10/2020	5-7-reciclada: (31 contactos) Cambiar	Finalizada	Opciones ▾
12	CAMPANA_CLONADA_12_(reciclada) 🔍	23/12/2020	23/12/2020	9-10-reciclada: (15 contactos) Cambiar	<div> ▶ Iniciar 🔄 Reciclar 👁 Dar permisos para ver 📋 Calificaciones 📊 Resultados de Base 📄 Reportes 🗑 Eliminar </div>	Opciones ▾
10	pruebaAMD 🔍	23/12/2020	10/01/2021	oml-example-multinum-db-activo: (48 contactos) Cambiar		
11	2pruebaAMD 🔍	23/12/2020	23/12/2020	9-11-reciclada: (16 contactos) Cambiar		
16	Dialer-14/7 🔍	14/07/2021	13/08/2021	12-13-reciclada: (48 contactos) Cambiar		

Figure 15: Campaign Recycle

The different call dispositions will be shown and we can choose between 2 recycling modes:

- Redial the selected contacts from the same campaign.
- In other words, redial the selected contacts, but on a new campaign similar to the original one and whose contact base will be the result of recycling.

To complete the procedure you must select the dispositions to be called again and then go through the stages of predictive campaign configuration in case you need to adjust any configuration parameter of the recycled campaign.

Nombre	Cantidad	Seleccionar
Buzón de voz	1	<input type="checkbox"/>
Encuesta	3	<input type="checkbox"/>
Encuesta cliente	1	<input type="checkbox"/>
Equivocado	1	<input type="checkbox"/>
Contestador	32	<input type="checkbox"/>
Ocupado	4	<input type="checkbox"/>
Cancelado	3	<input type="checkbox"/>
Congestion	3	<input type="checkbox"/>

Como desea reciclar esta campana

- ☐ Reciclar y crear una nueva campaña
- ☐ Reciclar y utilizar la campaña actual

Reciclar

Figure 16: Recycle, Call Dispositions

CAMPAÑA DIALER: DATOS BÁSICOS - **CANALES EN USO: 0**

Campaña Cola Opciones de calificación Actuación Reglas Incidencias Dar permiso a supervisores Adicionar agentes Sincronizar

Nombre: Los espacios, tildes, ñ-es y otros caracteres no ASCII no están permitidos.

Mostrar nombre al recibir llamado: ☒

Base de Datos de Contactos:

Fecha inicio: Fecha fin: Sistema externo: ID en sistema externo:

Tipo de Interacción: ☒ Formulario ☐ Url externa

Url Externa: Ruta Saliente: CID Ruta Saliente: Objetivo:

Speech:

[Paso siguiente](#)

Figure 17: Parametrization of Recycled Campaign

When executing the recycling, the campaign is in the *Inactive* state, therefore it remains to activate the campaign so that the recycled contacts begin to be marked by the dialer.

Campañas Inactivas

Id	Nombre	Fecha Ini	Fecha Fin	Base de datos	Estado	Acciones
17	CAMPANA_CLONADA_17_(reciclada)	14/07/2021	13/08/2021	13-14-reciclada: (32 contactos) Cambiar	Inactiva	Opciones

Campañas Finalizadas

Id	Nombre	Fecha Ini	Fecha Fin	Base de datos
13	pruebadiscador	08/06/2021	08/06/2021	oml-example-db: (14 contactos) Cambiar
9	venta	22/10/2020	31/10/2020	5-7-reciclada: (31 contactos) Cambiar
12	CAMPANA_CLONADA_12_(reciclada)	23/12/2020	23/12/2020	9-10-reciclada: (15 contactos) Cambiar

- [▶ Iniciar](#)
- [✎ Editar](#)
- [+ Agregar agentes](#)
- [✕ Finalizar](#)
- [👁 Dar permisos para ver](#)
- [📋 Calificaciones](#)
- [📊 Resultados de Base](#)
- [📊 Reportes](#)
- [🔄 Reglas de incidencia](#)
- [✎ Configuraciones para agentes](#)
- [🗑 Eliminar](#)

Campaign activation

Contact database rotation

A campaign can replace its contact database with a new one. This allows to continue operating with the same campaign but to renew the source of contacts to call. In this way the history of reports, recordings and other statistics in the same campaign is followed.

To carry out a contact database change, the campaign must be paused or under the status of *finished*. Then you must


click on the *database change* campaign option.

Campañas Pausadas

Id	Nombre	Fecha Ini	Fecha Fin	Base de datos	Estado	Acciones
----	--------	-----------	-----------	---------------	--------	----------

No existen campañas

Campañas Inactivas

Id	Nombre	Fecha Ini	Fecha Fin	Base de datos	Estado	Acciones
17	CAMPANA_CLONADA_17_(reciclada) 	14/07/2021	13/08/2021	13-14-reciclada: (32 contactos) Cambiar	Inactiva	Opciones ▾

Campañas Finalizadas


Id	Nombre	Fecha Ini	Fecha Fin	Base de datos	Estado	Acciones
13	pruebadiscador 	08/06/2021	08/06/2021	oml-example-db: (14 contactos) Cambiar	Finalizada	Opciones ▾

Figure 19: Contact Database Change

This will display a screen similar to the one shown in the following figure 17.

Base de Datos de Contactos:

▾

Evitar duplicados:

☐

Evitar sin telefono:

☐

Prefijo discador:

Telefonos:

- ☐ name
- ☐ surname
- ☐ tel2
- ☐ tel3
- ☐ tel4
- ☐ activo

Guardar

Figure 20: Contact Database Change Form

Important: The contact database structure that can be used as a substitute should be similar to the contact database to be replaced.

Once the database rotation is carried out it is necessary to activate the campaign again.

11.1.9 Inbound Campaigns

When talking about incoming calls we have to explain each functionality applicable to the flow of a incoming call. As we know an incoming call can go through a series of “nodes” until finally connecting with an agent. Therefore we will extend the concept of “incoming campaigns” to the following configuration items.

Inbound campaign creation

To create a new inbound campaign you must enter the menu item Inbound Campaigns -> New Campaign.

The first stage looks like figure 1.

CAMPAÑA ENTRANTE: DATOS BÁSICOS

Campaña Datos básicos de la cola Opciones de calificación Dar permiso a supervisores Adicionar agentes

Nombre:

Base de Datos de Contactos:

Los espacios, tildes, ñ-es y otros caracteres no ASCII no están permitidos.

Tipo de Interacción: ☒ Formulario ☐ Url externa

Url Externa:

Objetivo:

Mostrar nombre al recibir llamado: ☒

Habilitar video: ☐

Sistema externo:

ID en sistema externo:

Ruta Saliente:

CID Ruta Saliente:

Speech:

Paso siguiente

Figure 1: Campaign Parameters

- **Name:** campaign name
- **Contact database:** (optional field) contact database that the inbound campaigns can use to take and display extra data to the telephone number of person that makes the incoming call.
- **Type of interaction:** Here you select if the campaign is going to use a campaign form or will fire an HTTP request towards an external CRM system.
- **External URL:** in case of selecting an external URL interaction for each call, this field selects to wich CRMs should the campaign invoke.

- **Scope:** It is defined as the number of *positive steps* expected in the campaign. In the campaign monitoring, it is displayed in real time as the percentage of progress of the campaign with respect to this one.
- **Show name when called:** We enabled here the option to show the name of the campaign for which the call is coming.
- **Enable video:** Enable video if available.
- **External system:** external CRM system linked to launch *click to call actions or dispositions* on campaign contacts, through the API.
- **ID on external system:** this fields must have the campaign's ID in the external system from where the *click to call* or the *disposition requests* will be sent.
- **Outbound Routes:** An existing outbound route is assigned to an campaign.
- **CID on Outbound Routes:** This field must contain the CID assigned for an existing outbound route to a campaign.
- **Speech:** The campaign speech to be displayed on the agent console in the campaign calls.

Note: In incoming campaigns the contact base is used if you wish to display the additional information of each contact making an incoming call. Although to achieve this the callerid of the incoming call must match the telephone number of a contact at the base of the campaign. Assigning a contact base to an incoming campaign is optional

Then we continue with the creation of our incoming campaign.

CAMPAÑA ENTRANTE: DATOS DE COLA

Campaña Datos básicos de la cola Opciones de calificación Dar permiso a supervisores Agregar agentes

Parámetros de la campaña

Cantidad Max de llamadas: <input type="text"/>	Grabar llamados: <input type="checkbox"/>	Importancia de campaña: <input type="text"/>	Estrategia de distribución: <input type="text"/>
Musica de espera: <input type="text"/>	Tiempo de espera en cola: <input type="text"/> En segundos	IVR de escape: <input type="text"/>	Nivel de Servicio: <input type="text"/>
Destino en caso de timeout: Tipo destino: <input type="text"/>	Destino: <input type="text"/>	Anuncio de entrada para el agente: <input type="text"/>	

Anuncios para llamantes

Audio de ingreso: <input type="text"/>	Anunciar tiempo promedio espera: <input type="text" value="No"/>
Anunciar Posición: <input type="checkbox"/>	Frecuencia de anuncios de espera/posición <input type="text"/> En segundos
Anuncio periódico: <input type="text"/>	Frecuencia de anuncio periodico: <input type="text"/> En segundos

Parámetros de agentes

Tiempo de Ring: <input type="text"/> En segundos	Tiempo de Reintento: <input type="text"/> En segundos
Tiempo de descanso entre llamadas: <input type="text" value="2"/> En segundos	

[primer paso](#) [paso anterior](#) [Paso siguiente](#)

Figure 2: Campaign Parameters

- **Max amount of calls:** define the maximum number of callers who can be waiting in the queue at the same time (0 for unlimited). If not set to 0, and the maximum capacity is reached, additional calls would be sent to the failover destination
- **Enable recordings:** Enable all campaign calls to be recorded.
- **Campaign priority:** gives campaign a *priority* (priority level). The higher the weight, the higher the priority. (Default = 0) If there are agents common to multiple campaigns, the campaign with the highest priority will deliver its calls first.
- **Distribution strategy:** A strategy for how to handle the queue and divide calls between queue members
- **Music on hold:** Configures wich Playlist will be played for the client when his call is put on hold.
- **Queue timeout:** when the *Queue timeout* of a caller is hit, they will be pulled out of the queue immediately to the failover destination
- **IVR Breakdown** is an IVR destination that user can choose to ‘escape’ to it in a middle of waiting time. This parameter is related to the ‘periodic announcement’ because of the last parameter specify the audio messages that instructs the user how to ‘escape’ to these IVR destination.

- **Service level:** used for service level statistics (calls answered within service level time frame).
- **Failover dst:** set a failover destination here by choosing a valid destination from the drop-down menus. The caller would be sent to this destination if they exit the queue for reasons such as maximum wait time, queue capacity, or join empty/leave empty settings
- **Entry Announcement for Agent:** Audio played to an agent when receiving incoming call.
- The announcement played to callers before they join the campaign queue
- **Announce hold time** allows the user to specify the average hold time that the caller will wait for to be attended by an agent
- **Ring time:** The number of seconds an agent's phone can ring before we consider it a timeout. It should be considered that this parameter makes sense whenever the agent does not work within a group configured with the *Auto-attend inbound calls* mode.
- **Agent callback time:** the number of seconds to wait before trying call to a ready agent
- **Announce position:** Announce the caller's position in the queue.
- **Wait/position announce frequency:** It is the waiting or position time for an announce frequency.
- **Rest time between calls:** It is the rest time (in seconds) that each agent has between each successful call.
- **Periodic Announcement:** we can select some audio from our repository to play as a periodic announcement about the call on hold.
- **Announcement frequency:** how often (seconds) to announce a voice menu to the caller.

Note: The parameters “Ring time” and “Retry time” are left without effect for those calls delivered to agents whose group of agents is configured with the auto-answering of incoming calls attribute or for campaigns that have this option enabled in Agent Settings.

In the next configuration stage, the dispositions that the campaign requires for agents to rate each contact call must be assigned.

CAMPAÑA ENTRANTE: OPCIONES DE CALIFICACIÓN

[Campaña](#)
[Datos básicos de la cola](#)
[Opciones de calificación](#)
[Dar permiso a supervisores](#)
[Agregar agentes](#)

Equivocado	Sin acci...	-----	Ocultar <input type="checkbox"/>	Remover
No contesta	Sin Acci...	-----	Ocultar <input type="checkbox"/>	Remover
Venta	Gestión	venta	Ocultar <input type="checkbox"/>	Remover

[Agregar campo](#)

[primer paso](#)
[paso anterior](#)
[Paso siguiente](#)

Figure 3: Campaign Call Dispositions

In the next step, we add supervisors to the campaign:

CAMPAÑA ENTRANTE: ADICIONAR SUPERVISORES

[Campaña](#) [Datos básicos de la cola](#) [Opciones de calificación](#) [Dar permiso a supervisores](#) [Adicionar agentes](#)

Agregar Supervisores:

☐ Seleccionar Todos

☐ admin: ☐ mgarridoe: Miguel Garrido

[primer paso](#) [paso anterior](#) [Paso siguiente](#)

Figure 4: Supervisors Assignment

Finally, agents can be assigned to the campaign.

CAMPAÑA ENTRANTE: ADICIONAR AGENTES

[Campaña](#) [Datos básicos de la cola](#) [Opciones de calificación](#) [Dar permiso a supervisores](#) [Adicionar agentes](#)

Agregar grupo de agentes

Grupo:

agentespordefault ▾

Agregar grupo

Agente	Penalty	
Perico de los Palotes ▾	0 ▾	<div>Remove</div>
Fulanito De Tal ▾	0 ▾	<div>Remove</div>

Agregar agente

[primer paso](#) [paso anterior](#) [Finalizar](#)

Figure 5: Agent Assignment

The rest of the chapter details all about routing calls from trunk to our incoming campaigns.

Ringin VS Automatic incoming call attention

The function mode of the agent's Webphone against a call coming from an incoming campaign can be:

- **Ringin** with a duration associated with the * Ring time * parameter. During that time the agent's phone notifies the incoming call, awaiting the action of the agent that determines the attention or not of the call.
- **Auto-attend inbound call.** This behavior implies that each incoming call sent to an agent is automatically answered by the agent's phone notifying him with a *beep* before leaving it in line with the counterpart of the call.

This behavior depends on the *Agent Group* level configuration that the agent linked to the incoming campaign has. So if the group has activated the auto attend for inbound calls, the webphone will answer automatically the calls of any campaign, making no effect the parameters "Ring time" and "Agent callback time" as is mentioned in the note of this section

Nombre:

Despausar automaticamente: En segundos

Auto atender entrantes: ☒

Auto atender dailer: ☒

Forzar calificación: ☐

Llamada fuera de campaña: ☐

Permitir el acceso a las grabaciones: ☒

Figure 6: Agents Group Configuration

español (es)

CAMPAÑAS ENTRANTES

Campañas Activas

Id	Nombre	Base de datos	
2	CampaPrue	BD_contacto_cdc0a24b-6b4d-4985-b305-bcb35556e657 (0 contactos)	
5	campapruetest1	BD_contacto_18df120d-8e16-4623-90e4-1f1e099cdac5: (0 contactos)	
18	Encuesta_1	BD_contacto_24cac7c0-f298-462d-85a9-f9b6a00a1ecb: (0 contactos)	Activa Opciones ▾

+ Agregar agentes

⦿ Dar permiso para ver

📋 Calificaciones

📊 Resultados de Base

📊 Reportes

+ Agregar Contacto

+ Agregar Contactos

+ Restringir campos de contacto

📅 Ver Agendas

✎ Configuraciones para agentes

🗑 Eliminar

Figure 7: Configuration for Agents in Campaign

Configuración de comportamiento de agentes en campaña

Configurar auto atender entrantes:	<input checked="" type="checkbox"/>	Auto atender entrantes:	<input checked="" type="checkbox"/>
Configurar despausar automáticamente:	<input type="checkbox"/>	Despausar automáticamente:	0
Configurar forzar calificación:	<input type="checkbox"/>	Forzar calificación:	<input type="checkbox"/>

Seleccione las opciones a configurar para cambiar el comportamiento de los agentes en esta campaña.

Guardar

Figure 8: Configuration for Agents in Campaign

Inbound routes

Una vez disponible nuestra campaña entrante, se debe proceder con la vinculación de un número telefónico “DID” disponible en alguno de los troncales SIP por los que llegarían las solicitudes de llamadas con la campaña entrante en cuestión. Es decir, la asignación del número y la campaña entrante hacia donde derivar a todas las llamadas que ingresen a OMniLeads sobre dicho número DID.

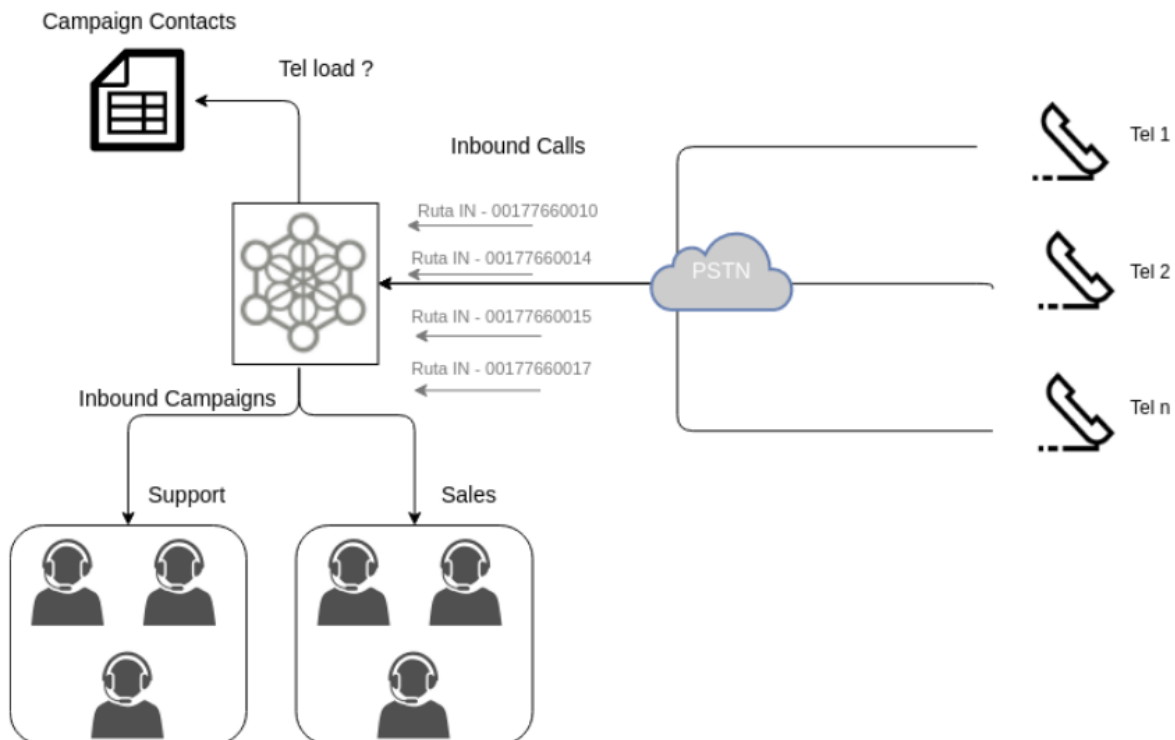


Figura 1: Rutas entrantes

Para generar una nueva ruta de llamadas entrantes, debemos acceder al menú *Telefonía -> Rutas entrantes*, en donde se listan las rutas creadas y además se pueden añadir nuevas.

En la siguiente figura, se puede visualizar una *ruta entrante* en su pantalla de configuración:

Editar ruta entrante

Nombre:	Número DID:
<input type="text" value="Despegar"/>	<input type="text" value="01112345678"/>
Prefijo caller id:	Idioma:
<input type="text"/>	<input type="text" value="Español"/>
Tipo de destino:	Destino:
<input type="text" value="Campaña entrante"/>	<input type="text" value="Campaña entrante: Despegar_IN"/>

Figura 2: *Parámetros de ruta entrante*

- **Nombre:** Es el nombre asignado a la ruta (alfanumérico sin espacios).
- **Número DID:** Es el número entrante que se valida como disparador del encaminamiento de la llamada sobre el destino seleccionado por la propia ruta.
- **Prefijo caller id:** El valor que se configure en éste campo, aparecerá como prefijo del CallerID que llega en cada llamada por el troncal correspondiente.
- **Idioma:** El idioma que se utiliza a la hora de reproducir locuciones por defecto del sistema, sobre los canales que ingresan por la ruta.
- **Tipo de destino:** El tipo de destino a donde se enviarán las llamadas ingresadas por dicha ruta. Dentro de los tipos de destinos existen campañas entrantes, IVRs, condicionales de tiempo, identificación de llamante, etc.
- **Destino:** Destino puntual al cual enviar los canales.

Es importante aclarar que está permitido que varias rutas tengan el mismo destino.

Incoming calls from the PBX

En ésta sección, se ejemplifica cómo configurar OMniLeads y una PBX basada en Asterisk, para derivar llamadas desde la PBXa hacia OMniLeads.

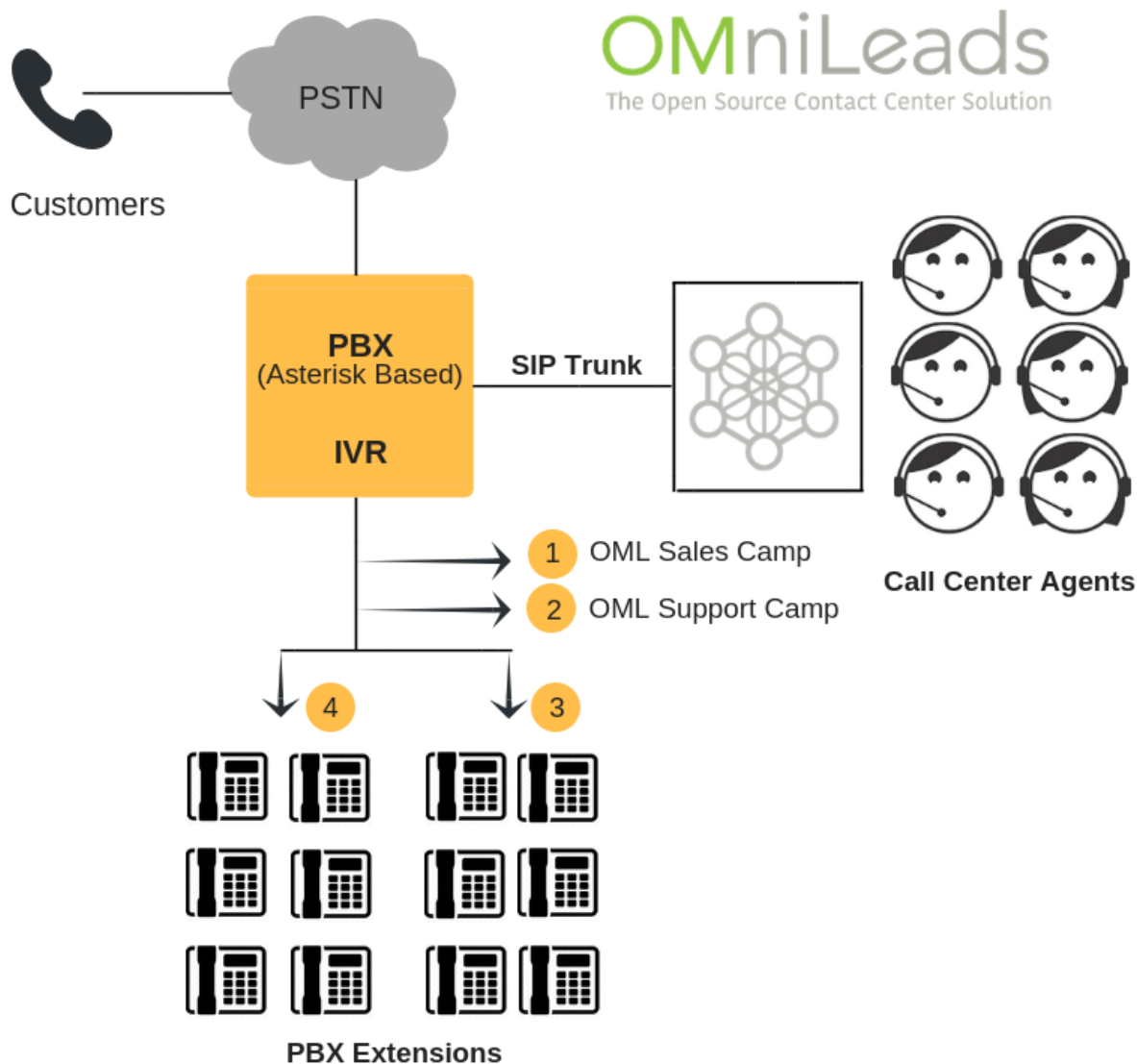


Figura 1: Derivación de llamadas desde PBX

Partimos del hecho de considerar que existe un troncal SIP que vincula OMniLeads con la PBX.

Lo primero que se debe definir es la numeración asignada a la ruta entrante que va a procesar la llamada desde la PBX hacia un destino de OMniLeads, ya que éste número (DID de la ruta) debe ser marcado por la PBX para enviar llamadas desde cualquier módulo (extensiones, IVRs, rutas entrantes, follow me, etc.) de dicha PBX, hacia el destino configurado en la ruta de OMniLeads:

Name:	DID number:
<input type="text" value="Sales"/>	<input type="text" value="123456"/>
Caller id prefix:	Language:
<input type="text"/>	<input type="text" value="English"/>
Destination type:	Destination:
<input type="text" value="Inbound campaign"/>	<input type="text" value="Inbound campaign: In02"/>
<input type="button" value="Save"/>	

Figura 2: Parámetros de ruta entrante

Tomando como ejemplo la ruta con el DID *123456* utilizado en la figura anterior, la PBX Asteisk deberá generar llamadas por el troncal SIP hacia el número mencionado, cada vez que algún recurso de la PBX necesite alcanzar el destino *123456* de OMniLeads.

Si nuestra PBX Asterisk dispone de una interfaz web de configuración, entonces simplemente podemos generar una *Nueva extensión custom* y hacer que la misma apunte a *SIP/trunkomnileads/123456*, donde “trunkomnileads” es el nombre configurado en la PBX para nombrar el troncal SIP con OMniLeads.

La idea de extensión apuntando hacia OMniLeads se ejemplifica en las siguientes figuras 3 y 4:

Add an Extension

Please select your Device below then click Submit

- Device

Device	<input type="text" value="Other (Custom) Device"/>
<input type="button" value="Submit"/>	

Figura 3: Extensión custom para OMniLeads

Si bien la extensión en la PBX puede tener cualquier numeración (se ejemplificó con *2222*), lo importante es enviar *123456* (en nuestro ejemplo) hacia OMniLeads, como se resalta en la siguiente figura:

Add CUSTOM Extension

- Add Extension

User Extension ?	2222
Display Name ?	OMniLeads Sales Camp
CID Num Alias ?	
SIP Alias ?	

+ Extension Options

+ Assigned DID/CID

- Device Options

This device uses custom technology.

dial ? SIP/omnileads/123456

Figura 4: Extensión custom para OMniLeads

Una vez disponible la extensión en la PBX, sólo es cuestión de invocarla desde cualquier módulo de la PBX, como por ejemplo un IVR:

Edit IVR: welcome

Used as Destination by 1 Object: ?

 Delete IVR: welcome











- IVR General Options

IVR Name ? welcome

IVR Description ? welcome

+ IVR Options (DTMF)

- IVR Entries

Ext ?	Destination	Return ?	Delete
4	Queues ▼	502 <502> ▼	 ? 
3	Ring Groups ▼	IT office <600> ▼	 ? 
2	Extensions ▼	<2222> OMniLeads Sales Camp ▼	 ? 
1	Extensions ▼	<321321> OML Customers ▼	 ? 
digits pressed	== choose one == ▼	 ? 	




Figura 5: Derivación a OMniLeads desde IVR

Si bien en la figura anterior se ejemplifica la derivación de llamadas hacia campañas entrantes de OMniLeads desde un IVR de la PBX, podemos concluir que también las extensiones de la PBX pueden marcar o transferir llamadas hacia OMniLeads, así como también módulos de la PBX como condiciones horarias, follow me, rutas entrantes, etc. podrán invocar una *extensión custom* de la PBX que derive llamadas hacia OMniLeads.

Incoming call flow based upon time and date

En ésta sección, se trabaja con el concepto de *Enrutamiento de llamadas entrantes condicionado por rango de tiempo*, para configurar el flujo de llamadas entrantes hacia diferentes destinos internos, a partir de comparar la fecha/hora en que llega una llamada y un patrón (de fechas y horas) relacionado, de manera tal que se pueda planificar de antemano si una llamada debe ir hacia un destino u otro de acuerdo al resultado de dicha comparación.

Por ejemplo, una llamada podría ir hacia una campaña entrante *dentro del rango de fecha y horario de atención al cliente* y hacia un IVR que reproduzca un anuncio acerca de los horarios de atención, cuando la llamada ingrese fuera de ese rango definido.

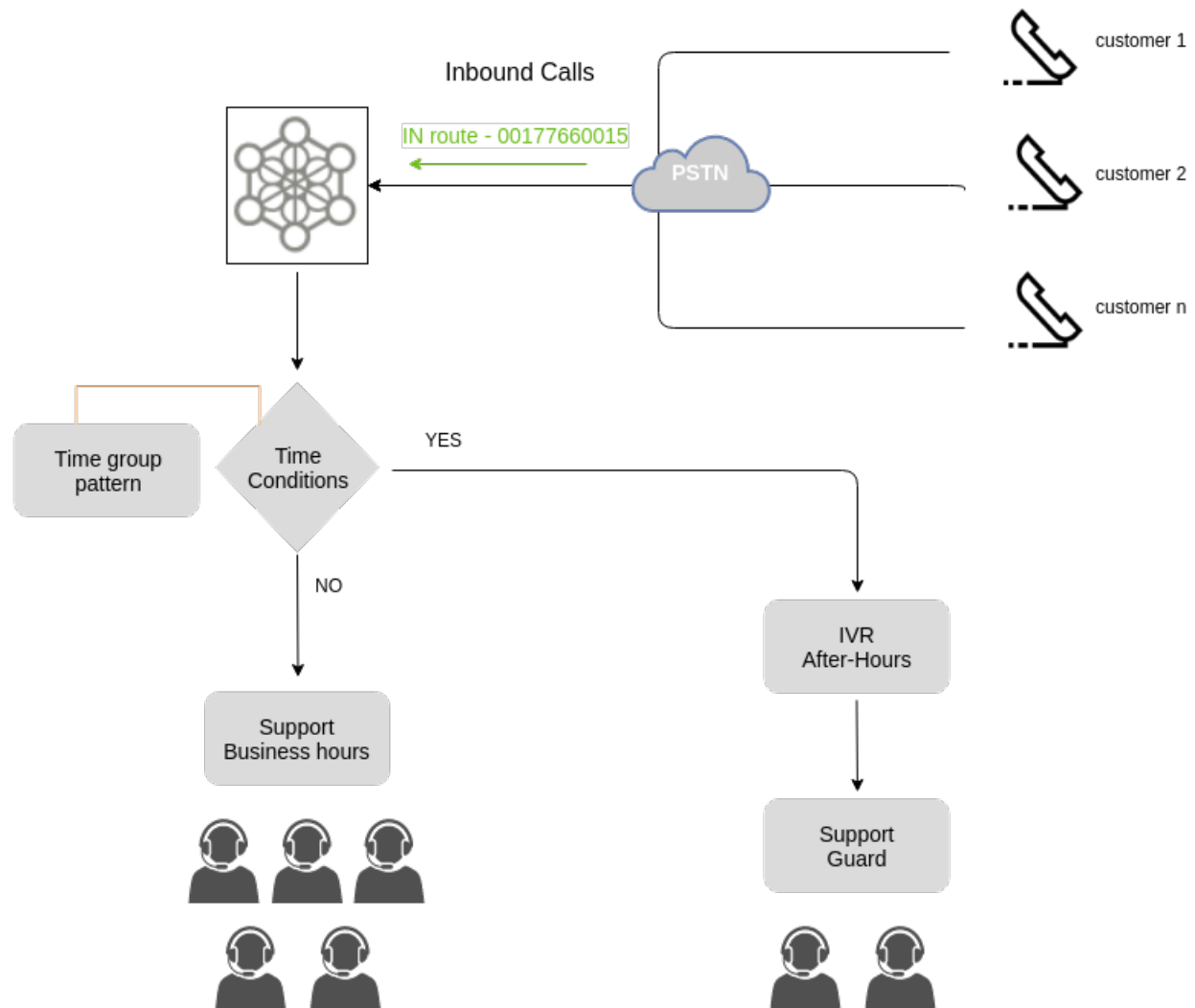


Figura 1: Condición de tiempo

Existen 2 módulos que trabajan en conjunto y que permiten implementar éste tipo de configuraciones, los cuales se detallan a continuación.

Time groups

Éste módulo permite agrupar patrones de fechas y horas como un objeto, para luego poder ser invocados por los objetos del tipo condicionales de tiempo.

Para definir o editar grupos de horarios, se debe acceder al menú *Telefonía -> Grupos horarios*. Para añadir un nuevo grupo se debe presionar el botón “Agregar grupo horario”.

La pantalla de grupos horarios se expone en la siguiente figura:

Editar grupo horario

Nombre:

Psicologos

Condiciones de tiempo

Tiempo inicio

08:00

Tiempo final

20:00

Día de la semana inicio

Lunes



Día de la semana final

Viernes



Remover

Día del mes inicio

1



Día del mes final

31



Mes inicio

Enero



Mes final

Diciembre



Tiempo inicio

09:00

Tiempo final

13:00

Día de la semana inicio

Sábado



Día de la semana final

Sábado



Remover

Día del mes inicio

1



Día del mes final

31



Mes inicio

Enero



Mes final

Diciembre



Agregar validación

Figure 2: Grupos horarios

Una vez generados los *grupos horarios*, podemos invocarlos desde el módulo complementario *Validaciones horarias*.

Validaciones horarias

Este módulo permite comparar la fecha y hora en el momento de procesar una llamada, con un grupo horario asignado como patrón de comparación. Luego, en base a la coincidencia o no con alguna franja de fecha/hora del grupo, la llamada se envía hacia el destino positivo o negativo de la comparación.

Un *nodo* condicional de ésta clase puede ser invocado por otros nodos:

- Rutas entrantes
- Opción de IVR
- Failover de una campaña entrante
- Otro condicional de tiempo

Para generar un elemento condicional de tiempo, se debe acceder al menú *Telefonía -> Validaciones horarias*.

La pantalla de configuración es similar a la siguiente figura:

Editar Validación Horaria

Nombre:	Grupo horario:
<input type="text" value="Psicologos"/>	<input type="text" value="Psicologos"/>

Destino si cumple la condición de tiempo Tipo de destino: <input type="text" value="IVR"/> Destino: <input type="text" value="IVR: PruebaKonecta"/>	Destino si NO cumple la condición de tiempo Tipo de destino: <input type="text" value="IVR"/> Destino: <input type="text" value="IVR: Fuera de horario Psicologos"/>
--	---

Guardar

Figure 3: Validaciones horarias

Finalmente, tenemos disponible éste elemento de enrutamiento para ser utilizado por ejemplo, como destino de una ruta entrante.

IVR - Interactive Voice Response

Los IVRs hacen posible que la persona que realiza una llamada entrante, pueda seleccionar un destino apropiado en base a informar los mismos a través de una grabación y aguardando la interacción a través de los tonos de teclado DTMF. Con ésta herramienta, un administrador puede enrutar llamadas entrantes hacia un IVR y configurar el mismo para que diferentes DTMFs se conmuten hacia diferentes campañas entrantes, IVRs, condicionales de fecha y hora, etc.

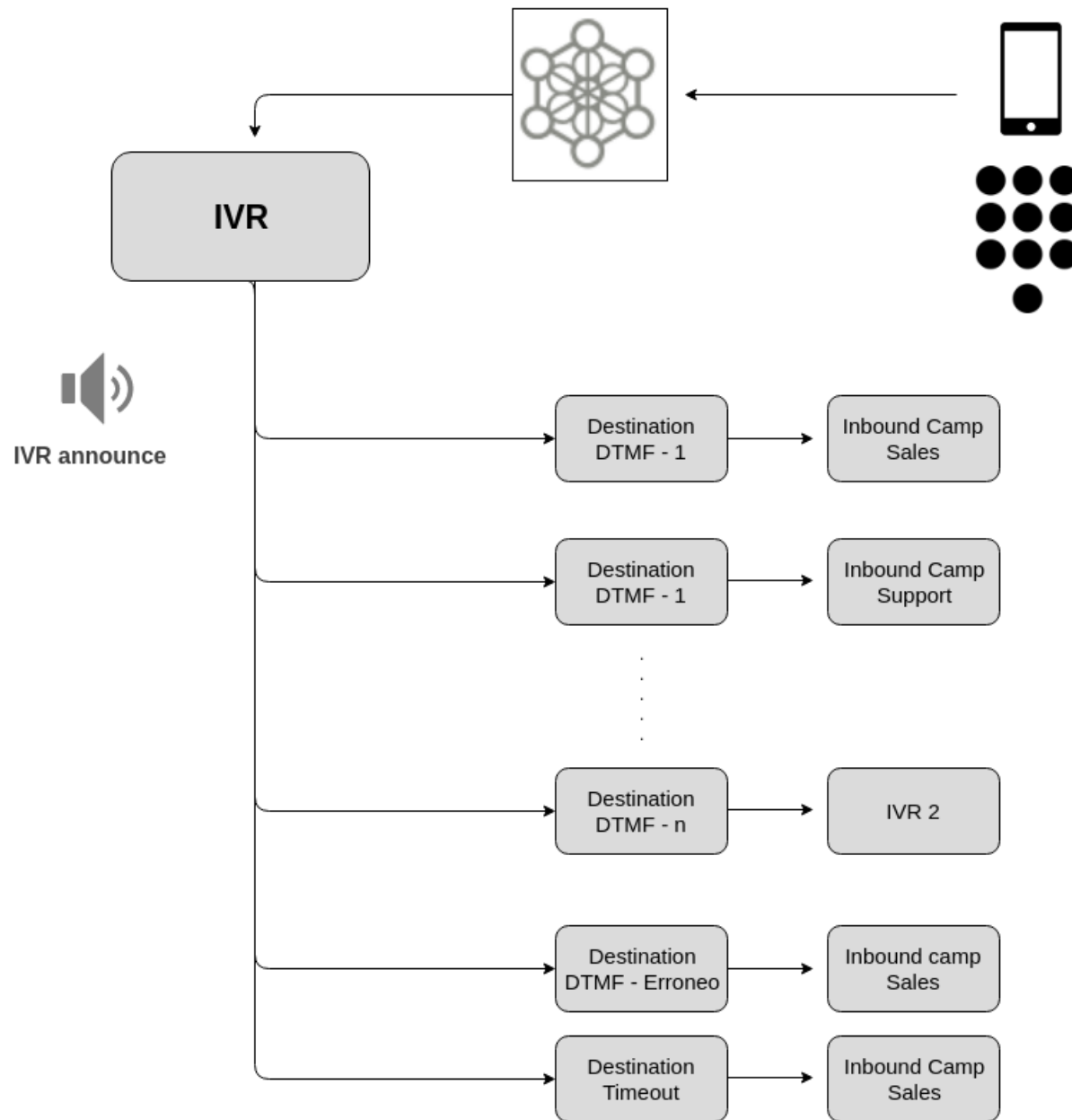


Figura 1: IVR

Para generar un nuevo IVR, se necesita como mínimo un audio a reproducir (disponible en la biblioteca de audios) y además un destino por defecto hacia donde enviar las llamadas que pasen por dicho IVR.

Para añadir un IVR, se debe acceder al menú *Telefonía -> IVR*, y seleccionar “Agregar IVR”. Se desplegará una pantalla similar a la siguiente figura:

Crear IVR

Sección general

Nombre:

Descripción:

☒ Archivo interno
☐ Archivo externo

Configuración time out

Time out:

Time out retries:

En segundos

Time out audio:

Time out ext audio:

☒ Archivo interno
☐ Archivo externo

Tipo de destino para time out:

Destino time out:

Configuración destino inválido

Invalid retries:

Invalid audio:

Invalid destination ext audio:

☒ Archivo interno
☐ Archivo externo

Tipo de destino para destino inválido:

Destino inválido:

Opciones de destinos

DTMF:

Tipo de destino:

Destino:

Figure 2: Parámetros de IVR

El formulario se puede dividir en 3 secciones, donde la primera sección contiene los siguientes campos:

- **Nombre:** Es el nombre del objeto, con el cual se referencia en el listado de IVRs.
- **Descripción:** Campo opcional dedicado a un comentario aclaratorio sobre el objeto.
- **Archivo interno:** Se selecciona ésta opción en caso de querer seleccionar como audio principal del IVR, un archivo previamente subido por el módulo de audios de OML.
- **Archivo externo:** Se selecciona ésta opción en caso de querer seleccionar como audio principal del IVR, un archivo y subirlo en el mismo instante al sistema.

Luego siguen las secciones para configurar acciones de timeout y opciones inválidas:

- **Time out:** Es la cantidad de segundos que se aguarda a que el “llamante” introduzca un DTMF a partir de la finalización de la reproducción del audio del IVR.
- **Time out retries:** Es la cantidad de intentos que el IVR ofrece al “llamante”, a la hora de dar falla por “time-out”. Es decir, se permite una cierta cantidad de intentos fallidos por timeout, para luego ejecutar la acción referenciada en “Tipo de destino para time out”.
- **Time out audio:** Cada vez que se da un timeout por no ingreso de DTMF, se puede reproducir un audio subido previamente al módulo de audio de OML, que indique el error.
- **Time out ext audio:** Cada vez que se da un timeout por no ingreso de DTMF, se puede reproducir un audio que se puede seleccionar y subir en el momento, que indique el error.
- **Tipo de destino time out:** En caso de cumplirse la cantidad de “retries” de timeout, la llamada es enviada por el IVR hacia el tipo de destino por defecto para los timeouts de IVR. En éste campo se indica dicha clase de destino.
- **Destino time out:** Finalmente, se selecciona puntualmente el objeto dentro de la familia de tipo de destino.

- **Invalid retries:** Es la cantidad de intentos que el IVR ofrece al “llamante” a la hora de dar falla de destino ingresado inválido. Es decir, se permite una cierta cantidad de intentos fallidos por opción inválida, para luego ejecutar la acción referenciada en “Tipo de destino para destino inválido”.
- **Invalid audio:** Cada vez que se da un ingreso de opción inválida, se puede reproducir un audio subido previamente al módulo de audio de OML, que indique el error.
- **Invalid destination ext audio:** Cada vez que se da un ingreso de opción inválida, se puede reproducir un audio que se puede seleccionar y subir en el momento, que indique el error.
- **Tipo de destino para destino inválido:** En caso de cumplirse la cantidad de “retries” de opción incorrecta, la llamada es enviada por el IVR hacia el tipo de destino por defecto para los ingresos de opciones incorrectas del IVR. En éste campo se indica dicha clase de destino.
- **Destino inválido:** Finalmente, se selecciona puntualmente el objeto dentro de la familia de tipo de destino.

Finalmente, la tercera sección despliega tantas filas como opciones de DTMF implique el IVR.

Por cada fila, se puede asignar un DTMF a un destino de conmutación de la llamada.

Como bien se conoce, la idea es seleccionar un “tipo de destino” y un “destino particular dentro del tipo”, para cada DTMF del IVR que se brinda como opción.

Finalmente, al seleccionar el botón guardar, se dispondrá del IVR.

Note: It is possible to nest an IVR inside another.

Customer identification on incoming calls

Presentation

Dentro del flujo de una llamada entrante, se puede añadir un nodo con la funcionalidad de *identificación de clientes*. Es decir, lanzar una *solicitud de identificación* sobre cada llamada entrante procesada, implementando a su vez la posibilidad de consultar a un sistema de gestión CRM externo y aguardar que éste último determine una decisión de encaminamiento de las llamadas entrantes provenientes del exterior, a través de una interacción entre OMniLeads y dicho sistema de gestión.

En su funcionamiento más básico, el módulo implementa la posibilidad de solicitar la identificación de un cliente que se ha comunicado a la compañía mediante el ingreso de tonos del teléfono *DTMF*, luego se comprueba si se ha ingresado algún valor, y finalmente se envía la llamada hacia un destino concreto en caso positivo, y hacia otro destino si el resultado fue negativo. Si la configuración involucra interacción con CRM, entonces el módulo envía el *número ingresado* por el cliente hacia el CRM y espera por una respuesta de éste mismo, para decidir hacia donde encaminar la llamada.

En ambos escenarios (modo básico y modo interactivo con CRM), se consigue que la llamada ingrese al agente con el *ID del cliente* como índice para obtener toda la información del cliente y desplegarlo en la pantalla de agente (ya sea sobre la vista de contacto o en el CRM externo configurado para la campaña).

El módulo permite 3 modos de funcionamiento, los cuales se explican a continuación.

Sólamente solicitar identificación

Bajo ésta configuración, cuando una llamada entrante es enviada hacia éste nodo, se lanza una solicitud de identificación a través de un audio reproducido sobre el canal telefónico del cliente que originó la llamada, para luego simplemente validar si el cliente ingresó o no un valor, y así tomar una decisión de encaminamiento hacia las 2 alternativas posibles: *Destino A* si se ha ingresado una identificación, o *Destino B* si el llamante no lo hizo.

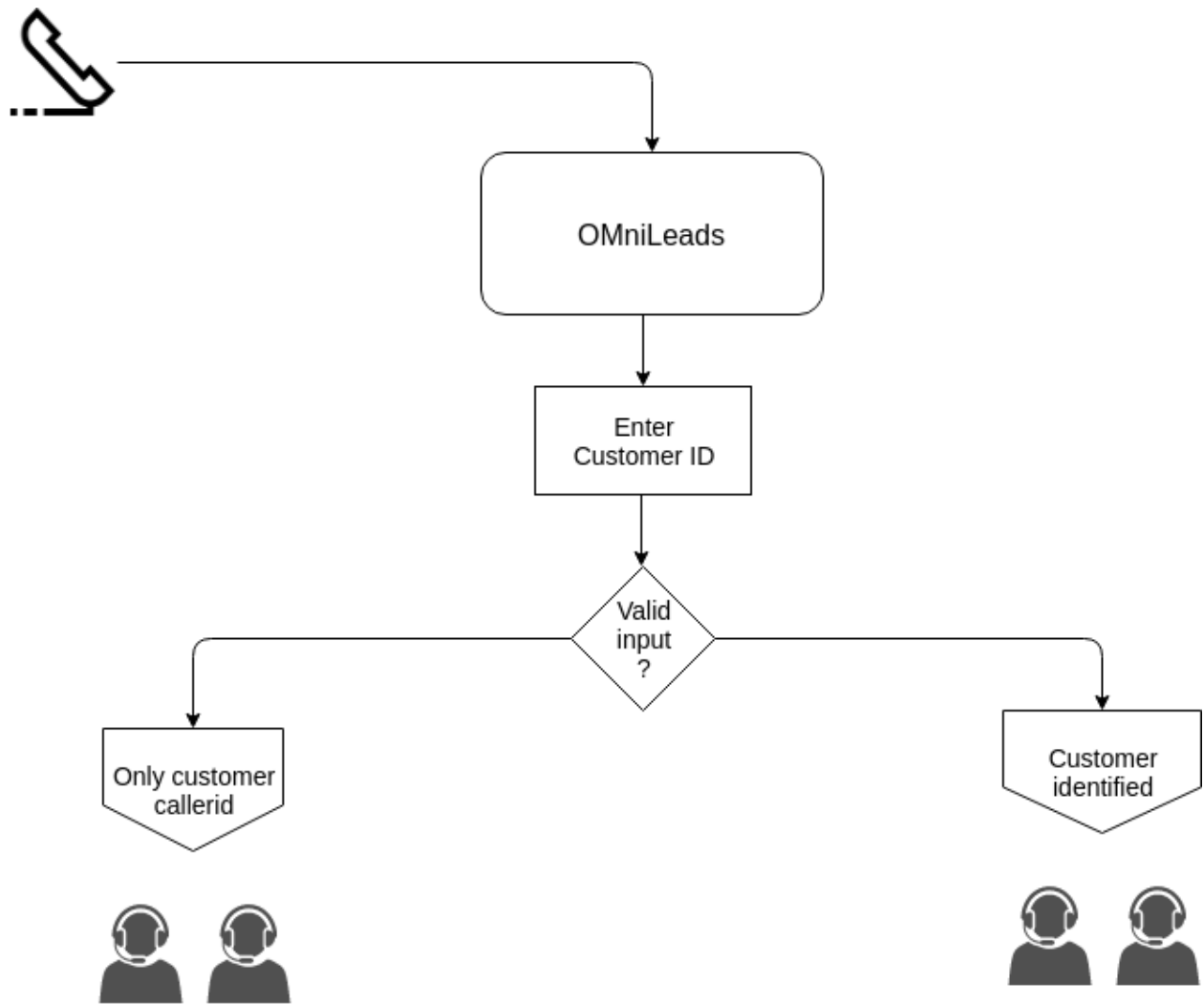


Figura 1: ID de cliente sin interacción con CRM

Solicitar identificación, notificar al CRM y aguardar respuesta *true/false*

Bajo ésta configuración, cuando una llamada entrante es enviada hacia éste nodo, se lanza una solicitud de identificación a través de un audio reproducido sobre el canal telefónico del cliente que originó la llamada, para luego lanzar una consulta hacia un servicio web (CRM) previamente configurado.

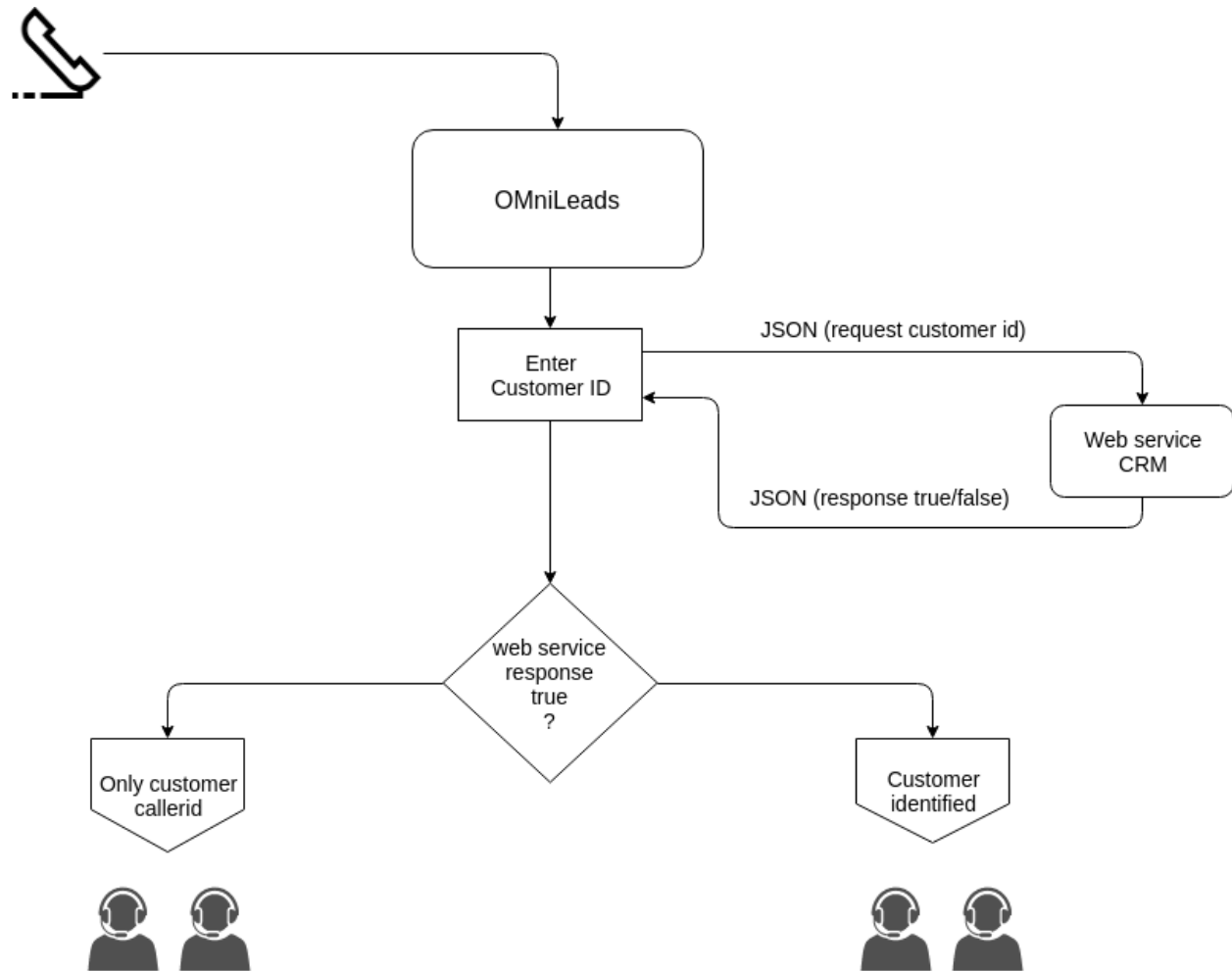


Figura 2: ID de cliente con interacción true/false con CRM

Aquí, el sistema de gestión CRM toma partido en lo que respecta al encaminamiento de los clientes que llaman a la compañía, ya que a partir de recibir desde OMniLeads la *clave de identificación del llamante*, el CRM debe responder al *request* recibido con una respuesta del tipo *true/false*, por lo cual OMniLeads luego debe encaminar las llamadas hacia cada uno de los 2 posibles destinos previamente configurados (*Destino A* si el CRM devuelve *true* o *Destino B* si el CRM devuelve *false*).

Un ejemplo podría ser el de una compañía que comprueba si el “número de cliente” se encuentra al día con los pagos del servicio, y en base a ello encaminar la llamada hacia una campaña con mayor o menor prioridad en términos de tiempo de espera en cola.

The external CRM choose the OMniLeads destination where route customer calls

Bajo éste modo, cuando una llamada entrante invoca la ejecución del nodo, éste último procede con la solicitud de identificación para luego validar si el cliente ingresó o no un valor, y en caso de haber ingresado, se ejecuta una consulta hacia un servicio web (CRM) previamente configurado.

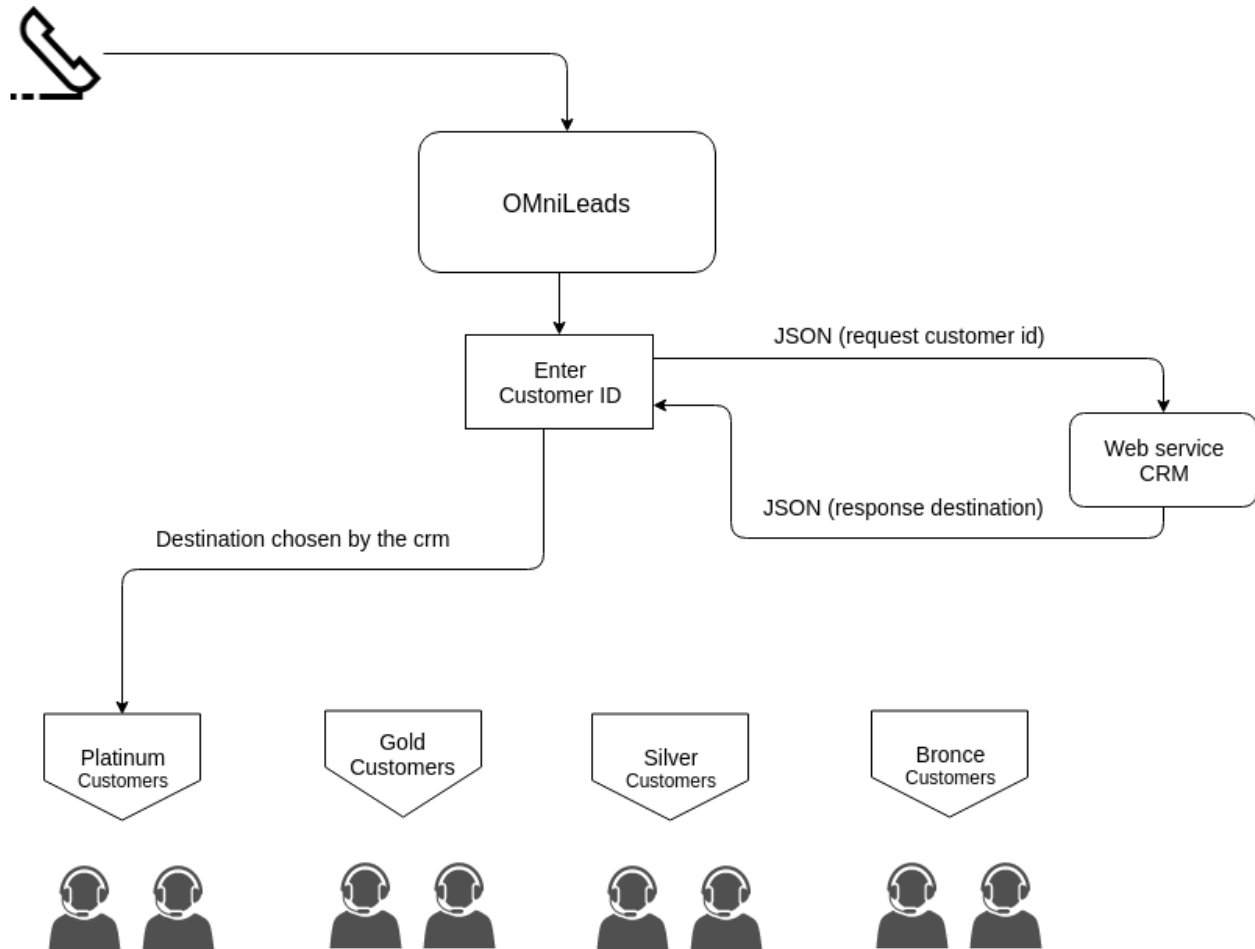


Figura 3: ID de cliente con interacción destino elegido CRM

Then the CRM choose the OMniLeads destination for the customers who call the company. Since after receiving the caller's identification key from OMniLeads, the CRM must respond to the request received with a OMniLeads internal destination like response, finally OMniLeads must route the calls to the destination selected by CRM

Un aplicación de esta funcionalidad podría ser, notificar al CRM acerca del *número de cliente* ingresado y que éste último decida hacia qué campaña entrante de OMniLeads enviar la llamada, utilizando como criterio el plan de suscripción que tiene contratado ese *número de cliente*.

Create a new customer identification node

Para generar un nuevo nodo, se debe acceder al menú *Telefonía -> Identificación de clientes*:

Crear Identificador de Clientes

Nombre:

Tipo de interacción:

Sin interacción externa



Url servicio identificación:

Audio:



Longitud de id esperado:

Timeout:

5

Intentos:

1

Destino si se identifica correctamente

Tipo de destino:



Destino:



Destino si NO se identifica correctamente

Tipo de destino:



Destino:



Guardar

Figura 4: Formulario de identificador de clientes

A continuación, se detallan los campos del formulario:

- **Nombre:** Nombre de referencia del nodo.
- **Tipo de interacción:**
 - Sin interacción: Sólo se comprueba si hubo o no un ingreso y su longitud.
 - Interacción externa tipo 1: Se envía ID de cliente y se espera “true/false” como respuesta.
 - Interacción externa tipo 2: Se envía ID de cliente y se espera un destino como respuesta.
- **URL servicio identificación:** Aquí, se indica la dirección web del servicio hacia donde enviar el número de identificación.
- **Audio:** Se trata del audio que se reproduce sobre la llamada entrante, para solicitar la identificación.
- **Logitud de ID esperado:** Se puede indicar el largo esperado del código de identificación.
- **Timeout:** El tiempo en segundos que el sistema espera a que se ingrese la identificación. En caso de expirar éste tiempo, se comprueba si ya se ha sobrepasado la cantidad de re-intentos para así ejecutar una nueva petición o derivar la llamada hacia el destino no exitoso.
- **Intentos:** La cantidad de intentos erróneos que se permiten al ingresar la identificación.
- **Destino si se identifica correctamente:** Tipo de destino y destino puntual para dicho tipo, al que se derivan las llamadas “positivas” en los tipos de interacción “sin interacción” e “interacción externa tipo 1”.
- **Destino si NO se identifica correctamente:** Tipo de destino y destino puntual para dicho tipo, al que se derivan las llamadas “negativas” en los tipos de interacción “sin interacción” e “interacción externa tipo 1”.

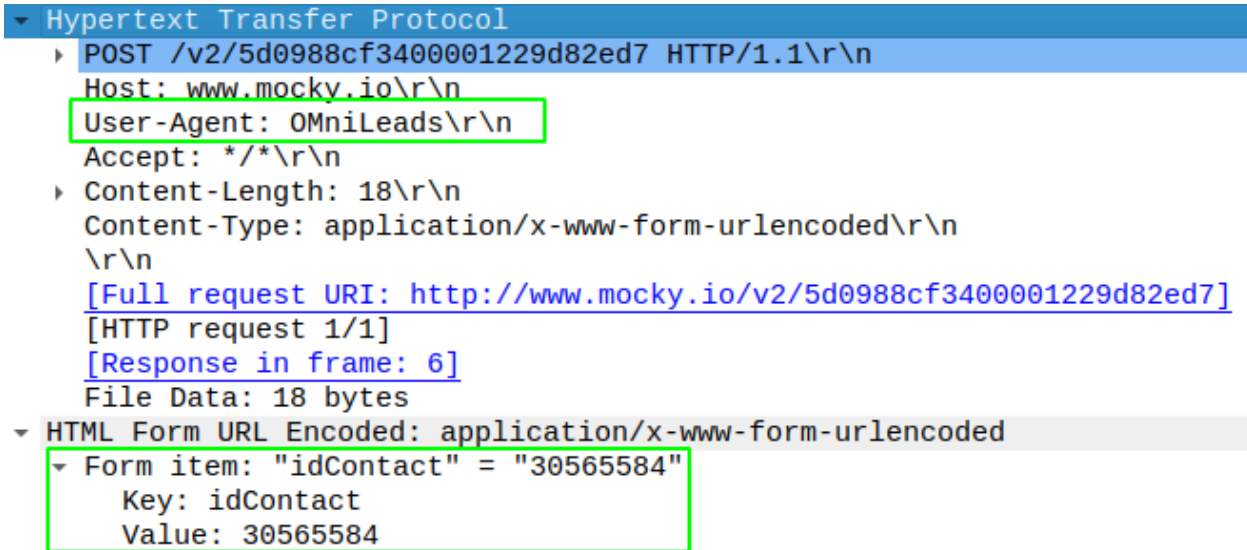
Important: Para poder implementar los modos que implican enviar la identificación hacia un servicio web externo, aguardando una respuesta del mismo para luego ejecutar el encaminamiento de la llamada, depende de que el sistema de gestión implemente un servicio web para recibir las peticiones de éste tipo.

Para los desarrolladores que deseen habilitar en el sistema de gestión éste tipo de interacción, pueden encontrar el formato en que OMniLeads envía la identificación hacia el servicio web en la siguiente sección: [Routing request to the external CRM](#).

Routing request to the external CRM

Ésta interacción implica que OMniLeads ejecute una solicitud HTTP-POST (plain/text) hacia la URL del sistema de gestión especificado dentro del módulo *Create a new customer identification node*, es decir en la definición de un nodo “identificación de clientes”.

Éste POST enviado hacia el sistema de gestión CRM, tiene el siguiente aspecto:



Como podemos observar el campo “User-Agent” debe llegar como “OMniLeads”, y en el cuerpo del POST, el número de identificación ingresado en la llamada se envía como “idContact”.

Answer generated by the web service’s external CRM

El servicio recibe de OMniLeads el request HTTP-POST con el número de identificación del cliente, y debe generar una respuesta a dicha solicitud. El sistema tiene la posibilidad de generar 3 tipos de respuestas:

- **true**
- **false**
- **X,Y**: Donde “X” es un número entero y se corresponde con el tipo de destino hacia a donde enviar la llamada identificada, y donde “Y” es el destino puntual para ese tipo de destino. Por ejemplo (1,3) indica que la llamada será enrutada hacia una campaña entrante (1) y puntualmente hacia la campaña entrante cuyo ID es (3). La clave asociada a la respuesta es “response”.

The response’s format must be “JSON”.

- JSON response

```
Content-Type: application/json
HTTP/1.1 200 OK

{
  "status": "ok",
  "destination": "value"
}
```

Donde “status” puede ser *ok* o *fail*, y “destination” podrá ser cualquiera de las 3 respuestas especificadas arriba.

Important: El sistema debe respetar el formato y nombre de parámetros (status y destination).

In case of generate a response with routing destination, the types of destinations are:

- 1: Campaña entrante.
- 2: Validación horaria.
- 3: IVR.

- 5: Hangup de la llamada.
- 9: Solicitud de identificación.

En un futuro se implementará un endpoint de la API para listar cada destino posible por cada tipo de destino. Mientras tanto, el desarrollador que desee implementar el enrutamiento de llamadas basado en la identificación ingresada en la llamada y el request generado desde OMniLeads, podrá ingresar a la interfaz de OMniLeads y en cada módulo (tipo de destino) listar los mismos y observar el ID.

Ejemplo de respuesta con destino de llamada

Se desea validar cada ID enviado desde OMniLeads y responder con 2 posibles tipos de destinos de enrutamiento. Por un lado una campaña entrante llamada “clientes gold” y otra llamada “clientes bronce”.

Para ello, suponemos que existen las 2 campañas entrantes como se indica en la siguiente figura:

Nombre	Base de datos	Estado	Acciones
banco_x_without_idCliente	BD_contacto_9d26c1e7-2d8b-4101-b017-323c4dcf0db0: (0 contactos)	Activa	Opciones ▾
banco_x_with_idCliente	BD_contacto_66456371-b91c-4d6c-89f9-dd2b81fa38de: (0 contactos)	Activa	Opciones ▾
gold_clients	BD_contacto_057fe536-3332-4ca7-88f5-cf415478569a: (0 contactos)	Activa	Opciones ▾
silver_clients	BD_contacto_cc52490f-aa64-4755-a507-aae564968812: (0 contactos)	Activa	Opciones ▾

Tan sólo con posicionar el mouse sobre el nombre de la campaña, podremos dilucidar el “id” particular de cada una.

Por lo tanto, a partir de conocer los “id” de cada campaña, el sistema de gestión a partir de su lógica de negocio, podrá evaluar cada llamada e indicar a OMniLeads hacia donde encaminarla devolviendo el par “X,Y”.

Custom asterisk dialplan

Resulta sumamente útil disponer de la posibilidad de poder forzar a que una llamada ejecute un “plan de discado” generado a imagen y semejanza de cualquier requerimiento puntual del modelo de negocios implicado.

Pero, ¿a qué nos referimos con “plan de discado”?

Como bien sabemos, OMniLeads utiliza [Asterisk](#) como pieza fundamental dentro del módulo de “gestión de llamadas telefónicas”, y por lo tanto, cualquier programador con conocimientos de sintaxis de “dialplan” podrá generar sus propias rutinas de tratamiento de llamadas, pudiendo a su vez invocarlas dentro del flujo de llamadas de:

- Rutas entrantes
- Inbound campaigns
- Failover destination
- Outbound routes
- Etc.

Por lo que se permite entonces, generar un “nodo” invocable dentro de una llamada procesada en OMniLeads, siendo éste “nodo”, lógica de programación de Asterisk, personalizada de acuerdo a cualquier necesidad puntual que esté por afuera del alcance de los módulos típicos del sistema.

For example, a developer can write a dialplan to integrate different business processes and numerous data sources to your IVR using API integrations with CRM or ERP.

Custom destination settings

Para crear un destino personalizado, se debe acceder al menú *Telefonía -> Destinos personalizados*.

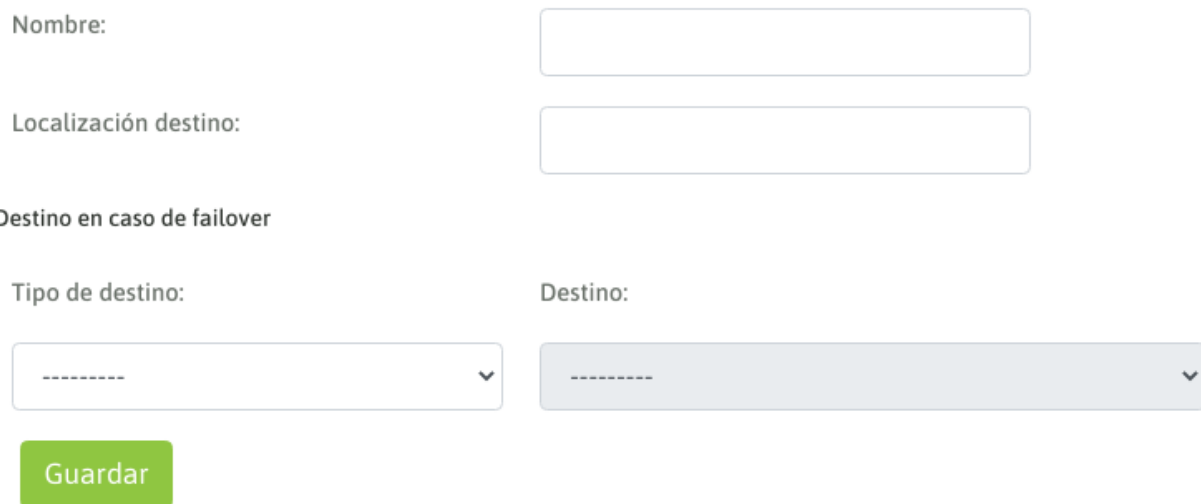
El módulo “destino personalizado” simplemente involucra un formulario sencillo donde se indica el nombre del nodo de “dialplan personalizado” y la tríada:

- Context
- Extension
- Priority

Además, contamos con la necesidad de indicar un destino en caso de fallo.

Todo esto, se visualiza en la siguiente figura:

Crear Destino Personalizado



Formulario de creación de destino personalizado. El formulario contiene los siguientes campos:

- Nombre:** Campo de texto.
- Localización destino:** Campo de texto.
- Destino en caso de failover:** Sección con dos campos de selección:
 - Tipo de destino:** Menú desplegable con una opción visible (-----).
 - Destino:** Menú desplegable con una opción visible (-----).
- Guardar:** Botón verde para guardar los datos.

Figura 1: Formulario de destino personalizado

Por otro lado, el programador podrá generar su código a nivel archivo de texto “*oml_extensions_custom.conf*”. Éste será cargado en tiempo real y también tenido en cuenta a la hora de generar los Backup&Restore de la plataforma.

Example

We will implement the dialplan

```
[omnileads_custom]
exten => s,1,Verbose(** Ejemplo de destino personalizado **)
same => n,Answer()
same => n,Playback(demo-congrats)
same => n,Hangup()
```

Then, we will write the code on *oml_extensions_custom.conf* the file is on “*/opt/omnileads/asterisk/etc/asterisk*” dir.

Luego, debemos generar el “nodo” destino personalizado, sobre la interfaz de configuración de OMniLeads:

Create Custom Destination

Name:

Custom destination:

Failover destination

Destination type:

Destination:

Figure 2: Ejemplo de destino personalizado

Finalmente, podemos invocar a nuestro nodo, desde una opción del IVR, validación horaria o ruta entrante:

Inbound route creation

Name:

DID number:

Caller id prefix:

Language:

Destination type:

Destination:

Figure 3: Ejemplo de destino personalizado

11.1.10 Campaign Templates

It is often recurring that the parameters of a “class” of campaigns (for example, survey preview campaigns) do not vary too much except perhaps by the assigned agent group, a contact database, or by assign supervisor. Therefore instead of having to create very similar campaigns always from scratch, you can generate templates and then quickly create new campaigns from cloning those templates.

This functionality is granted by the OMniLeads campaigns *Templates*.

Once you have generated a template (which is generated similarly to a campaign), you can create new campaigns simply by selecting the template and the *Create campaign from template* option. Each new campaign will be available with all the parameters specified in its parent template.


Name	Actions
survey_A	<div>Options ▾</div> <div><div>+ Create Campaign from Template</div><div> Delete</div></div>

Figure 15: Templates

11.1.11 Interaction with external CRMs

OMniLeads is designed from a perspective that prioritizes integration with the company’s preferred CRM/ERP system. Thus providing the possibility for a company to maintain the use of its CRM/ERP system appropriate to its vertical market (health, sales, customer service, etc.).

Through its own functionalities and API methods, OMniLeads allows the following interactions:

- Open a specific view of the CRM in an incoming or outgoing communication, using communication parameters (agent id, contact id, campaign id, etc.) as dynamic data to invoke the CRM.
- Execute *click to call* from a contact view in the CRM and thus activate a call through a campaign and OMniLeads agent.
- Record the management of a CRM contact and that the disposition impacts OMniLeads, so that there is a correlation between the CRM system and the Contact Center system within each campaign.
- Request the “ID” of the person on incoming calls, send this “ID” to the CRM (via http request) and wait for the latter to respond on which OMniLeads campaign to route a call.

These concepts and configurations are expanded in the following link [CRM integration](#).

METRICS, REPORTS, RECORDINGS AND SUPERVISION

Inside this chapter, all about the metrics, statistics, reports, recordings and real time supervision, etc.

12.1 Metrics, Recordings and Supervision

12.1.1 Recordings

All campaigns that operate with the call recording option enabled generate recording files with listening and download actions. The OMniLeads *Recordings* module allows to search for recordings using filters as search criteria.

Menu access

In order to access the recording search engine, go to *Recordings* Menu. The module view can be seen below, where a list of recordings is displayed according to a filter criteria like date and a set of parameters as:

- **Date:** allows to filter your search by date or specific date range.
- **Call Type:** allows to filter according to call type criteria (manual, incoming, preview, predictive dialer).
- **Phone Number:** this field allows to filter search engine by a specific telephone number.
- **Calle ID:** We can search for a call if we have its CallerID, making a quick and effective search.
- **Agent:** allows to filter search engine for calls managed by specific agent.
- **Campaign:** allows to filter search engine for calls managed by specific campaign.
- **External ID:** We can search all calls from a particular contact, with the `external_id` field defined in the contact database at the time of creation.
- **Minimum duration:** this field is used to limit search engine for Recordings with a specific value of duration, as minimum.
- **Checked:** this check is used to recover recordings of calls that have been observed/marked by agent during conversation.

- **Engaged Calls:** as we well know, in OMniLeads there are normal Call Dispositions and “Engaged Call Dispositions”. Last one triggers campaign forms. In this case, this check allows administrator/supervisor to recover dispositioned calls of type Engaged.
- **Recordings per page:** We define how many recordings we want to list per page.
- **Generate zip of recording files:** Once the recordings are listed, we can select all that the page allows and download them in ZIP format.

BUSCAR GRABACIÓN

Fecha:

Tipo de llamada:

Teléfono Cliente:

Call ID:

Agente:

Campaña:

---- Todas ----

ID de contacto externo:

Duración mínima:

En segundos

Calificada como gestión:

☐

Marcadas:

☐

Grabaciones por página:

10

Buscar

Generate recordings .zip file

<input type="checkbox"/>	Date	Call type	Phone number	Agent	Campaign	Recording	Call Disposition	Description
<input type="checkbox"/>	2021-06-14 19:05	Dialer call	6553131	Pedro Salvador	Prueba_dialer_nombre_largo_verificando	▶ 0:00 / 0:05 <div></div> 🔊 ⋮		Description
<input type="checkbox"/>	2021-06-14 19:02	Dialer call	6553132	Pedro Salvador	Prueba_dialer_nombre_largo_verificando	▶ 0:00 / 0:20 <div></div> 🔊 ⋮		Description
<input type="checkbox"/>	2021-06-14 19:02	Dialer call	6553133	Pedro Salvador	Prueba_dialer_nombre_largo_verificando	▶ 0:00 / 0:01 <div></div> 🔊 ⋮	No Contesta	Description
<input type="checkbox"/>	2021-06-14 18:59	Dialer call	6553134	Pedro Salvador	Prueba_dialer_nombre_largo_verificando	▶ 0:00 / 0:02 <div></div> 🔊 ⋮	Encuesta realizada	Description
<input type="checkbox"/>	2021-06-14 18:58	Dialer call	6553135	Pedro Salvador	Prueba_dialer_nombre_largo_verificando	▶ 0:00 / 0:03 <div></div> 🔊 ⋮	Encuesta realizada	Description
<input type="checkbox"/>	2021-06-14 18:17	Dialer call	6553137	Pedro Salvador	Prueba_dialer_nombre_largo_verificando	▶ 0:00 / 0:04 <div></div> 🔊 ⋮		Description

1

Figure 1: Recordings

12.1.2 Inbound Campaigns output

This section describes and analyzes all staff related to the “output” generated by incoming call campaigns.

Inbound Campaign Reports

This section covers all available reports for an Inbound Campaign.

INBOUND CAMPAIGNS

Active Campaigns

Name	Databases	Status	Actions
sales_1	survey_2503: (14 contactos)	Active	Options ▾
Sales_online_1	BD_contacto_308ae6c4-5d53-45ce-8611-3cb1fc85bede: (0 contactos)	Active	Options ▾
Sales_marketing_camp	BD_contacto_f298fd3d-1ea0-4235-aa38-2e2fcb5a34e8: (0 contactos)		<div> + Add Agents + Add Supervisors Call Dispositions Database Results Reports Delete </div>

Deleted Campaigns

Show Hidden

Figure 1: Cmpaign Reports View

General Report for Inbound Campaign

This report gives us a summary of several aspects of an Inbound Campaign. Metrics such as number of calls received (attended and not attended) and performed in the campaign are displayed in detail with respect to the agent call dispositions.

In order to access to this report, go to “Reports” option within the campaign.

The first information that comes up are the “received calls” and the “manual calls” within the campaign.

Remember that in OMniLeads an agent can process a manual call and associate it with an incoming campaign. For example, the agent can redial an incoming call number that has been accidentally hanged up, and handle it as a manual outbound for the campaign.

Also it give us information about average waiting and abandon time for call attempts to the campaign.

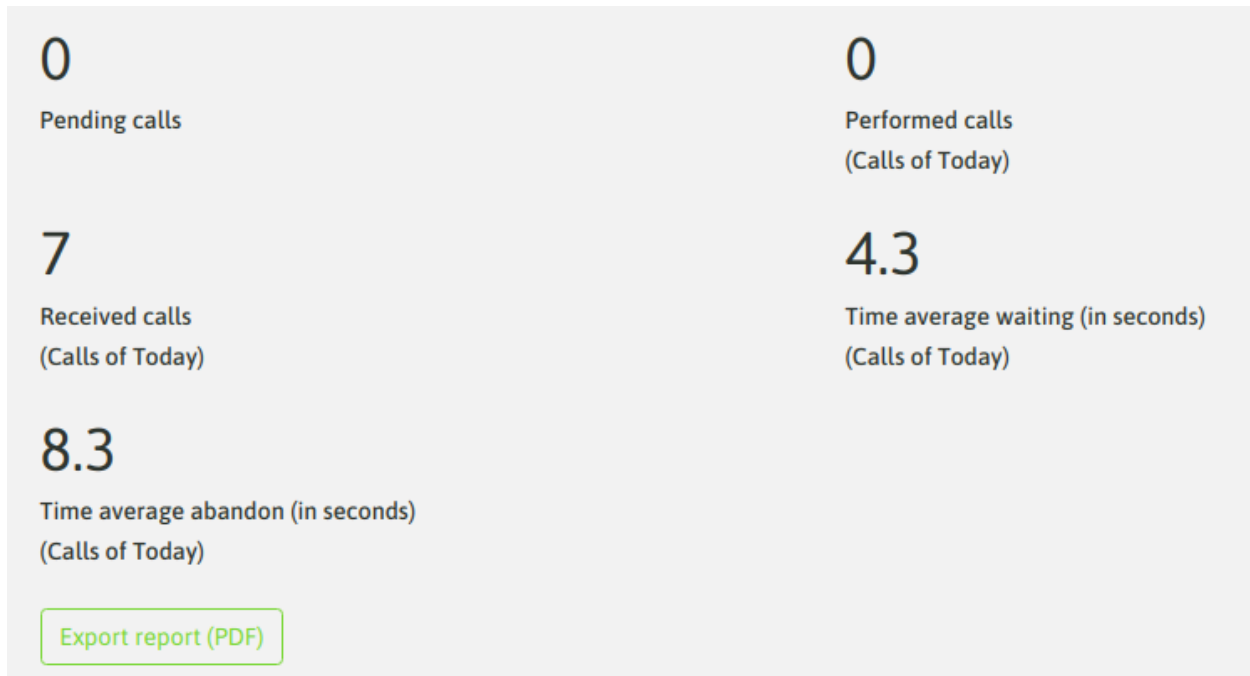


Figura 2: Llamadas entrantes / Llamadas realizadas

Following the tour of this screen, we will find the first button to export information to a CSV. In this case, the button allows us to export to “CSV/Spreadsheet” all calls answered in the campaign.

Telefono	nombre	apellido	dni	telefono2	telefono3	Fecha-Hora	Contacto	Tel status	Calificado	Observaciones	Agente	base de datos	Sale order	Deliv
4357142						2019/04/01 09:47:38	Fuera de base	Llamada Atendida sin calificacion			mikael akerfeld	Fuera de base		
4550870						2019/04/01 09:48:01	Contactado	Wrong number			mikael akerfeld	Fuera de base		
4773806	dario	argento	432432423			2019/04/01 09:48:39	Contactado	Sale			martin lopez	Fuera de base	322333333	10/04
4311649	peter	targtren	636666666			2019/04/01 09:49:37	Contactado	Sale			mikael akerfeld	Fuera de base	4444444444444	23/04

Figure 3: Attended Calls CSV

As it can be seen, the file presents as the first column, the Telephone from which the campaign was called. Although the contact can have more than one phone associated, in this report the column “Phone Number” refers to the phone that originated the communication to the campaign and to which the agent call disposition is associated (column “Call Disposition”).

It is not mandatory for incoming campaigns, to associate them to a contact database. When an incoming campaign is NOT associated to a contact database, all calls in reports are mentioned as “out of database” in the database column of our spreadsheet (figure3).

Below the first export button, a report table can be seen. It represents the different call dispositions made by agents for those answered calls.(figure 4).

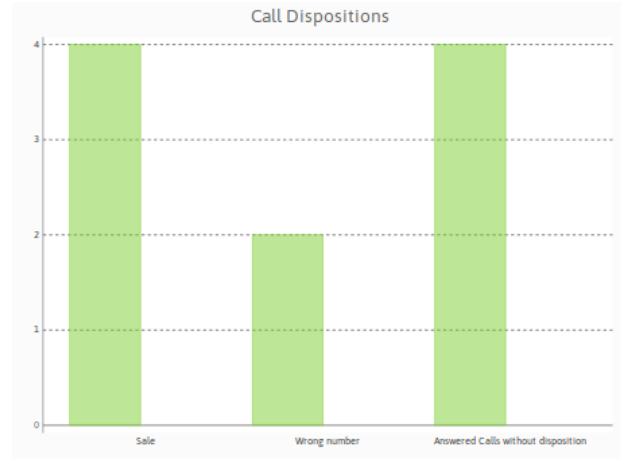
In this case and in general all tabulated information can be exported to CSV file.

Contacted Calls Report

[Export Contacted Calls Report \(CSV\)](#)

Calls connected to Agent

Call Disposition	Number
Sale	4
Wrong number	2
Answered Calls without disposition	4
Total assignments	10

[Export Call Dispositions Report \(CSV\)](#)

Call Dispositions Report

In the following section, a list of not answered calls can be obtained. These ones represent all the failed attempts from inbound point of view: abandoned calls, expired calls.

Amount of No Answered Calls

Result	Number
No Answered Calls	0
Cancelled	0
Answer Machine Detected	0
Busy	0
No Available Channels	0
Failed	0
Other	0
Blacklist	0
Abandoned Calls	1
Expired Calls	4
Total of Not attended calls	5

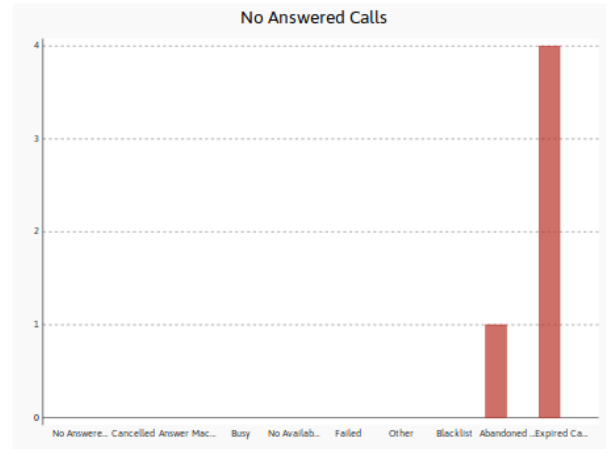
[Export Not attended Calls Report \(CSV\)](#)

Figure 5: Not Attended Calls

Below we can see an Agent Performance report. It counts all the call dispositions registered by agents. There is also

an agent link that redirects to a screen with more information about agent performance in that campaign.

Agent	Engaged Call Dispositions	Agenda	Wrong number	Sale	Dispositioned Contacts
martin lopez	2	0	2	2	4
mikael akerfeld	2	0	0	2	2
Totals	4				6

Figure 6: Pending Contacts / Terminated Calls

Then if you click on a specific agent, a new screen is displayed, which includes in depth the information of:

- Accumulated Time in the campaign
- Pause Time in the campaign
- On Call time in the campaign
- Amount of Processed Calls
- Average Call Time
- Amount of Failed attempts
- Effectiveness percentages

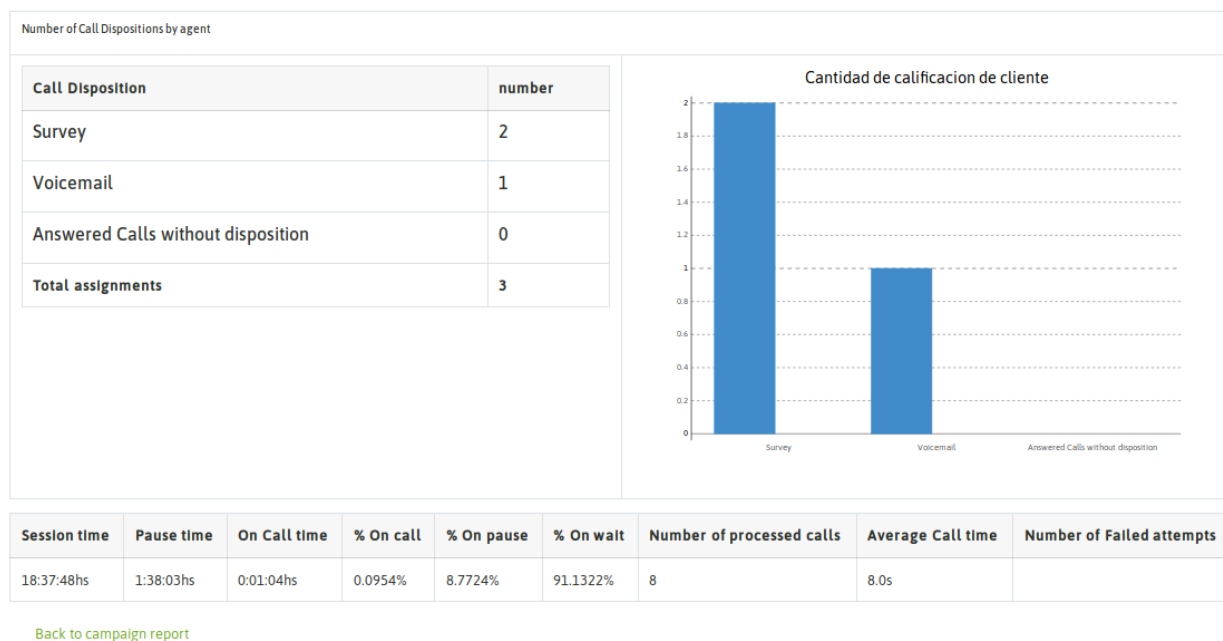


Figure 8: Agent Performance Detail

Returning to the campaign report, the last item is a List of all calls processed in the campaign and their results. It exposes separately all manual calls that have been made in the campaign. Remember that in any type of campaign, agent can generate manual calls on behalf of selected campaign.

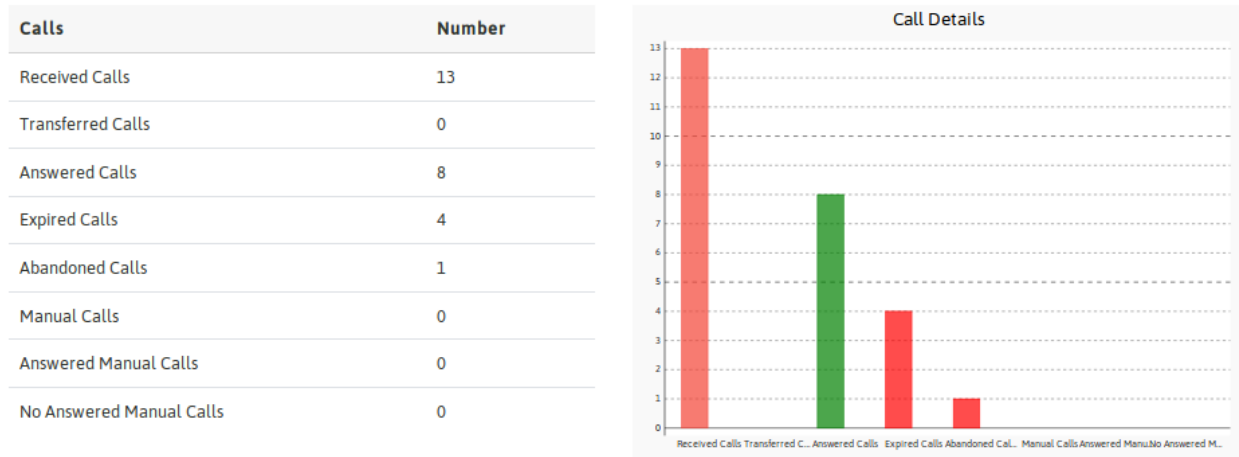


Figure 9: Calls Detail

Call Dispositions Report

This report lists all call dispositions made by an agent in the campaign. It can also be exported as a CSV file, containing full details for both “normal” and “engaged” call dispositions.

[Export Call Dispositions \(CSV\)](#)
[Export Engaged Call Dispositions \(CSV\)](#)

Phone Number	Data	Call Disposition	Comments	Databases
4317381	["thomas", "lang", "546654564", "2121222", "2121333"]	Wrong number		Out of database
4541962	["Devin", "Townsend", "4324234324", "", ""]	Sale		Out of database
4550870	["", "", "", "", ""]	Wrong number		Out of database
4773806	["dario", "argento", "432432423", "", ""]	Sale		Out of database
4311649	["peter", "targtren", "63666666", "", ""]	Sale		Out of database

Figure 10: Call Dispositions Detail

It is important to clarify that this list also includes those contacts dynamically registered in the campaign by agents. For example, contacts who call the campaign and are not part of a contact database.

Contact Database Results

Here is a flat list of all the contacts associated with the campaign database and the result of the last call made by the contact to the incoming campaign. The difference between this report and the previous one is that here we will NOT find those contacts introduced dynamically by agents (out of database). It is expected for this report to present a mapping between the contact database assigned to the campaign and the results once processed by the campaign.

Note: NOTE: It is important to note that this report is very useful in Preview and Predictive Dialer Campaigns, being perhaps not very irrelevant in incoming campaigns.

CONTACTED CALLS CAMPAIGN STATUSPRUEBA_DIALER_NOMBRE_LARGO_VERIFICANDO

Export Contacted Calls (CSV)

Phone Number	name	surname	id_contact	tel2	tel3	tel4	Call Disposition	Contacted Calls
6553124	nombre20	apellido20	434343240	55001134	66001135	77001136	No Contesta	
6553125	nombre21	apellido21	434343241	55001135	66001136	77001137	No Contesta	
6553133	nombre29	apellido29	434343249	55001143	66001144	77001145	No Contesta	
6553134	nombre30	apellido30	434343250	55001144	66001145	77001146	Encuesta realizada	
6553135	nombre31	apellido31	434343251	55001145	66001146	77001147	Encuesta realizada	
6553137	nombre33	apellido33	434343253	55001147	66001148	77001149	Encuesta realizada	
6553136	nombre32	apellido32	434343252	55001146	66001147	77001148		Ocupado
6553132	nombre28	apellido28	434343248	55001142	66001143	77001144		Contacted
6553131	nombre27	apellido27	434343247	55001141	66001142	77001143		Contacted
6553130	nombre26	apellido26	434343246	55001140	66001141	77001142		Canales Saturados
6553129	nombre25	apellido25	434343245	55001139	66001140	77001141		Canales Saturados

Contact Database Results

12.1.3 Outbound Campaigns output

This section describes and analyzes everything related to the “output” generated by outbound call campaigns.

Outbound Campaign Reports

This section covers all available reports for an Outbound Campaign.

```
preview_sales_2904
```

Active
Options

Completed Campaigns

Name	Status	Actions
There are no campaigns		

Deleted Campaigns

Edit
+ Add Agents
+ Add Supervisors
Call Dispositions
Database Results
Reports
Show assigned contacts
Delete

Figure 1: Campaign Reports View”

General Report for Outbound Campaign

This report gives us a summary of several aspects of the campaign. Number of pending calls that remains on campaign, total amount and details for all campaign calls (contacted and not contacted), etc.

In order to access this report, go to “Reports” option within the campaign.

The first information that comes up is the “Pending Contacts to Manage” Versus “Total Calls Made” in the campaign. These calls contemplate the fact that it is possible to dial a contact more than once. Therefore, it is expected that the number of calls made exceeds the amount of contacts available in the contact database of the campaign.

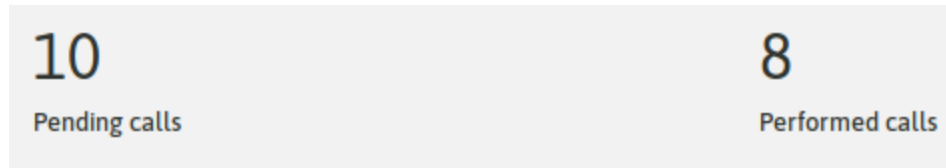


Figure 2: Pending Calls / Terminated Calls

Following the tour of this screen, we will meet with the first button to export information to CSV. In this case, the button allows us to export to “CSV/Spreadsheet” all contacted phones (answered calls) within the campaign with their call dispositions.

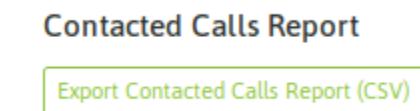


Figure 3: Contacted Calls

Information is presented as shown in Figure 4.

Phone Number	tel2	tel3	name	surname	id	Datetime Contact	Phone Number	Status	Dispositioned	Comments	Agent	database	Sale order	Delivery deadline	Product	Extra information
6149047	5149092	6149047	Herman	Hesse	1000005	2019/03/30 09:50:10	Contacted		Wrong number		mikael akerfeld	multinum-tests: (13 contactos)				
6149017	5149015	6149017	Timothy	Leary	1000002	2019/03/30 09:49:38	Contacted		Sale		mikael akerfeld	multinum-tests: (13 contactos)	321321	04/04/2019	A	
4149014	5149015	6149017	Timothy	Leary	1000002	2019/03/30 09:49:09	Contacted		Attended call without disposition		mikael akerfeld	multinum-tests: (13 contactos)				
4149052	5149090	6149090	Atahualpa	Yupanqui	1000010	2019/03/30 09:45:29	Contacted		No answer		mikael akerfeld	multinum-tests: (13 contactos)				
5149092	5149092	6149067	Jiddu	Krishnamurti	1000007	2019/03/30 09:41:05	Contacted		Sale		martin lopez	multinum-tests: (13 contactos)	432432	05/04/2019	C	
6149037	5149095	6149037	Aldous	Huxley	1000004	2019/03/30 09:39:06	Contacted		Voicemail		martin lopez	multinum-tests: (13 contactos)				
6149032	5149095	6149037	Aldous	Huxley	1000004	2019/03/30 09:38:33	Contacted		Attended call without disposition		martin lopez	multinum-tests: (13 contactos)				
6149027	5149025	6149027	Albert	Hoffman	1000003	2019/03/30 09:37:30	Contacted		Sale		martin lopez	multinum-tests: (13 contactos)	32323232	03/04/2019	A	

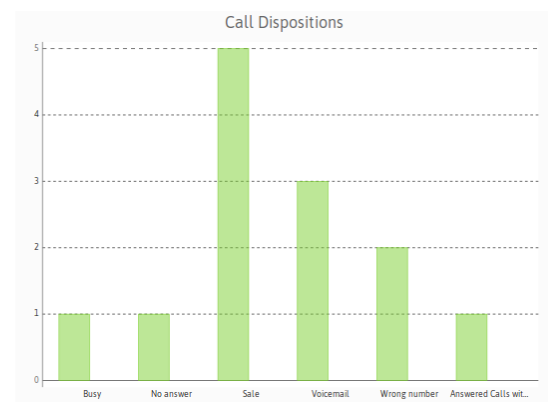
Figure 4: Contacted Calls CSV

As it can be seen, the file presents as the first column the contacted telephone number. Although the contact may have more than one associated phone, in this report the column “Phone number” references to the telephone number contacted by the campaign and to the associated call disposition (column “Call Disposition”).

The next report that is presented has to do with the number of call dispositions agents have made within the campaign (Figure 5).

In this case and in general, all tabulated information is able to be exported to CSV in order to have the data on a spreadsheet.

Call Disposition	Number
Busy	1
No answer	1
Sale	5
Voicemail	3
Wrong number	2
Answered Calls without disposition	1
Total assignments	13



[Export Call Dispositions Report \(CSV\)](#)

Figure 5: List of Call Dispositions

Following the next report sections, the system displays a list of all campaign calls that were not contacted: i.e. failed attempts according to what the system detects in response to the PSTN/SIP Provider.

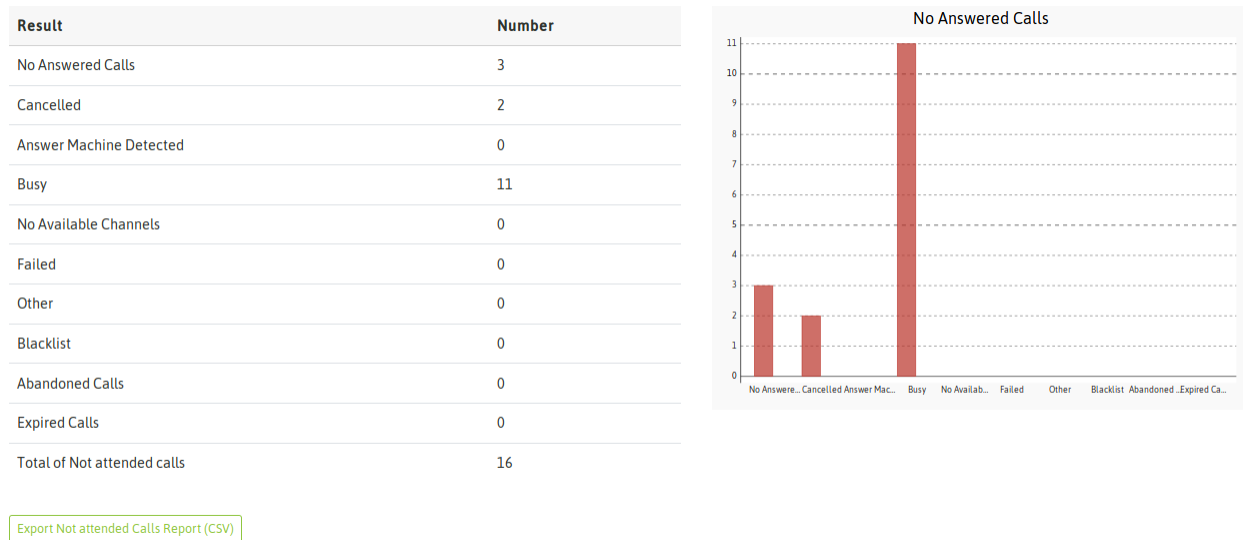


Figure 6: Not Attended Calls Detail

Below we can see an Agent Performance report. It counts all the call dispositions registered by agents. There is also an agent link that redirects to a screen with more information about agent performance in that campaign.

Agent	Engaged Call Dispositions	Agenda	Wrong number	Busy	No answer	Voicemail	Sale	Dispositioned Contacts
martin lopez	2	0	0	1	1	1	2	5
mikael akerfeld	3	0	2	0	0	2	3	7
Totals	5							12

*Figure 7: Agent Performance

Then if you click on a specific agent, a new screen is displayed, which includes in depth the information of:

- Accumulated Time in the campaign
- Pause Time in the campaign
- On Call time in the campaign
- Amount of Processed Calls
- Average Call Time
- Amount of Failed attempts
- Effectiveness percentages



[Back to campaign report](#)

Figure 8: Performance Agent Detail

Returning to the campaign report, the last item is a List of all calls processed in the campaign and their results. It exposes separately all manual calls that have been made in the campaign. Remember that in any type of campaign, agent can generate manual calls on behalf of selected campaign.

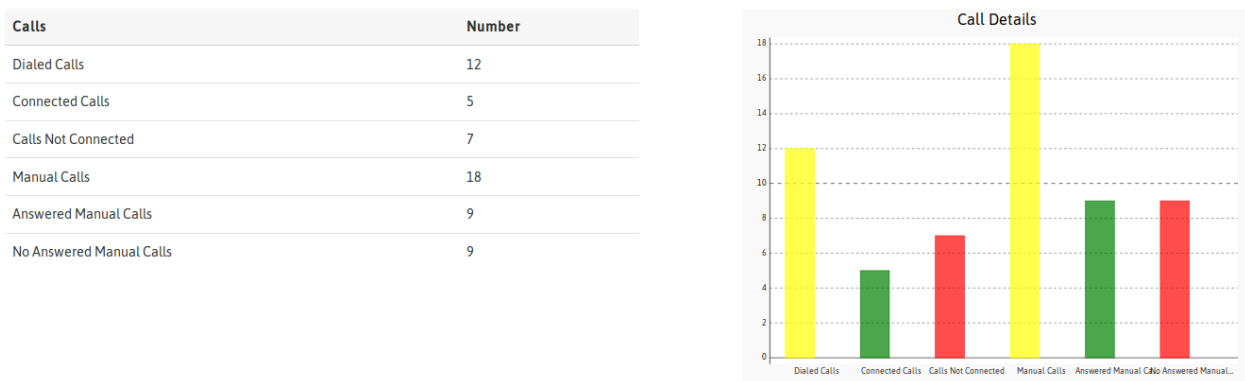


Figure 9: Total Calls

Call Dispositions Report

This report lists all call dispositions made by an agent in the campaign. It can also be exported as a CSV file, containing full details for both “normal” and “engaged” call dispositions.

Export Call Dispositions (CSV)		Export Engaged Call Dispositions (CSV)		
Phone Number	Data	Call Disposition	Comments	Databases
4149055	["5149095", "6149000", "Carl Gustav", "Jung", "1000011"]	No answer		multinum-tests: (13 contactos)
4149050	["5149090", "6149010", "Isaac", "Luria", "1000012"]	Busy		multinum-tests: (13 contactos)
4149020	["5149025", "6149027", "Albert", "Hoffman", "1000003"]	Sale		multinum-tests: (13 contactos)
4149032	["5149095", "6149037", "Aldous", "Huxley", "1000004"]	Voicemail		multinum-tests: (13 contactos)
4149050	["5149092", "6149067", "Jiddu", "Krishnamurti", "1000007"]	Sale		multinum-tests: (13 contactos)
4149052	["5149090", "6149090", "Atahualpa", "Yupanqui", "1000010"]	Voicemail		multinum-tests: (13 contactos)
4149014	["5149015", "6149017", "Timothy", "Leary", "1000002"]	Sale		multinum-tests: (13 contactos)
4149040	["5149092", "6149047", "Herman", "Hesse", "1000005"]	Wrong number		multinum-tests: (13 contactos)
4149052	["5149092", "6149055", "Jorge Luis", "Borges", "1000006"]	Wrong number		multinum-tests: (13 contactos)
4149050	["5149090", "6149087", "Facundo", "Cabral", "1000009"]	Voicemail		multinum-tests: (13 contactos)
4149057	["5149095", "6149026", "Robert Anton", "Wilson", "1000013"]	Sale		multinum-tests: (13 contactos)
4149090	["5149095", "6149097", "Terence", "McKenna", "1000001"]	Sale		multinum-tests: (13 contactos)

Figure 10: Call Dispositions Detail

It is important to clarify that this list also includes those contacts dynamically registered in the campaign by agents. For example, out-of-base contacts who were referred by another contact within the campaign.

By displaying any call dispositioned contact, we will see the complete form associated with that disposition:

CAMPAIGN PRUEBA_DIALER_NOMBRE_LARGO_VERIFICANDO




From: 14/06/2021

To: 22/06/2021

Database: nueva_base_de_contactos_nombre_largo: (47 contactos)

Export Call Dispositions (CSV)

Export Engaged Call Dispositions (CSV)

Phone Number	Data	Call Disposition	Comments
6553124	[{"nombre20", "apellido20", "434343240", "55001134", "66001135", "77001136"}]	No Contesta	
6553125	[{"nombre21", "apellido21", "434343241", "55001135", "66001136", "77001137"}]	No Contesta	
6553133	[{"nombre29", "apellido29", "434343249", "55001143", "66001144", "77001145"}]	No Contesta	
<div> 6553134</div>	[{"nombre30", "apellido30", "434343250", "55001144", "66001145", "77001146"}]	Encuesta realizada	
Disposition history			
Date-Time	Call Disposition	Comments	Engaged Call Dispositions
2021/06/14 15:59:44	Encuesta realizada	[{"calidad de atencion": "1", "tiempo de respuesta": "3", "recomienda nuestro servicio tecnico": "si", "Comentarios": "prueba"}]	
<div> 6553135</div>	[{"nombre31", "apellido31", "434343251", "55001145", "66001146", "77001147"}]	Encuesta realizada	
<div> 6553137</div>	[{"nombre33", "apellido33", "434343253", "55001147", "66001148", "77001149"}]	Encuesta realizada	

This report is accessible from any status of the campaign.

Contact Database Results

Here is a flat list of all the contacts associated with the campaign database and the result of the last call made to the contact. The difference between this report and the previous one is that here we will NOT find those contacts

introduced dynamically by agents (out of database). It is expected for this report to present a mapping between the contact database assigned to the campaign and the results once processed by the campaign.

CONTACTED CALLS CAMPAIGN STATUSPRUEBA_DIALER_NOMBRE_LARGO_VERIFICANDO

Export Contacted Calls (CSV)

Phone Number	name	surname	id_contact	tel2	tel3	tel4	Call Disposition	Contacted Calls
6553124	nombre20	apellido20	434343240	55001134	66001135	77001136	No Contesta	
6553125	nombre21	apellido21	434343241	55001135	66001136	77001137	No Contesta	
6553133	nombre29	apellido29	434343249	55001143	66001144	77001145	No Contesta	
6553134	nombre30	apellido30	434343250	55001144	66001145	77001146	Encuesta realizada	
6553135	nombre31	apellido31	434343251	55001145	66001146	77001147	Encuesta realizada	
6553137	nombre33	apellido33	434343253	55001147	66001148	77001149	Encuesta realizada	
6553136	nombre32	apellido32	434343252	55001146	66001147	77001148		Ocupado
6553132	nombre28	apellido28	434343248	55001142	66001143	77001144		Contacted
6553131	nombre27	apellido27	434343247	55001141	66001142	77001143		Contacted
6553130	nombre26	apellido26	434343246	55001140	66001141	77001142		Canales Saturados
6553129	nombre25	apellido25	434343245	55001139	66001140	77001141		Canales Saturados

Contact Database Results

12.1.4 General Call Reports

This section reviews each report generated by general call reports. *Reports - Calls*.

Calls General Report

The report can be run as in all cases, based on a date or date range. It displays information about all calls managed by the platform according to the filter selected.

The first report has to do with the total amount of calls transacted by the platform, classified in:

- Manual Outbound Calls
- Preview Outbound Calls
- Predictive Outbound Calls
- Inbound Calls
- Calls transferred to campaigns.

49

Processed calls

Dialer outbound calls	23	Inbound calls	6
Attended	19	Attended	0
Not attended	4	Expired	0
Lost Calls	1	Abandon	6
		Abandoned during welcome audio	0
Group by date		Group by date	
Manual Outbound calls	12	Preview outbound calls	8
Connected	11	Connected	7
Not Connected	1	Not Connected	1
Group by date		Group by date	
Transferred Calls to Campaigns	0		
Connected	0		
Not Connected	0		
Export total calls (CSV)		Export to CSV all reports	

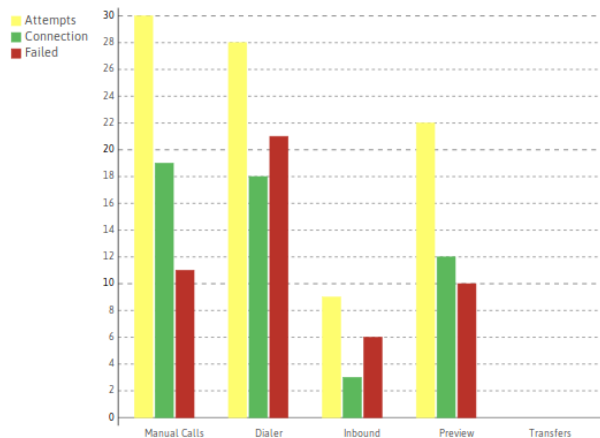
Figure 1: Call Types

The information is also detailed in terms of whether the calls were connected or not “connected”.

Then there are a couple of graphs that represent the statistics above mentioned.

Totals by call types

Totals by call types and completion type



Calls percentage by call type

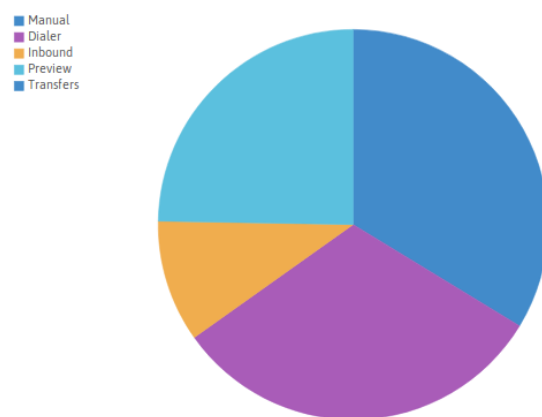


Figure 2: Types Call Graph

The following information is presented as a table in which all calls transacted by the platform can be divided in terms of the campaigns where they were processed.

Number of Calls by Campaign

Survey_preview_1	0	0	Preview
Survey_preview_2	3	2	Preview
Survey_Dialer_1	14	0	Dialer
Sales_Dialer	14	0	Dialer
Sales_online_1	1	0	Inbound
Sales_marketing_camp	3	0	Inbound
prev_multinum	8	5	Preview
prev_multinum_2	11	5	Preview
preview_sales_2904	30	18	Preview

Figure 3: Calls and Campaigns

In the following three sections, it is presented in a tabular and graphic way a breakdown of all calls processed by OMniLeads in terms of its nature (incoming, manual, preview and predictive). Then, for each type of campaign, all call information can be seen according to each type of call.

This way you can quickly deduce the effectiveness of each campaign analyzing tabular and graphical information.

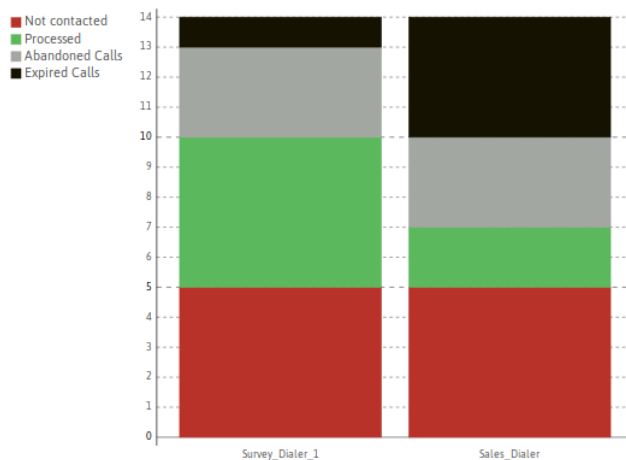
- Predictive Calls

Calls type distribution in dialer outbound campaigns

Full information 

Campaign	Performed calls	Connection	Manual Calls
Survey_Dialer_1	14	0:00:03	0
Sales_Dialer	14	0:00:02	0

Call types by dialer campaign



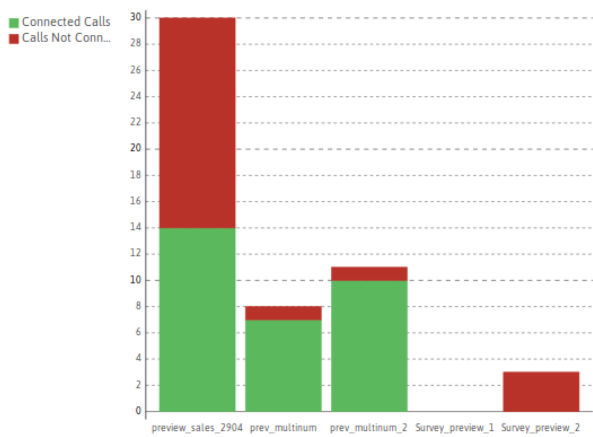
Export call types by Dialer Campaign (CSV)

Predictive Calls

- Preview Calls

Name	Performed calls	Connection	Manual Calls	Connection
preview_sales_2904	30	0:00:04	18	0:00:05
prev_multinum	8	0:00:04	5	0:00:05
prev_multinum_2	11	0:00:07	5	0:00:05
Survey_preview_1	0	0:00:00	0	0:00:00
Survey_preview_2	3	0:00:18	2	0:00:17

Call types by preview campaign



Preview Calls

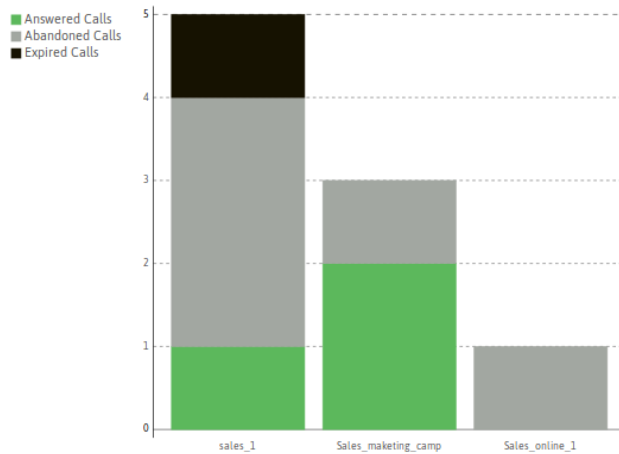
- Inbound Calls

Call types distribution in inbound campaigns

Full information ▾

Name	Received Calls	Wait	Manual Calls
sales_1	5	0:00:00	0
Sales_marketing_camp	3	0:00:00	0
Sales_online_1	1	0:00:00	0

Call types by inbound campaigns



Inbound Calls

12.1.5 Agent Reports

This section reviews each report generated by the general report of agent activity. *Reports - Agents*

Agent Reports

When accessing *Reports -> Agents* menu, a filter engine allows to select the agent or group of agents, and the date or range of dates. After clicking in Search button, agent activity report is displayed.

Date:

30/03/2019 - 30/03/2019

Agent:

mikael akerfeld
martin lopez

All agents:

☒

Agents group:

Search

Figure 1: Date/time and Agent Filters

The first information that can be seen is a table with the summary of all the agent activity.

Agent	Session time	% On call	Processed calls	Average time	Failed attempts
mikael akerfeld show by dates	2:23:14hs 0:42:48hs pause, 0:01:08hs call	0.7912% 29.8844% pause, 69.3244% waiting	13	5.23s	7
martin lopez show by dates	10:51:42hs 6:17:02hs pause, 0:01:12hs call	0.1841% 57.8557% pause, 41.9602% waiting	10	7.2s	13

Figure 2: Agent Activity Summary

From our summary we can break down plenty of details as for example the accumulated time in pause. There you will have a detailed report of the pauses and the time each agent took.

Agent	Pause	Type	Time
martin lopez show by dates	Coaching	Productive	0:00:46hs
martin lopez show by dates	ACW	Productive	6:17:02hs
martin lopez show by dates	Break	Recreational	0:01:52hs
martin lopez show by dates	Sales	Productive	0:00:34hs
mikael akerfeld show by dates	Coaching	Productive	0:02:15hs
mikael akerfeld show by dates	ACW	Productive	0:42:48hs
mikael akerfeld show by dates	Break	Recreational	0:00:23hs

Figure 3: Pauses Detail

The next parameter has to do with the amount of calls processed by each agent in terms of campaigns. That is to say, it presents a detail by agent of how many calls and how much time agent spent, per campaign.

Agent	Queue	Calls time	Processed calls
martin lopez	Sales_maketing_camp	0:00:15hs	1
martin lopez	prev_multinum_2	0:00:40hs	5
martin lopez	preview_sales_2904	0:00:17hs	4
mikael akerfeld	Sales_maketing_camp	0:00:06hs	1
mikael akerfeld	prev_multinum_2	0:00:07hs	2
mikael akerfeld	preview_sales_2904	0:00:55hs	10

Figure 4: Calls and Campaigns

Finally, a last table is presented detailing the amount of calls per type (manual, preview, dialer and incoming).

Agent	Total	PREVIEW	DIALER	INBOUND	MANUAL	TRANSFERRED
martin lopez	10	3	0	1	6	0
mikael akerfeld	13	6	0	1	6	0

Figure 5: Call Types

12.1.6 Realtime Supervision

This section reviews the OMniLeads Supervision module.

Supervision

This module allows Supervisors to view the status of Inbound Campaigns, Outbound Campaigns (Manual, Dialer and Preview) and Agents.

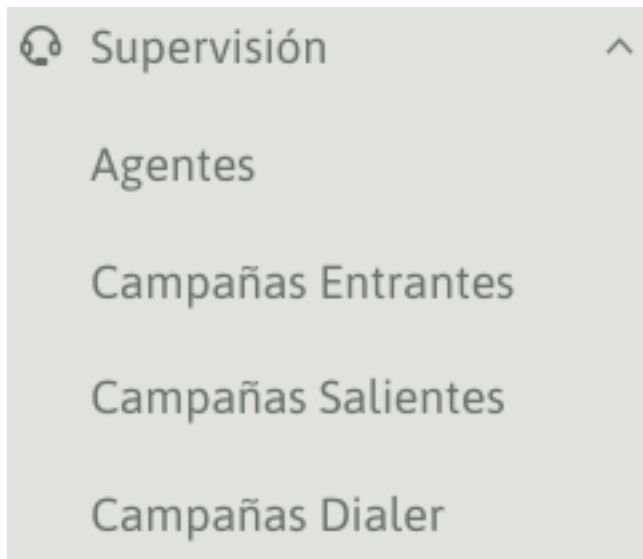


Figure 1: Supervision Possibilities

Agent Status View

In Agent section, it is possible to appreciate all the Active Agents operating on the system and the status they are in (Ready, On Call, Paused, Dialing, Offline).









































Important:

- Agents must be assigned to at least one campaign in order to appear in this supervision view.
- When an agent logs out of the system, she/he changes the active status to “Offline”, remaining only a few seconds in that transition before disappearing from the list of agents.
- When an agent enters the “Unavailable” state, it means that the agent losts connection or closes browser without logging out correctly.

Supervisor registrado
● Registrado

Filtrar grupo Filtrar Campaña Finalizar

Mostrar 10 entradas Buscar:

Nombre	Campaña	Contacto	Status	Tiempo	Acciones
Agustin Nicolas	13_CLASICA	6141644	PAUSE-ACW	00:00:20	   
Aldana Magali	16_CLASICA	8468446	ONCALL	00:00:40	   
Ana Belen	13_CLASICA	6387679	PAUSE-Carga_de_Venta	00:11:19	   
Angel Esteban	13_CLASICA	0821291	PAUSE-ACW	00:00:01	   
Dalana Denise	17_CLASICA_C	4286923	PAUSE-ACW	00:00:08	   
Eliana Mariel			UNAVAILABLE	00:01:54	   
Elisa Del Rosario			READY	00:00:03	   
Florencia Guadalupe	13_CLASICA	1104491	ONCALL	00:06:29	   
Gabriel Isaac			PAUSE-Coaching	00:22:01	   
Gabriel Nicolas	13_CLASICA	4973855	ONCALL	00:00:11	   

Mostrando 1 a 10 de 31 entradas Anterior 1 2 3 4 Siguiente

Figure 2: Agent Supervision View

Supervisors are able to take some actions on each agent. That is the reason why action buttons appear next to the agent status. All possible functions will be explained further below (from left to right).

- **Spy:** the supervisor listens the *active call* between agent and customer. Clicking on *Finish* button, allows the supervisor to finish the listening action.
- **Spy and Whisper:** the supervisor is able to listen and speak during an agent conversation without customer perception. Clicking on *Finish* button, allows supervisor to finish listening and whispering actions.
- **Pause Agent:** assuming the agent is out of the box and forgot to pause the session, the supervisor is able to force a pause action from supervisor view. This is useful as to avoid receiving calls while agent is absent but ready. Supervisor can also un-pause session by pressing the same button again.
- **Logging Out Agent:** assuming the agent closes browser session incorrectly (without using Log-Out button), agent session still remains active. Supervisor has the ability to log agent out with this button. It is important to perform this action as to avoid having inconsistencies in session time reports (and more important to instruct agent to close session from log-out button!).

Note: The supervisor has a small webphone. In order to perform all the possible actions stated above, a message of **Supervisor Registered** needs to be present on supervisor console.

Inbound Campaigns View

This view exposes a summary of all productive Inbound Campaigns, in terms of the accumulated results of the operation day: received calls, answered calls, abandoned calls, abandoned during welcome audio calls, average waiting time, expired calls, on queue calls, average abandon time, and Engaged CallsDispositions (*) per campaign.

CAMPAÑAS ENTRANTES

Nombre	Llamadas	Atendidas	Abandonadas	Abandonadas en anuncio	T. promedio espera	Expiradas	En espera	T. promedio abandono	Gestiones
Entrante-1	7	4	3	0	4.3 segundos	0	0	8.3 segundos	0
sales_1	2	2	0	0	11.5 segundos	0	0	0.0 segundos	0
Sales_online_1	4	0	4	0	0.0 segundos	0	0	6.3 segundos	0
Sales_marketing_camp	1	0	1	0	0.0 segundos	0	0	11.0 segundos	0

Inbound Campaigns View

Outbound Campaigns View

As in the previous point, Outbound Campaigns also count on with a real time summary of the results for each campaign: dialed calls, answered calls, not answered calls, and their respective Engaged Call Dispositions (*).

CAMPAÑAS SALIENTES

Nombre	Llamadas discadas	Atendidas	No atendidas	Gestiones
testmanual	2	2	0	0
testdialer	0	0	0	0
testpreview	0	0	0	0

Outbound Campaigns View

Outbound Campaigns View

Dialer campaigns also have an updated summary in real time. On the one hand, the disposition of agents, online agents, on call agents, on hold agents. And on the other hand, the call arrangement, dialed, answered, not answered, answering machines, dialing channels, connected calls, lost calls, call dispositions and pending calls.

CAMPAÑAS DIALER

Mostrar entradasBuscar:

Agentes			Llamadas								
Nombre	Agentes Online	Agentes en llamada	Agentes en pausa	Llamadas discadas	Atendidas	No atendidas	Contestadores	Canales discando	Llamadas conectadas perdidas	Gestiones	Llamadas pendientes
Campaña 1	82	26	31	6339	2126	405	3551	138	3649	1763	56074
Campaña 2	82	26	31	4024	785	442	2652	55	2704	719	12654
Campaña 3	82	26	31	1219	133	90	969	13	977	102	2388
Campaña 4	2	0	2	0	0	0	0	0	0	0	0
Campaña 5	1	1	0	0	0	0	0	0	0	0	0

Figure 5: Dialer Campaigns Supervision View

Note: It is expected for this view to display results on a day-by-day basis, from 00:00 to 23:59. After that time range, statistics are reset.

Note: Engaged Call means a call that the agent closes with an engaged call disposition, the one in charged of triggering a campaign form.

Every time an agent generates an engage with a contact, there is the possibility of auditing it from the backoffice module.

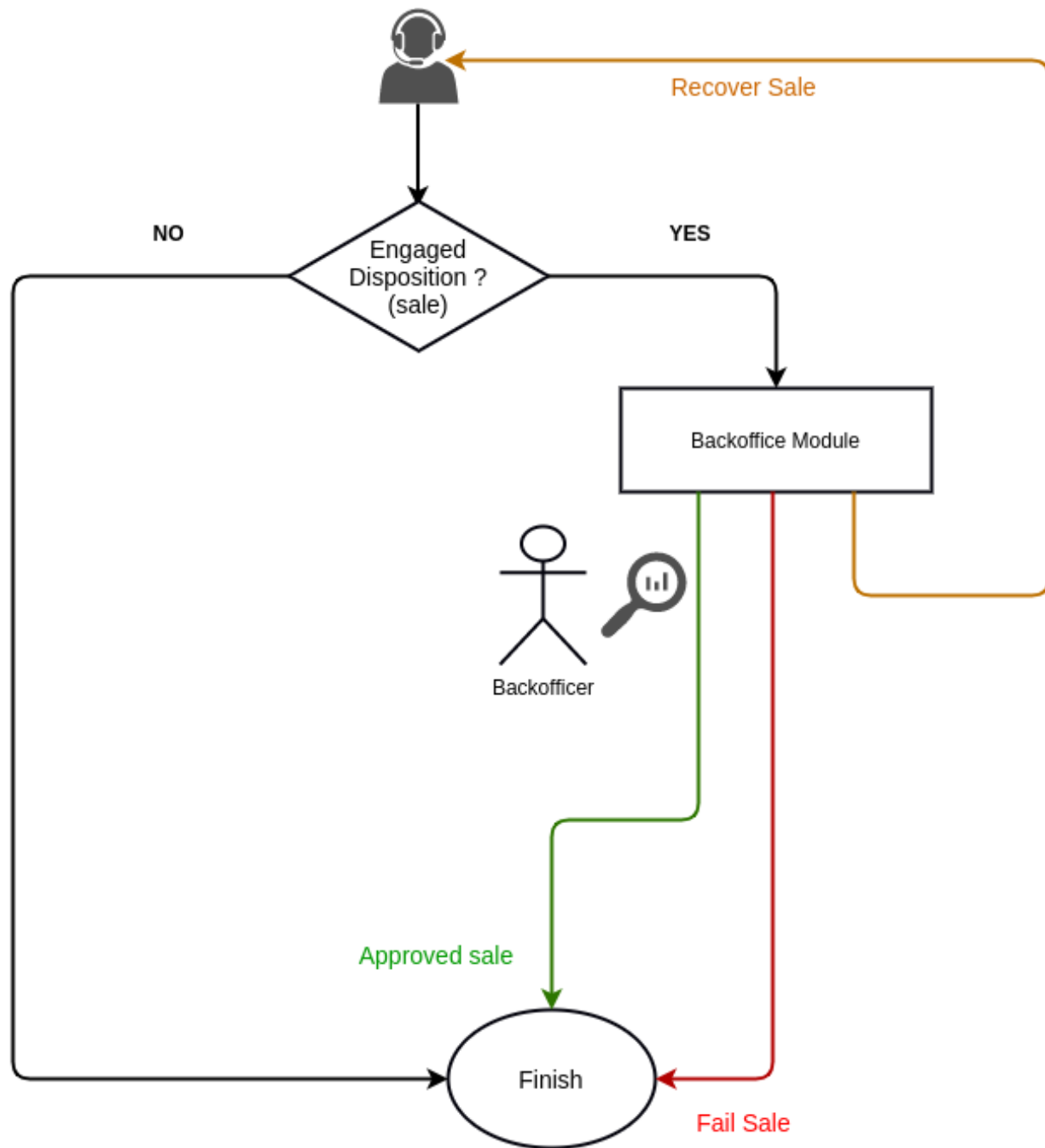
13.1 Backoffice module

This module allows to define system users assigned to audit each engage generated by an agent in each campaign. In Contact Center operations the workflow commonly involves, on the one hand, Agents and Supervisors coordinating the communications management operation and, on the other hand, the auditing or backoffice sector controlling that each record qualified as Positive Management (or sale) has been correctly implemented by the agent.

OMniLeads facilitates a module in which the auditors are able to list all the management procedures performed by the Agents, to then inspect in order to validate one by one the authenticity of those managements based on the parameters of the operations. The auditors have the possibility to display the contact information, the form completed by the agent and the recordings related to management.

Flow diagram

The following diagram is intended to illustrate the operation of the module.



The first thing that needs to be cleared is that only the qualified registers with a management rating (Campaigns, Ratings and Forms) will be sent to the Audits office. There is where auditors will then work on each pending register.

Date:	Agent:
<input type="text"/>	<input type="text"/>
Campaign:	Agents Group:
<input type="text"/>	<input type="text"/>
ID de contacto externo:	Contact ID:
<input type="text"/>	<input type="text"/>
Phone Number:	Call ID:
<input type="text"/>	<input type="text"/>
Audit status:	
<input type="text"/>	

Date/Time	Agent	Contact ID	Phone Number	Status	Details
2020-06-18 11:01	Adrian Belew	48	6553135	Pending	Audit Call Disposition
2020-06-18 11:01	Adrian Belew	49	6553136	Pending	Audit Call Disposition
2020-06-18 11:00	Adrian Belew	50	6553137	Pending	Audit Call Disposition

On each register the auditor will be able to enter in order to analyze the engage.

Audit Call Disposition

[Ver datos de Contacto](#)

Call Disposition	Date	Comments	Recording
Success	June 18, 2020, 11 a.m.		<div>▶ 0:00 / 0:14</div> <div>⏮ ⏪ ⏩ ⏭</div>

Resultado:

⌵

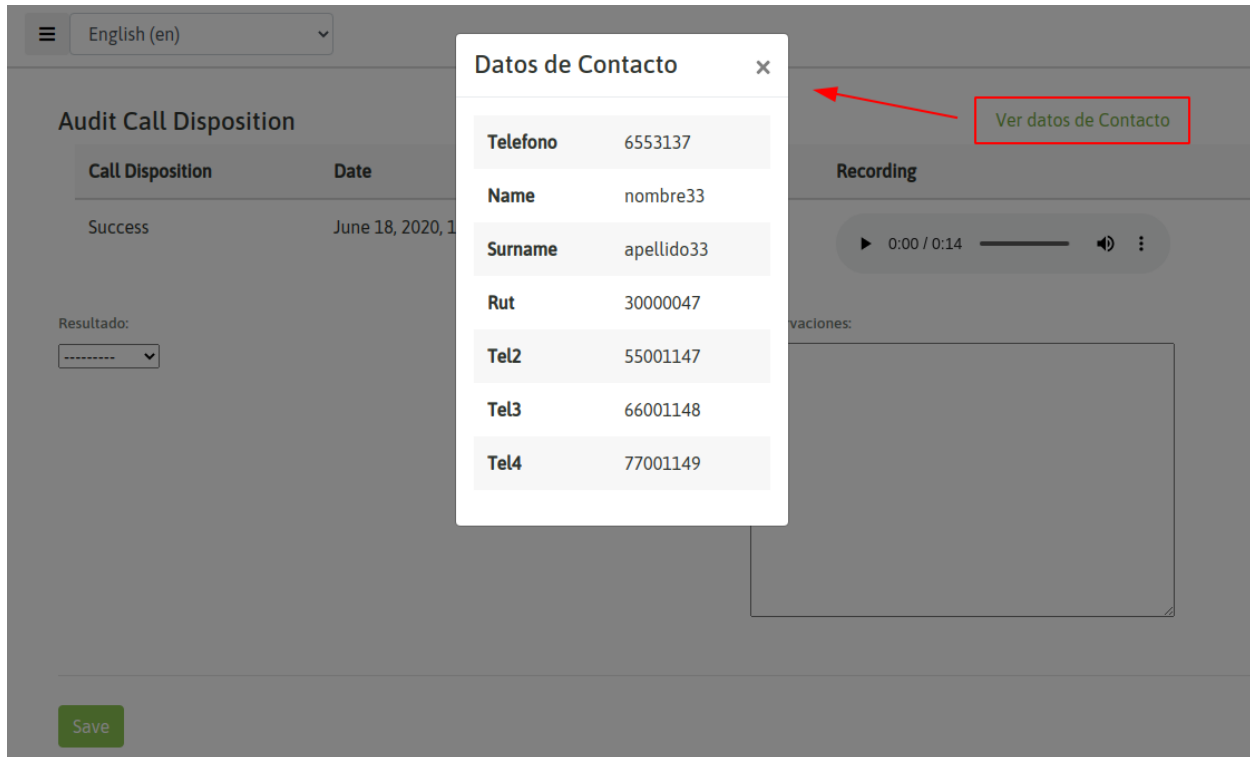
Approved

Rejected

Observed

Observaciones:

In the detailed view, the recordings inherent to the management can be reproduced. Also, both the contact data and the data entered in the management form can be displayed.



Audited records can be classified as:

- **Approved management:** after analyzing recordings, contact details and fields completed in the management form, the auditor verifies that the operation has been successful.
- **Rejected management:** after analyzing recordings, contact data and fields completed in the management form, the auditor detects that the operation does not meet the requirements to be approved.
- **Observed management:** after analyzing the recordings, contact data and fields completed in the management form, the auditor detects that the operation has been successful but sends it back to the agent (with the pertinent explanation) so that he or she can get in touch with the contact once more.

Search					
Date/Time	Agent	Contact ID	Phone Number	Status	Details
2020-06-18 11:01	Adrian Belew	48	6553135	Approved	Audit Call Disposition
2020-06-18 11:01	Adrian Belew	49	6553136	Rejected	Audit Call Disposition
2020-06-18 11:00	Adrian Belew	50	6553137	Observed	Audit Call Disposition

Observed management behavior

As discussed, observed managements are procedures that the agent must perform again, seeking to solve the observations left by the auditor in order to obtain subsequent approval. All agents have access to the auditors qualifications by entering the Ratings view displayed in the agent screen. Once an observation is noticed, the agent is able to read the description of the inconvenience left by the auditor.

Once the agent performs the management again, the auditor will be able to inspect and finally decide whether to approve, reject or observe again.

Search filters

Search Engaged Call Dispositions

Search

Date:

Agent:

Campaign:

Agents Group:

ID de contacto externo:

Contact ID:

Phone Number:

Call ID:

Audit status:

The module has the typical search filters (by agent, campaign, date, etc.). However, it is important to highlight the fact that it is also possible to search records by contact database ID. In other words, to use the unique identifier of the contact within the Base uploaded to the system. This option requires the use of the external contact ID field.

Important: In order to apply the filter option using the mentioned field, it is necessary to indicate the corresponding field at the time of uploading the base.

telefono	name	surname	RUT	tel2	tel3	tel4
4553101	Terence	McKenna	30000001	55001101	66001102	77001103
4553102	Timothy	Leary	30000002	55001102	66001103	77001104

Campos de teléfono:

- ☒ telefono
- ☐ name
- ☐ surname
- ☐ RUT
- ☒ tel2
- ☒ tel3
- ☒ tel4

Id externo:

RUT

[Volver a nueva base de datos](#)

[Guardar Base Datos](#)

In this chapter is covered all the actions that an OMniLeads agent can do inside an operation. Referring to receive/make of calls, dispositions, scheduling, call transfers, and much more.

14.1 Agent Manual

We will split Manual Agent in the following topics stated below.

14.1.1 Agent Session

All related to agent session is decribed in this piece.

Login

Access the system using OMniLeads URL from a browser.

If there are no commercial certificates installed, a security exception takes place when trying to access for the first time. This is because the presence of a self-signed certificate under https protocol.

In this case, just accept the exception and continue, as shown in figure 1 and 2.



Your connection is not private

Attackers might be trying to steal your information from **omniapp** (for example, passwords, messages, or credit cards). [Learn more](#)

NET::ERR_CERT_AUTHORITY_INVALID

☐ Help improve Safe Browsing by sending some [system information and page content](#) to Google.
[Privacy policy](#)

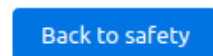
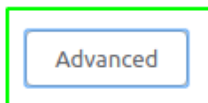


Figure 1: Not Trusted Certificate



Your connection is not private

Attackers might be trying to steal your information from **omniapp** (for example, passwords, messages, or credit cards). [Learn more](#)

NET::ERR_CERT_AUTHORITY_INVALID

☐ Help improve Safe Browsing by sending some [system information and page content](#) to Google.
[Privacy policy](#)

Hide advanced


Back to safety

This server could not prove that it is **omniapp**; its security certificate is not trusted by your computer's operating system. This may be caused by a misconfiguration or an attacker intercepting your connection.

[Proceed to omniapp \(unsafe\)](#)

Figure 2: Not Trusted Certificate

After accepting security exception, agent is able to access using username and password.



OMniLeads
 The Open Source Contact Center Solution

User

makerfeld


Password

.....

Log In

Figure 3: Login Screen

After successful login, agent screen is shown. The first thing that comes up is a browser pop-up screen telling us that OML needs access to microphone. As expected, this action needs to be allowed in order for the agent to use those resources for operating.



OMniLeads
 The Open Source Contact Center Solution

mikael akerfeld

Operación: 00:00:15 Pausa: 00:00:00

 Conectado
 Pausado
Reaundar

AGENDAS

Buscar

Filtrar

Fecha	Hora	Es personal	Es smart	Medio de comunicación	Descripción
No existen eventos para el día de hoy					

Teléfono

Registrando...
● Registrado

00:00:00

Transfer
Cancel Transfer
Hold

campaña:

Figure 4: Microphone Acces Permission

Once this is accepted, agent is able to login successfully and can listen a brief login message over the headsets. Agent will see the Webphone, and if all is going on a Registered status will be displayed as it is shown in figure 5.

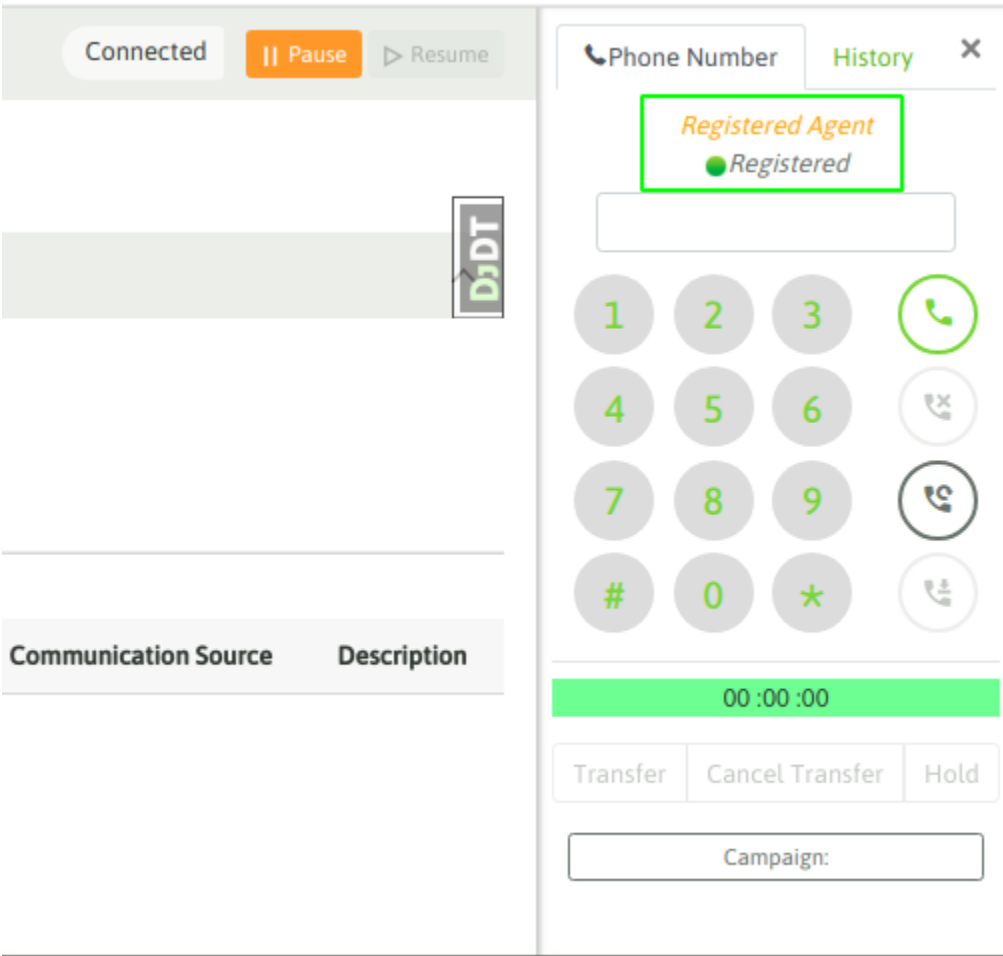
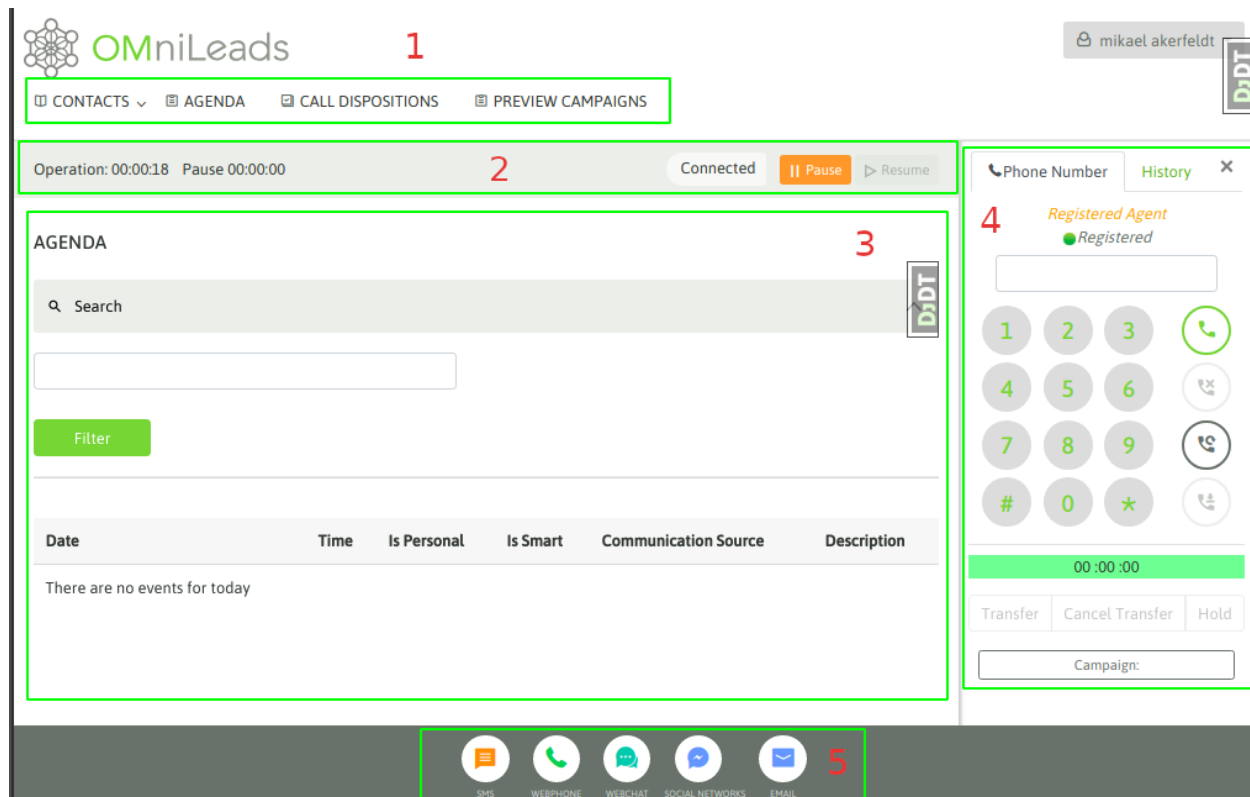


Figure 5: Agent registered in the system

Agent Console

Agent console is the component where all call management flows. In the following picture, we can see the different sections that will be explained below.



Agent Console

Dashboard de agente

When agents logged into the system will see a dashboard view that shows the following statistics of its current day:

- The list of last 10 calls received/realized
- The number of connected calls and a pie graphic showing the relation between inbound and outbound calls
- The agent recreational pause time and a pie graphic that shows the relation between the total session time and the total pause time
- The number of its engaged dispositioned calls and a pie graphic that shows the relation between engaged dispositioned calls and engaged dispositioned calls tagged as 'observed' on and audit

This dashboard is updated periodically and allows the agent get custom performance información in the current day

Finally, is important to notice that the call list visualized in dashboard shows information about disposition and its engagement if corresponds. Also in this case also allows to modify disposition and engaged values and call the contact using the number phone shown

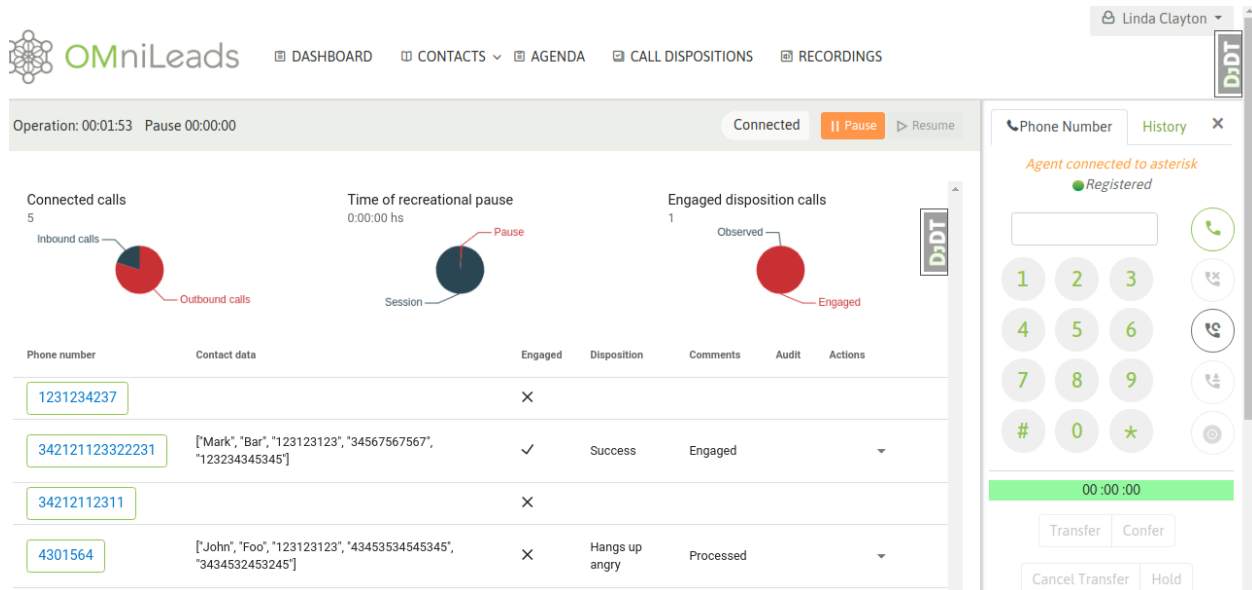


Figure 7: Dashboard

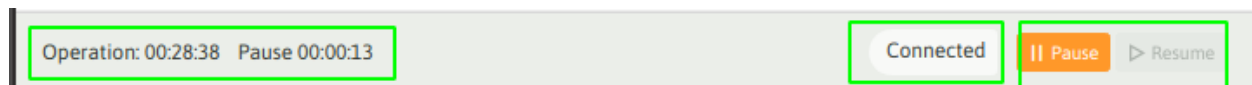
(1) Contact Management

In this section we see the way the agent can navigate among different campaign contacts, scheduled callbacks, dispositioned calls, and also access to Preview Campaigns in order to retrieve contacts to dial.

All the above will be explained in depth in the next sections.

Status Bar

In this bar, session information is shown.



Status Bar

In the status bar there are two chronometers with the amount of agent time in Pause and Ready status. Its color changes according to whether the agent is in the Ready (gray), On call (green) or Paused (yellow) status. A status legend can also be seen as a description. Finally the Pause and Un-pause buttons appear, so that the agent can go to pause and leave it.

Dynamic Information Area

All the information that the agent is requesting while browsing can be seen in this section, as well as Contact Information when agent processes a new call.

(4) Webphone

OMniLeads Webphone is the main component within Telephony module. Let's review the embedded buttons.

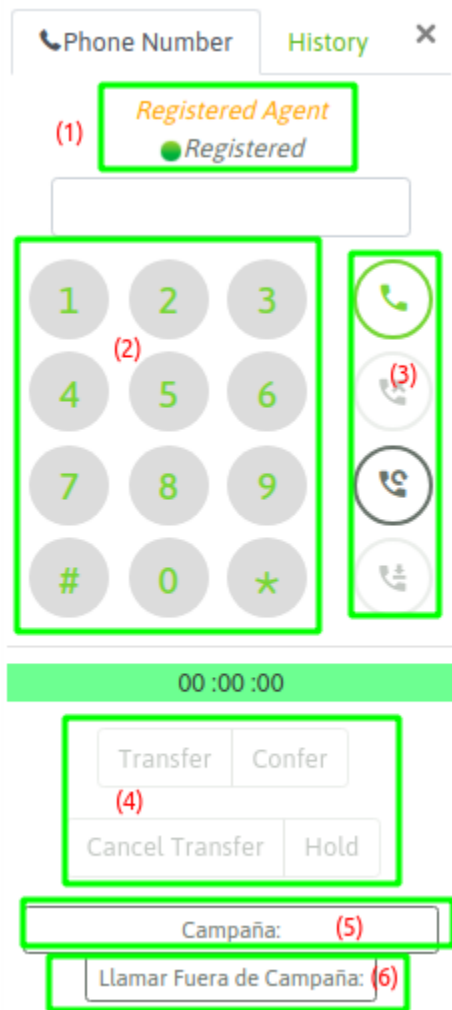


Figure 9: Webphone

Webphone can be separated in 5 pieces:

- (1) Webphone Status. It should be seen always as *Registered*. If that is not the case, please cotact the System Administrator.
- (2) Dialpad: used for dialing numbers manually or pressing DTMF options in a connected call.
- (3) These buttons allow the agent to shoot certain actions: dial, hangup, re-call, and the possibility to “leave a mark” in a conversation, i.e. a specific comment.
- (4) These buttons allow to the agent to make actions like: make a transfer call, make a conference with a third party and hold the call
- (5) This button allows to modify the campaign which is processed each manual call. More information in “manual campaign” section
- (6) This button allows to make a call to another agent, also to make an outbound call to an external number withouth association to a campaign

Almost all of the functionalities are analyzed in depth in the related sections.

Dock

In the current version, OMniLeads just supports Webphone component to make/receive calls. In future versions Product Team will be activating other modules.

Pauses

Agent has the ability to enter in Pause mode so that no incoming or predictive campaign calls are delivered. As explained in the “Initial configuration” section, there are different types of pauses the Administrator can create and maintain in the system. Therefore, when entering the pause status, the agent must select the type of pause.

For entering in Pause mode, Agent must click in the “Pause” button in the status bar of the console.



Figure 10: Pause

Se despliega entonces el menú de selección del tipo de pausa:

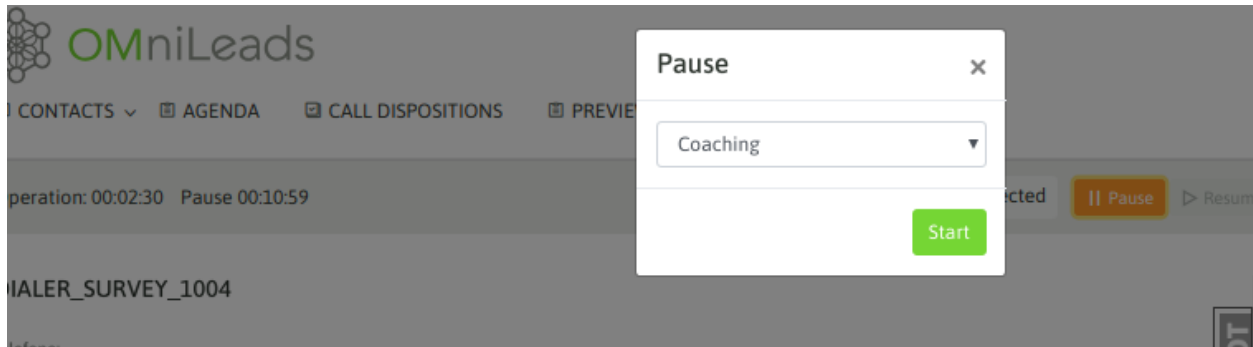


Figure 11: Pause Type

Finally, the agent is paused. It should be noted how the status bar changes to “yellow” color, also the current pause type can be read in the status bar and the Pause timer starts to run. On the other hand, Operation timer stops.

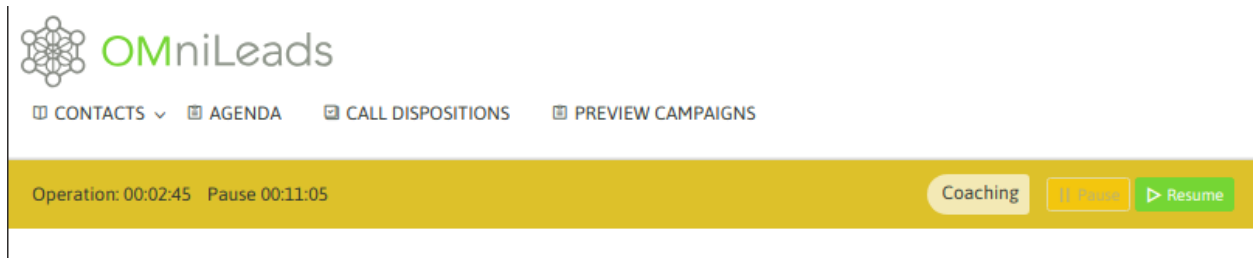


Figure 12: On Pause

Logout

In order to log out of the system, agent needs to click on the exit button as displayed below.

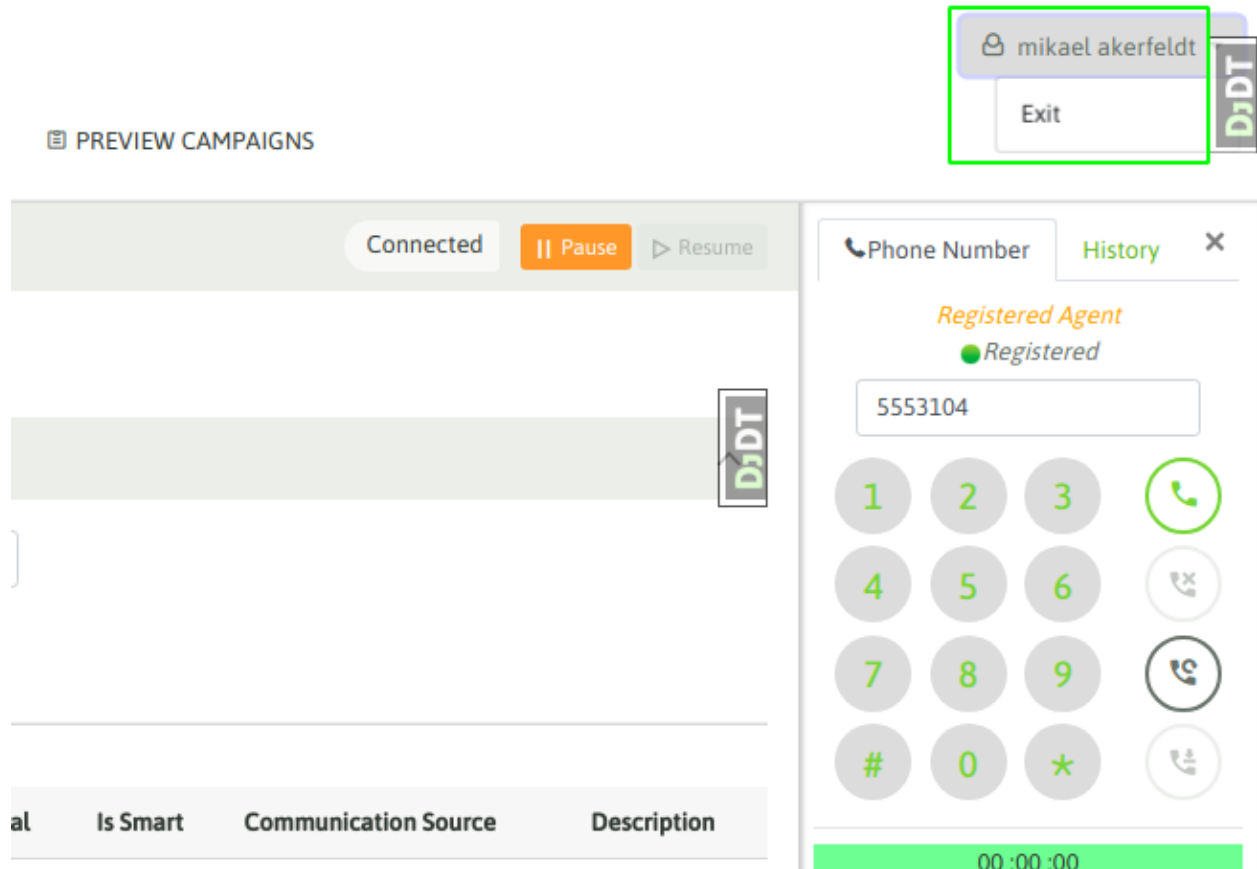


Figure 13: Agent Logout

14.1.2 Manual Campaigns

Agent and Operation in Manual Campaigns.

Manual Calls from Contact List

When an agent is assigned to a Manual Campaign, she/he can make calls from Contact List, this is: going to menu Contacts -> Contact List and selecting the Manual Campaign where the contact is reachable to be dialed. This can be seen in figure 1.

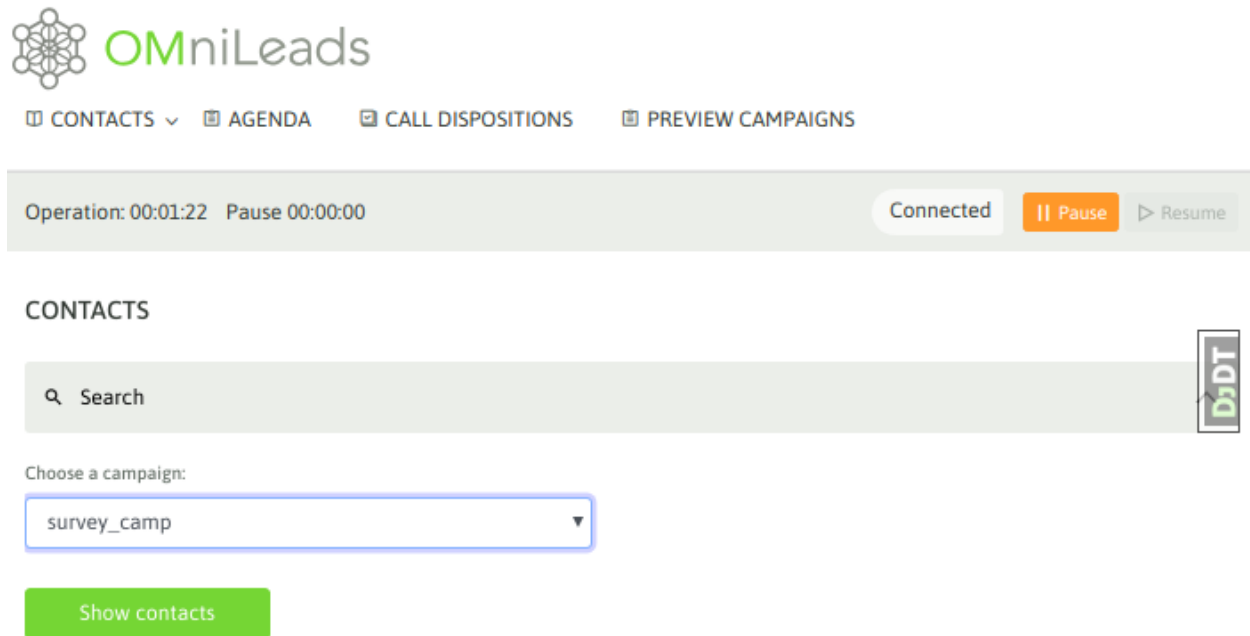
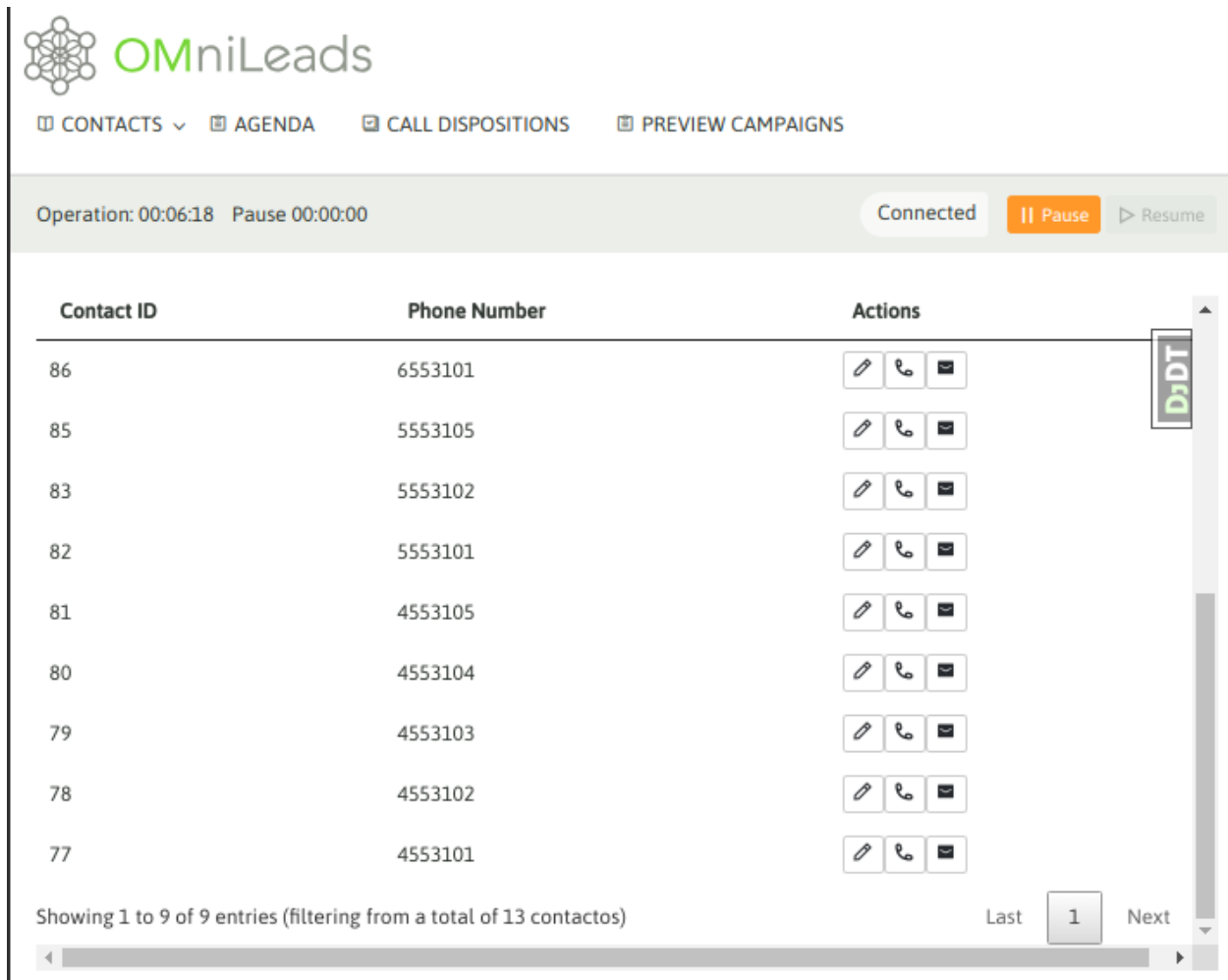


Figure 1: Contacts List

If agent clicks on Show Contacts, all of them appear listed as shown in figure 2.



The screenshot displays the OMniLeads application interface. At the top, there is a navigation bar with the OMniLeads logo and four menu items: CONTACTS (selected), AGENDA, CALL DISPOSITIONS, and PREVIEW CAMPAIGNS. Below the navigation bar, a status bar shows 'Operation: 00:06:18', 'Pause 00:00:00', and a 'Connected' status with 'Pause' and 'Resume' buttons. The main area features a table with three columns: 'Contact ID', 'Phone Number', and 'Actions'. The table lists 9 contacts, each with a unique ID and phone number. The 'Actions' column contains three icons: a pencil (edit), a phone (call), and an envelope (email). A vertical sidebar on the right shows a 'D3DT' button. At the bottom, a pagination bar indicates 'Showing 1 to 9 of 9 entries (filtering from a total of 13 contactos)' and includes 'Last', '1', and 'Next' buttons.

Contact ID	Phone Number	Actions
86	6553101	[edit] [call] [email]
85	5553105	[edit] [call] [email]
83	5553102	[edit] [call] [email]
82	5553101	[edit] [call] [email]
81	4553105	[edit] [call] [email]
80	4553104	[edit] [call] [email]
79	4553103	[edit] [call] [email]
78	4553102	[edit] [call] [email]
77	4553101	[edit] [call] [email]

Showing 1 to 9 of 9 entries (filtering from a total of 13 contactos)

Last 1 Next

*Figure 2: Contacts List

Then, the agent can make a call to any of the phones listed, by clicking on the phone icon. From that moment the contact information is presented in the Agent screen and immediately the system attempts the dial process.

Operation: 00:09:42 Pause 00:01:46

On Call

SURVEY_CAMP

tel1:

5553102

5553102

name:

Jiddu

surname:

Krishnamurti

id_contact:

30000007

tel2:

33213211

33213211

tel3:

33213211

Phone Number History

Connected to 5553102

Registered

00:00:02

Transfer Cancel Transfer Hold

Campaign:

Figure 3: Call to Contact

If the communication has ended or the phone cannot be contacted, then the agent is able to re-attempt with another contact number (if the contact has more than one phone loaded). If this is the case, then the agent can click on any of the extra telephone numbers and will automatically dial the new number.

OMniLeads

CONTACTS AGENDA CALL DISPOSITIONS PREVIEW CAMPAIGNS

Operation: 00:17:30 Pause 00:09:17

On Call

Jiddu

surname:

Krishnamurti

id_contact:

30000007

tel2:

33213211

33213211

tel3:

5343431

5343431

Call Disposition:

Comments:

Phone Number History

Connected to 5343431

Registered

00:00:15

Transfer Cancel Transfer Hold

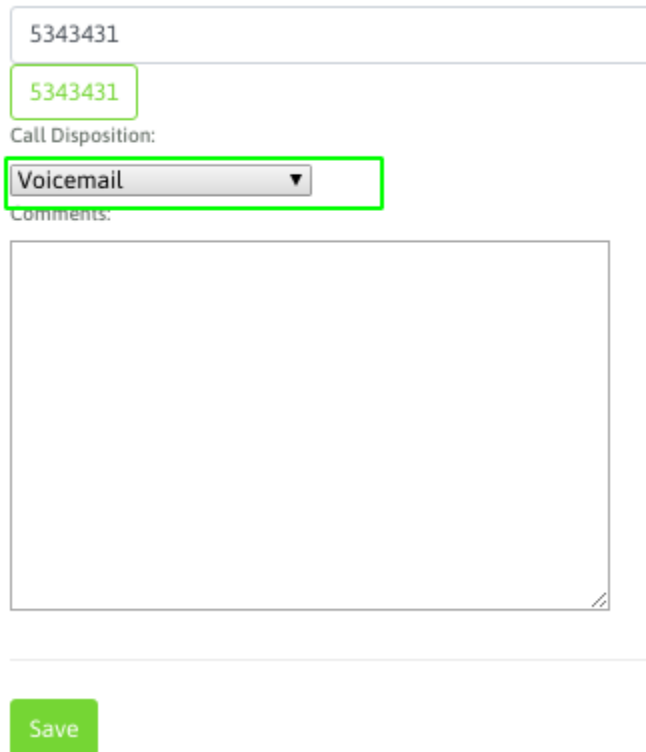
Campaign:

Figura 4: Callback

All calls from the contact can be dispositioned, as long as the call disposition is not managed.

Finally, the agent is able to disposition the call through the Call Dispositions available. This list of Call Dispositions

was generated by the administrator for each campaign.

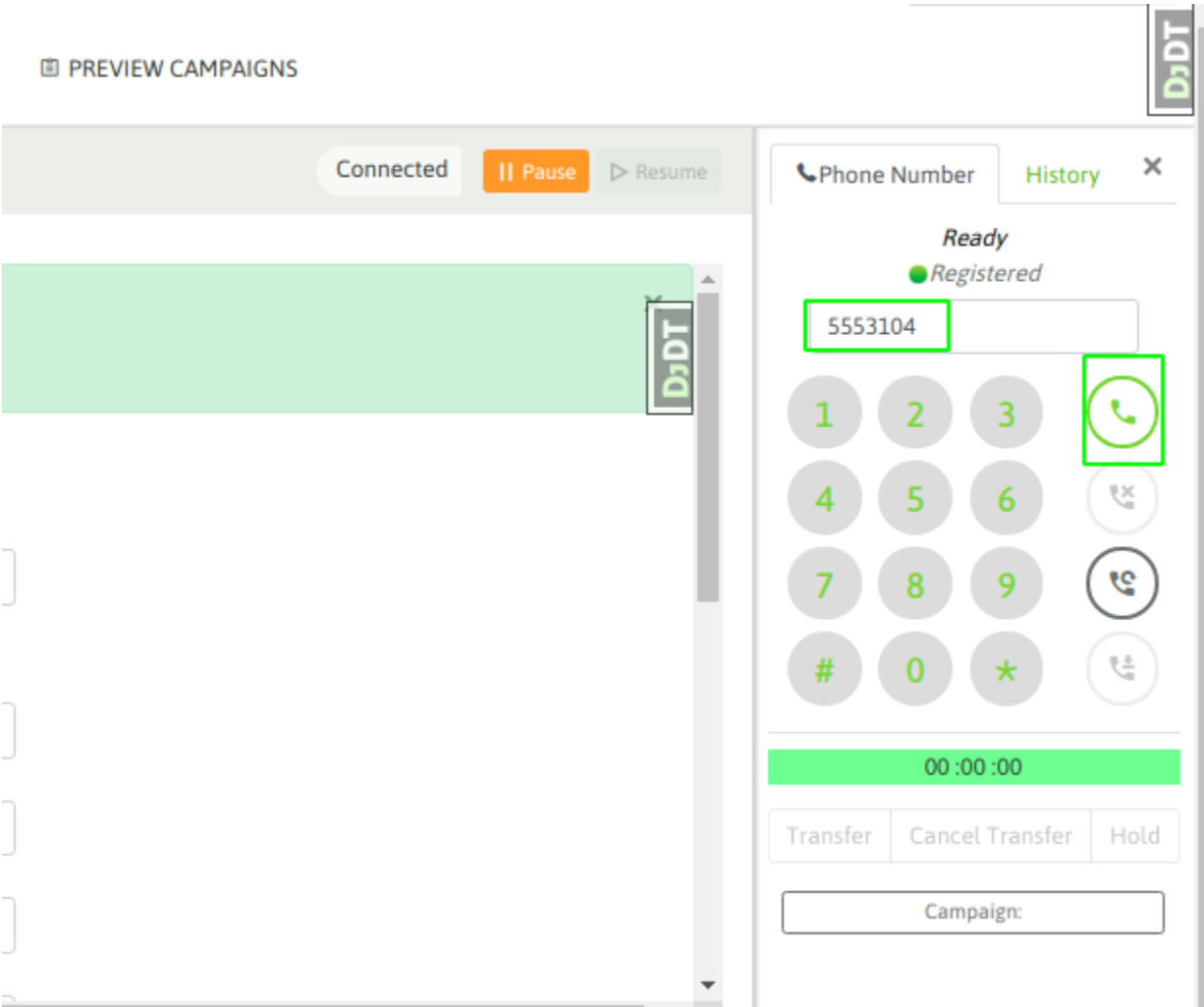


The screenshot shows a web form for recording call dispositions. At the top is a text input field containing the number '5343431'. Below it is a smaller green-bordered box also containing '5343431'. The label 'Call Disposition:' is positioned above a dropdown menu that currently displays 'Voicemail'. Below the dropdown is a label 'Comments:' followed by a large, empty text area for notes. At the bottom left of the form is a green 'Save' button.

Figure 5: Call Disposition

Manual Calls dialing from Webphone

The agent can make calls directly from the webphone. It is typical in some Call Centers to distribute contacts between agents using a spreadsheet or looking for the data in an external CRM.



Manual Calls dialing from Webphone

Just pressing * enter * button or clicking on * dial * button, the call is made. If there is no pre-selected campaign, the system asks to select one prior to attempt the call.

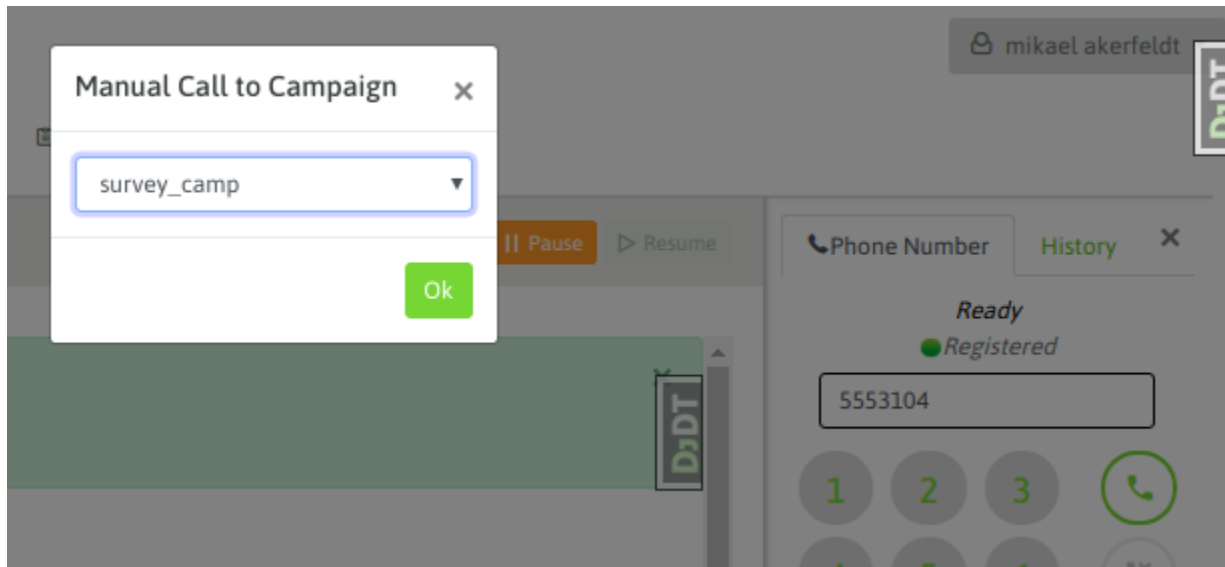


Figure 7: Campaign Selection

OMnileads makes a lookup of the dialed phone within the Contact Lists assigned. If it exists, all the contacts matching that number are listed. Then the agent selects the contact to dial and the call is launched presenting Contact Information in the agent console.

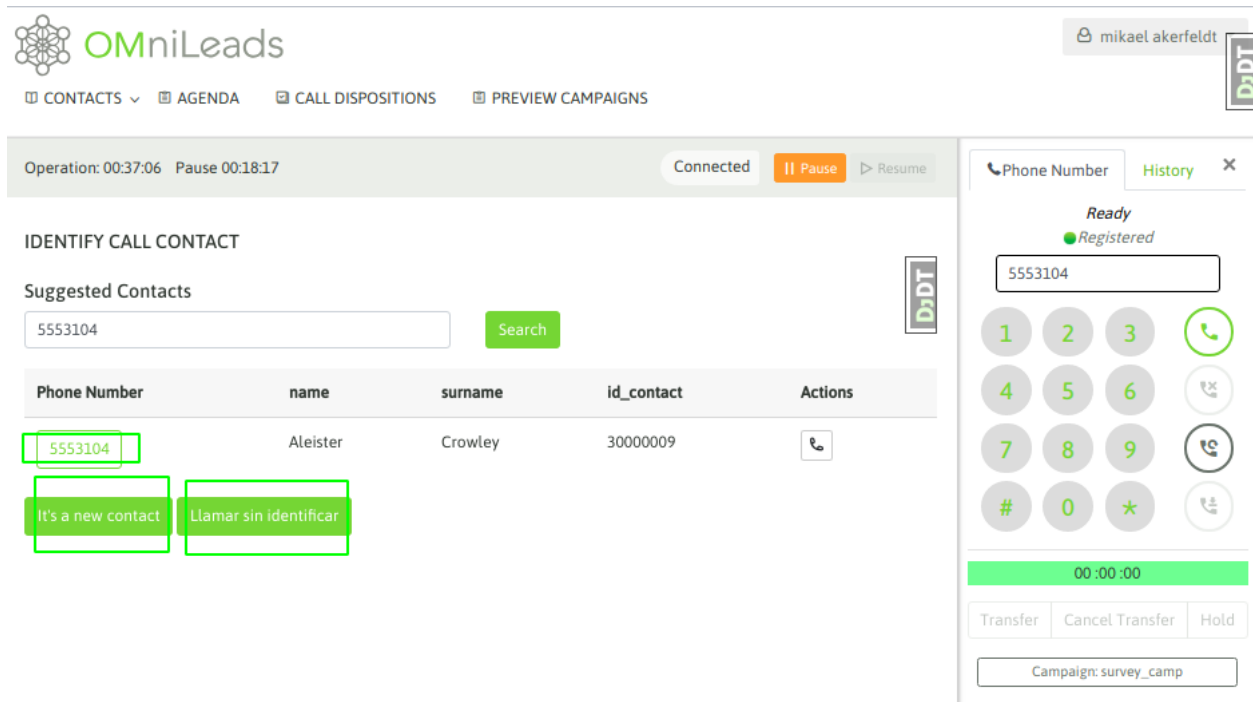


Figure 8: Contact Selection

SURVEY_CAMP

tel1:

5553104

5553104

name:

Aleister

surname:

Crowley

id_contact:

30000009

tel2:

33213211

33213211

tel3:

D3DT

Connected to 5553104

Registered

5553104

1234567890*#

00:00:01

TransferCancel TransferHold

Campaign: survey_camp

Figure 9: Call to Contact

Manual Calls to Non-Existing contacts

It can also happen that the dialed phone number does not match any contact, as shown in Figure 10.

IDENTIFY CALL CONTACT

Suggested Contacts

3424234234

Search

Phone Number	name	surname	id_contact	Actions
It's a new contact		Llamar sin identificar		

D3DT

Agent in Pause

Registered

3424234234

1234567890*#

00:00:00

TransferCancel TransferHold

Campaign: survey_camp

Manual Calls to Non-Existing contacts

So, the agent can directly dial the phone and then save the new contact filling the campaign form, as shown in Figure 11.

SURVEY_CAMP

tel1:
3424234234

name:

surname:

id_contact:

tel2:

tel3:

Call Disposition:
----- ▾

Agent in Pause
● Registered

3424234234

1 2 3 4 5 6 7 8 9 # 0 *

00:00:00

Transfer Cancel Transfer Hold

Campaign: survey_camp

Figure 11: Call to Contact without Identification

Or load the new contact to the campaign before dialing the number as indicated in figure 12.

tel1: 4324232

name:

surname:

id_contact:

tel2:

tel3:

Save and call

Agent in Pause
● Registered

4324232

1 2 3 4 5 6 7 8 9 # 0 *

00:00:00

Transfer Cancel Transfer Hold

Campaign: survey_camp

Figure 12: Load Contact and Call

14.1.3 Preview Campaigns

Agent and Operation in Preview Campaigns.

Preview Campagins

When an agent is assigned to a Preview Campaign, Preview menu needs to be used.

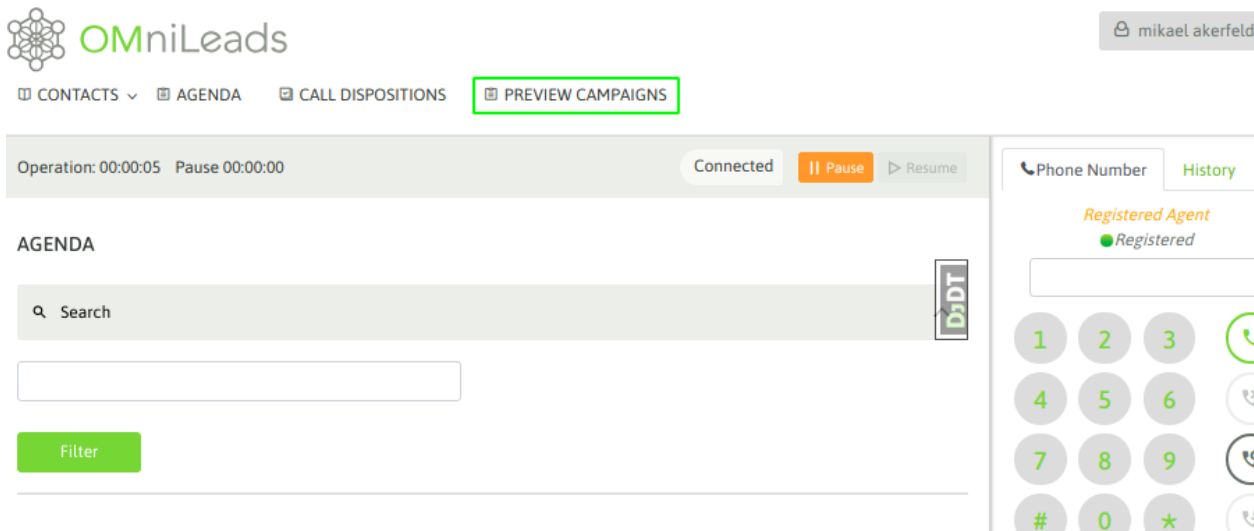
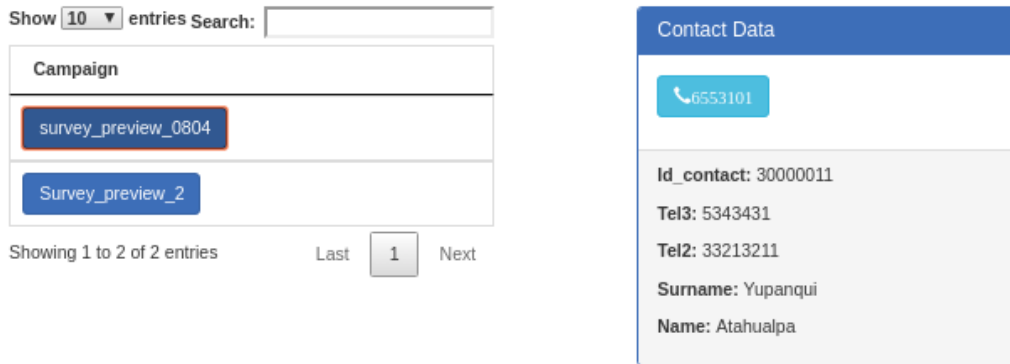


Figure 1: Preview Campaigns List

All assigned Preview Campaigns are listed here. Agent must select one of them and this will return a contact to manage.



Figure 2: New Contact retrieve



*Figure 3: Delivered Contact

Once the contact is delivered, the agent can either dial the phone number by clicking on the number, or by clicking again on the preview campaign in order to retrieve another contact. The last one will be released.

Assuming a contact is called by clicking on it, contact information is displayed on the agent's view, as we have seen in Manual Calls section.

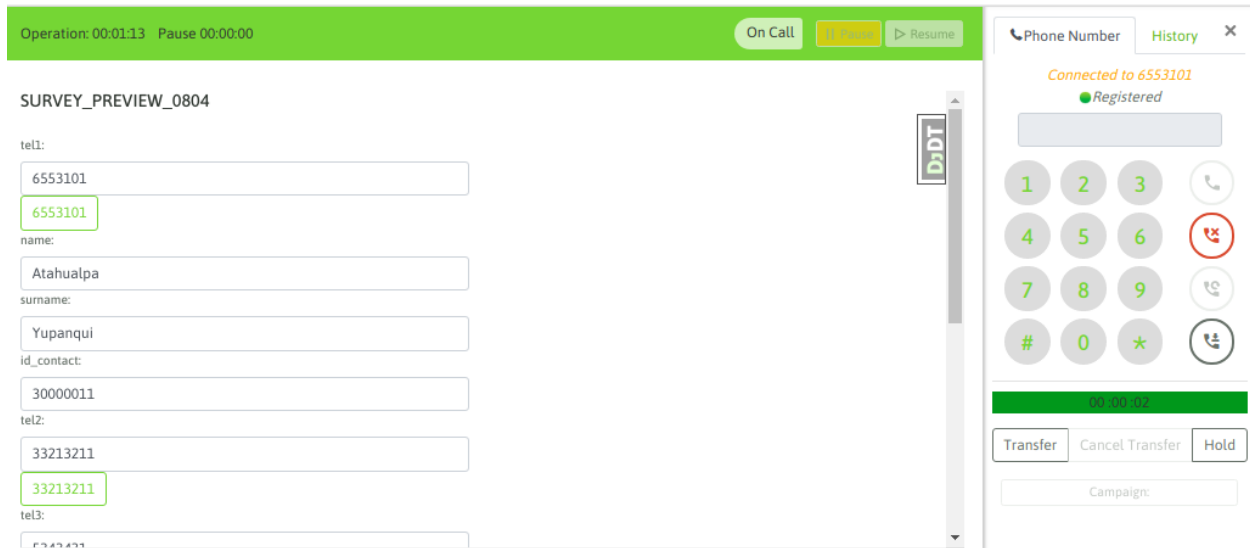
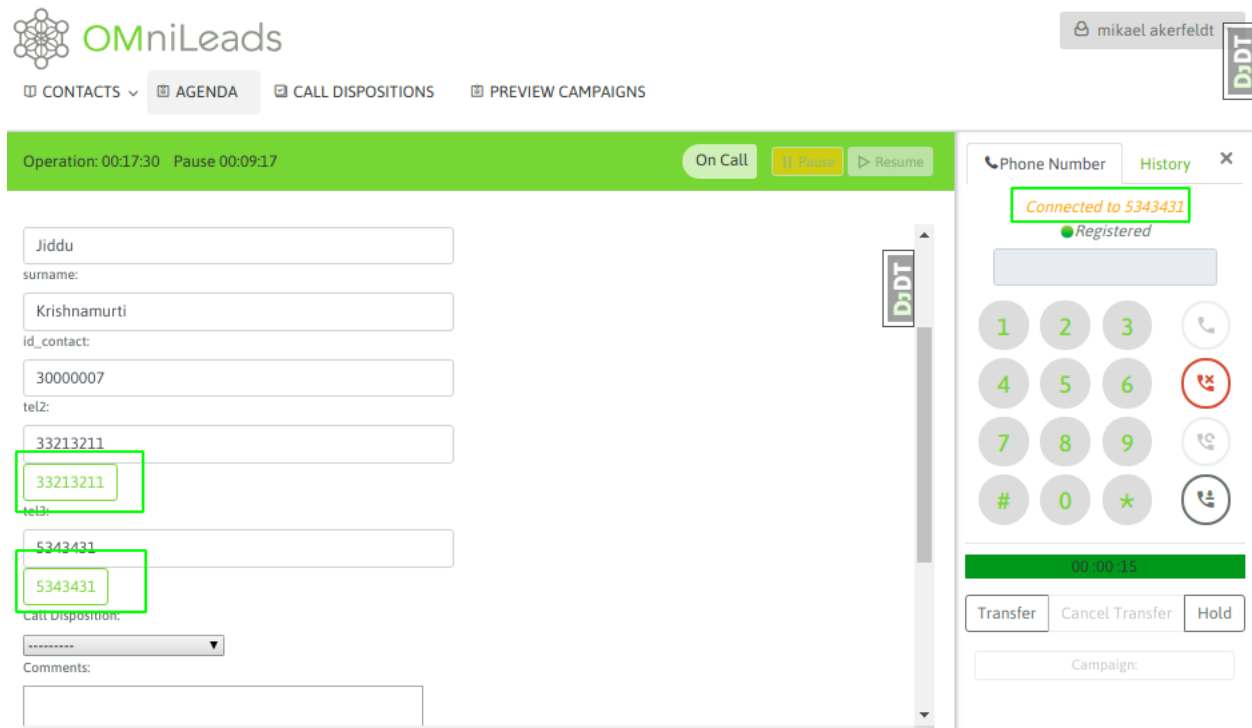


Figure 4: Dialed Contact

If the communication has ended or the phone could not be contacted, then the agent can try again with another phone number (if the contact has more than one phone loaded). If this is the case, then the agent can click on any of the extra telephones and will automatically dial the contact to the new number.

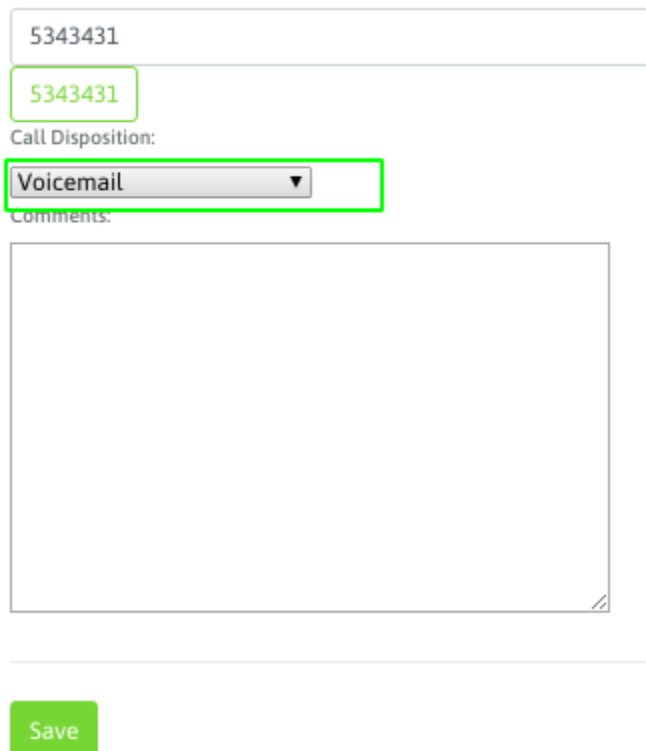


The screenshot shows the OMniLeads interface. At the top, there's a navigation bar with 'CONTACTS', 'AGENDA', 'CALL DISPOSITIONS', and 'PREVIEW CAMPAIGNS'. A green status bar indicates 'Operation: 00:17:30 Pause 00:09:17' and 'On Call' status. The main form displays contact details for 'Jiddu' with surname 'Krishnamurti' and ID '30000007'. The 'tel2' field is '33213211' and 'tel3' is '5343431', both highlighted with green boxes. The 'Call Disposition' dropdown is set to 'Voicemail'. A 'Comments' field is at the bottom. On the right, a 'Phone Number' panel shows 'Connected to 5343431' and 'Registered' status, with a numeric keypad and call control buttons like 'Transfer', 'Cancel Transfer', and 'Hold'.

Figure 5: “Callback to Contact”

All calls from the contact can be dispositioned, as long as the call disposition is not managed.

Finally, the agent is able to disposition the call through the combo of Call Dispositions. This list has been generated from the Admin view.



This screenshot shows a close-up of the 'Call Disposition' dropdown menu. The 'tel2' field is '5343431' and 'tel3' is '5343431', both highlighted with green boxes. The 'Call Disposition' dropdown is open, showing 'Voicemail' as the selected option, also highlighted with a green box. A 'Comments' field is at the bottom. A green 'Save' button is at the bottom left.

Figure 6: Call Disposition

The agent have the possibility to select more quickly the next contact to call in the last preview campaign select by clicking on the that is shown in the operation bar

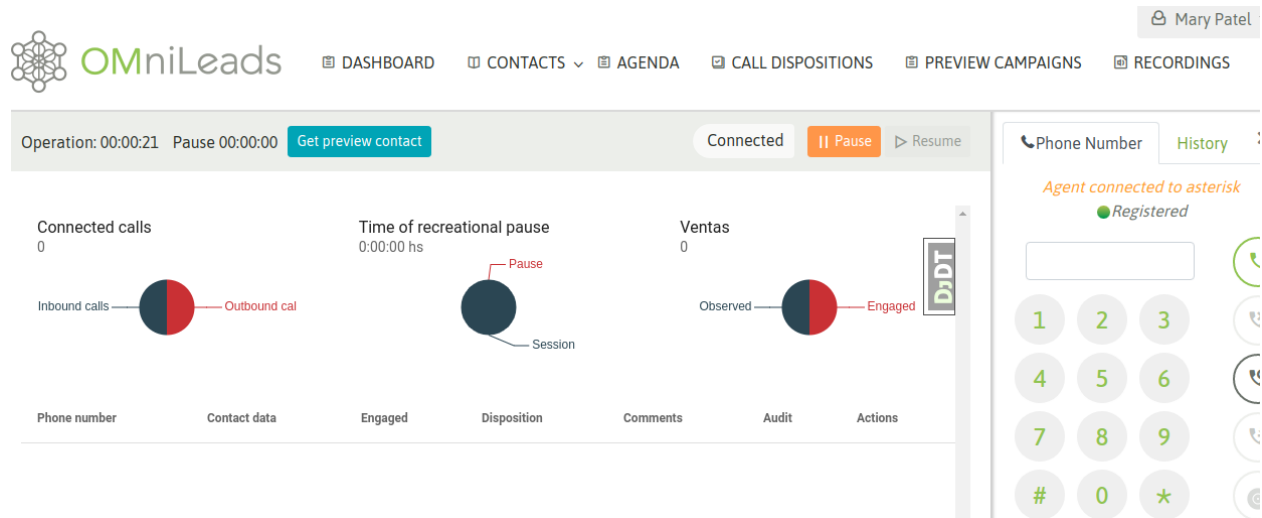


Figure 7: Retrieve a new preview contact

14.1.4 Outbound Dialer Campaigns

Agent and Operation in Predictive Dialer Campaigns.

Agent in Predictive Mode

An active Agent assigned to an active Predictive Campaign, is able to receive calls from the dialer module.

The incoming calls are notified with a “beep” audio that is reproduced to the agent headset as soon as the call is delivered. Contact details are also presented to the agent once the call is answered.

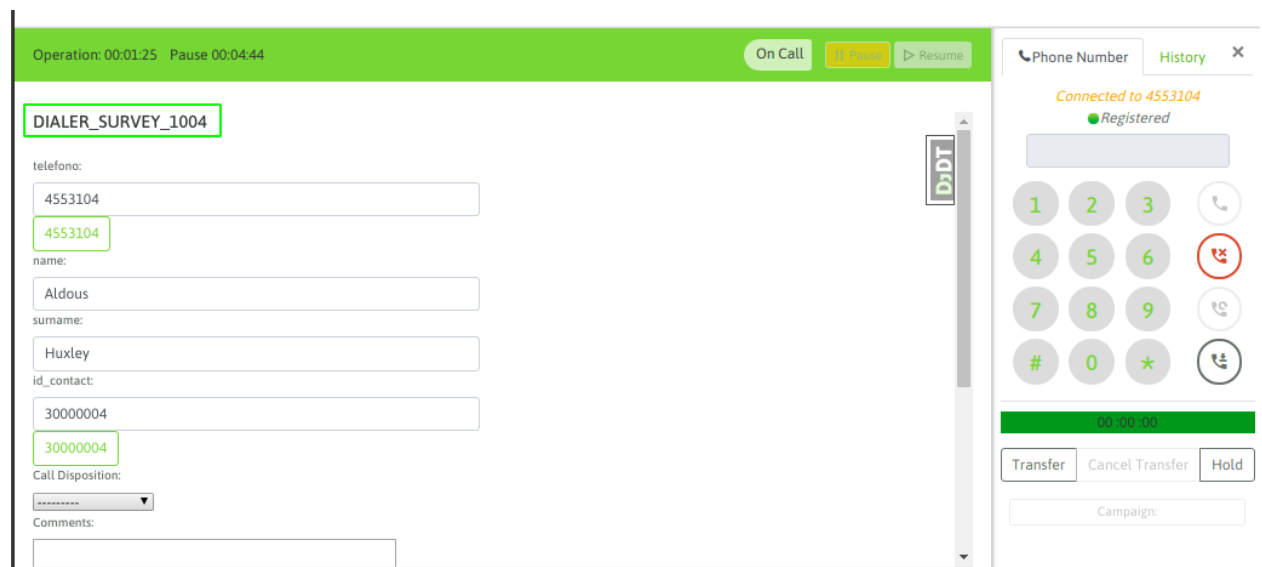


Figure 1: dialer call connect

Note: It is important to note that an agent working on a Predictive Campaign needs to be very alert as the system is configured to deliver calls to agents as soon as the contact has been reached and answers the attempt.

In addition to the reproduction of audio notification (beep message), the agent can notice the change in the status bar of the screen. The bar has green color when agent is “On call” and also the name of the campaign the call belongs to can be read.

Once the call ends (regardless of which end finishes the call) , the system forces the Agent status to ACW (After Call Work), which allows the agent to finish filling the contact form with the respective Call Disposition. After that, the agent will change this ACW status to available again , so that the dialer knows she/he is ready to receive new calls.

Note: Agent is able to exit ACW pause status by inducing the agent to leave the pause or after a period of time (in seconds) defined by the administrator. In the latter case, the agent simply has that grace period for call disposition process and then automatically will be put back inReady for new calls.

How to make a Manual Call

There may be cases in which the agent assigned to a predictive campaign needs to dial a number manually. In this case the agent is suggested to enter to pause mode and then from that state, generate the manual call. Since otherwise it may happen that at the time of being dialing the manual call, the dialer sends a predictive call and drops manual call attempt.

This manual call can be done to other phone numbers of the same contact, or external numbers out of database. The important thing is to remember to go to pause at the time of the manual attempt.

14.1.5 Inbound Campaigns

Agent and Operation in Inbound Campaigns.

Incoming Call Distribution

As we well know from the “Initial configuration” section, the system can be configured for incoming calls to generate a “Ring” on the agent’s webphone, giving the possibility of deciding whether or not to answer the call. Or can be configured to directly connect incoming calls to agents (that is to say, auto-answer the call).

Behavior can be seen in figure 1.

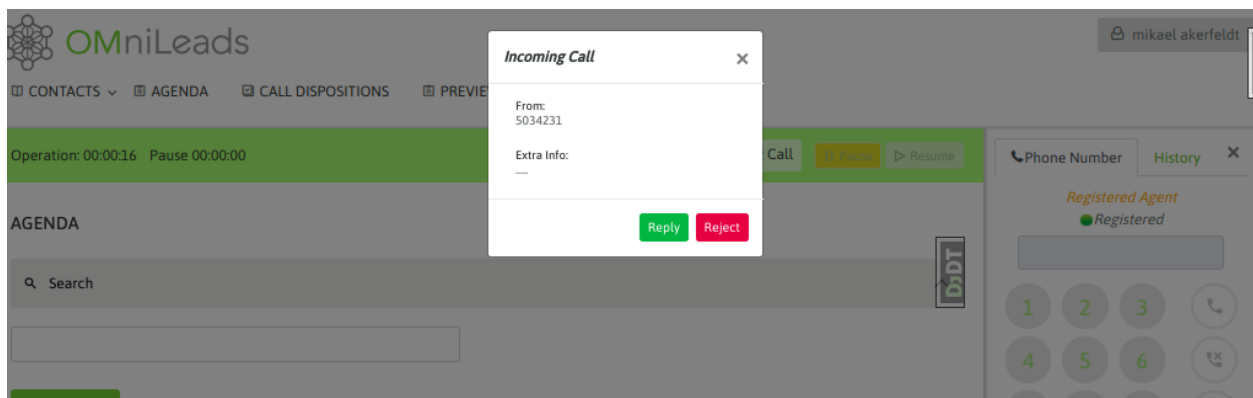
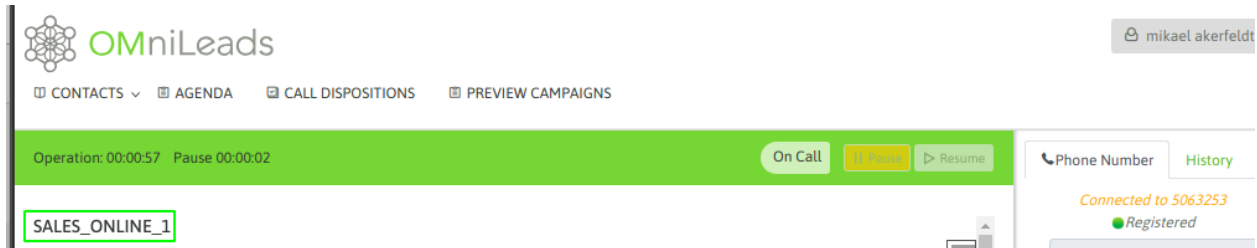


Figure 1: Inbound Call Ringing

If the configuration is in auto-answer mode (like a predictive campaign) , the agent will hear a “beep” message once the call is connected and will see the name of the campaign the call belongs to.



Figurae2: Inbound Campaign Call

Contact Management

Incoming calls can be associated to contacts already available in database, or new contacts to be saved.

SALES_ONLINE_1

☐ Do you want to disposition the call?

telefono:

5063253

nombre:

apellido:

dni:

telefono2:

telefono3:

Figure 3: Inbound Campaign Call without Contact Data

In this case, the agent can decide whether to disposition the call or not. In the positive case, then the agent can proceed with the contact information upload and subsequent call disposition.

14.1.6 Call Actions

When Agent is on call, it counts on with multiple features that improve the call management.

Call to an Agent

To make a call to another OMniLeads agent, we must go to the “Call out of Campaign” button available at the bottom of the webphone. When agent clicks on it, a window is displayed in order to facilitate the selection of an agent from a list available and then the call is performed (Figure 1).

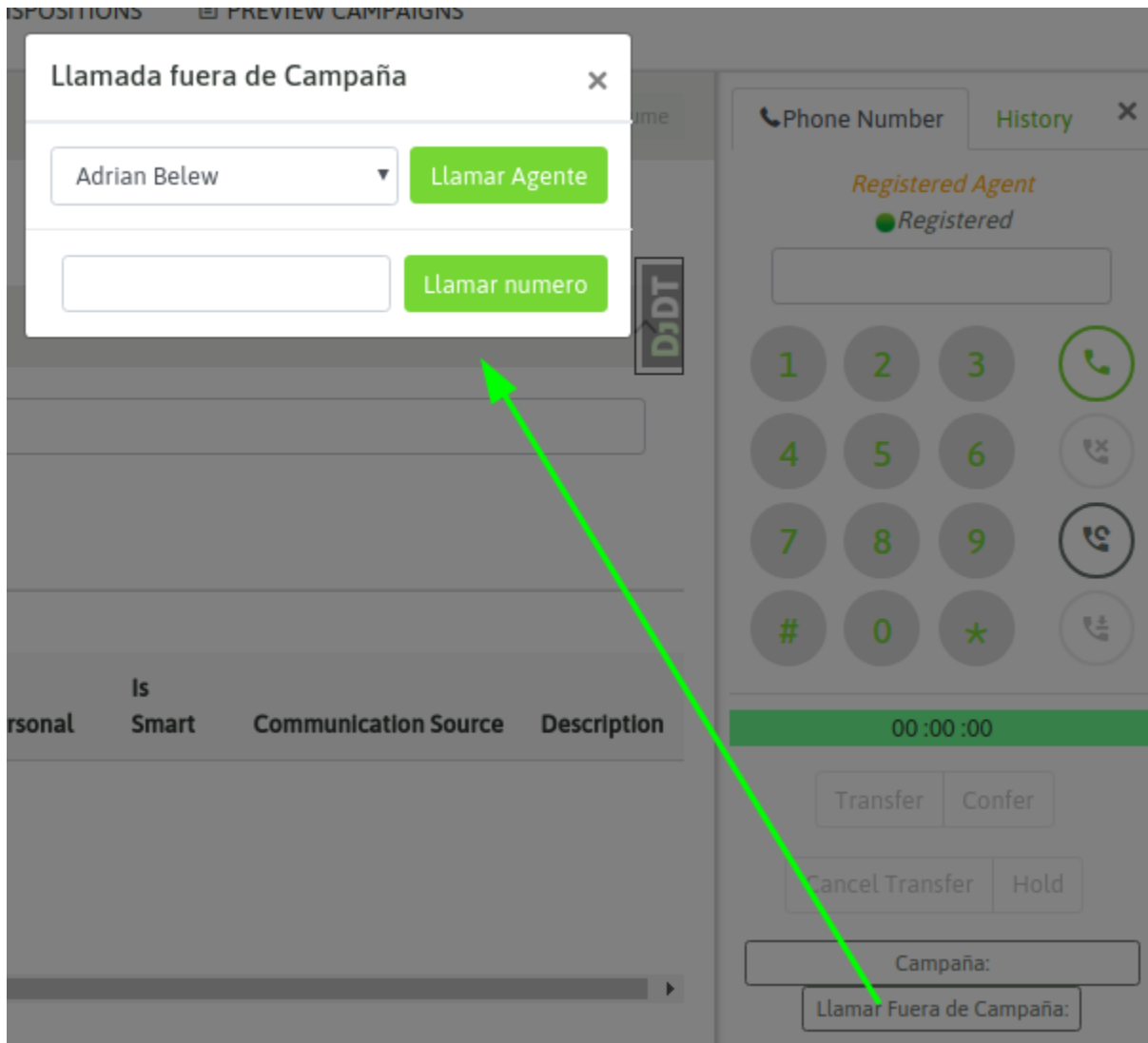


Figure 1: Out of Campaign - Call

Outbound Calls Out of Campaign

Sometimes, it is necessary to make an outside call (number of subscriber or extension of the PBX exchange), without the need to manage the contact. This is allowed by clicking the button “Call outside of campaign” available at the bottom of the webphone.

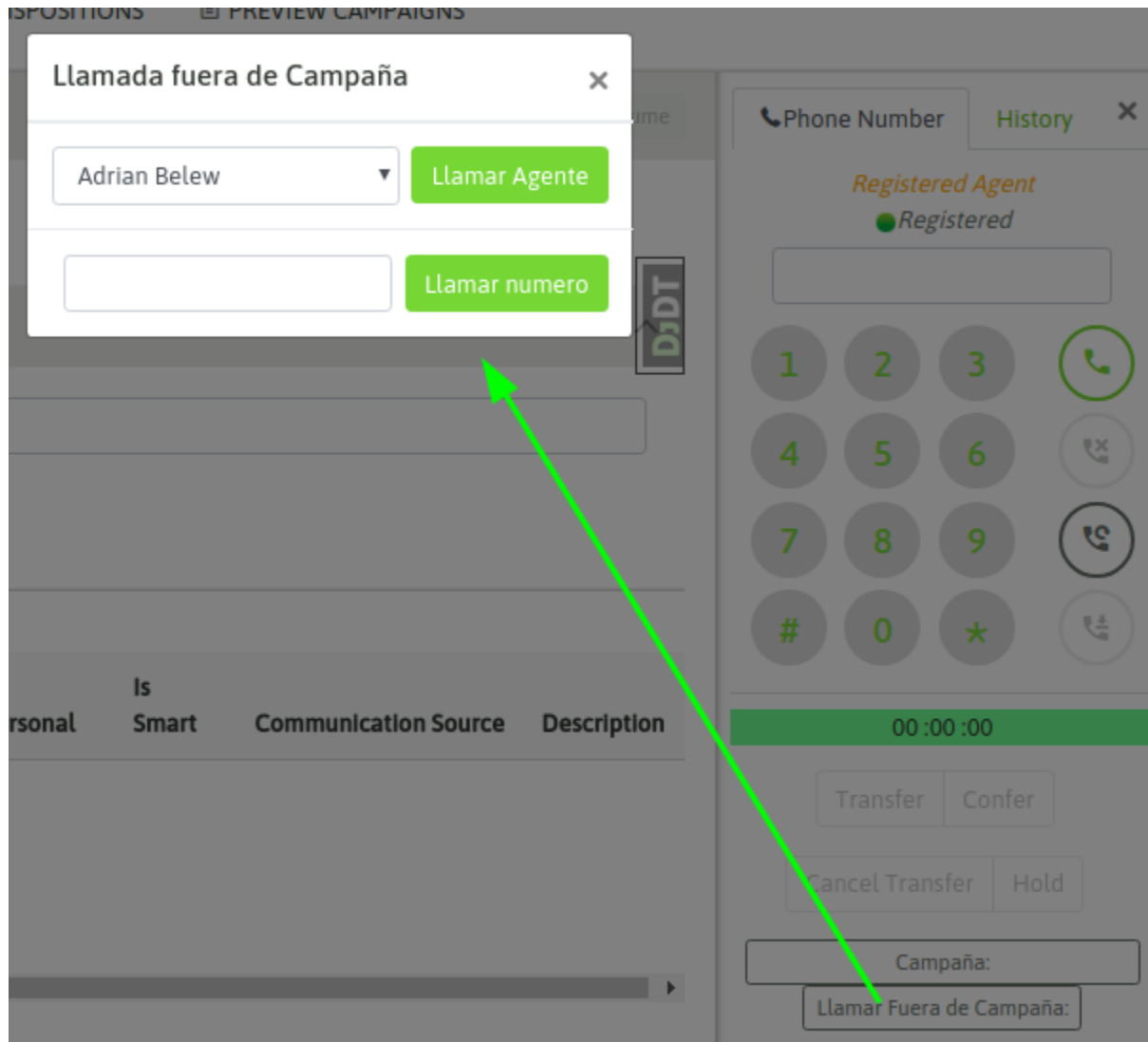


Figure 2: Out of Campaign Call

Put a call On Hold

During a call, the agent has the ability to hold the call for a while. This is possible by clicking the “hold” button in the webphone.

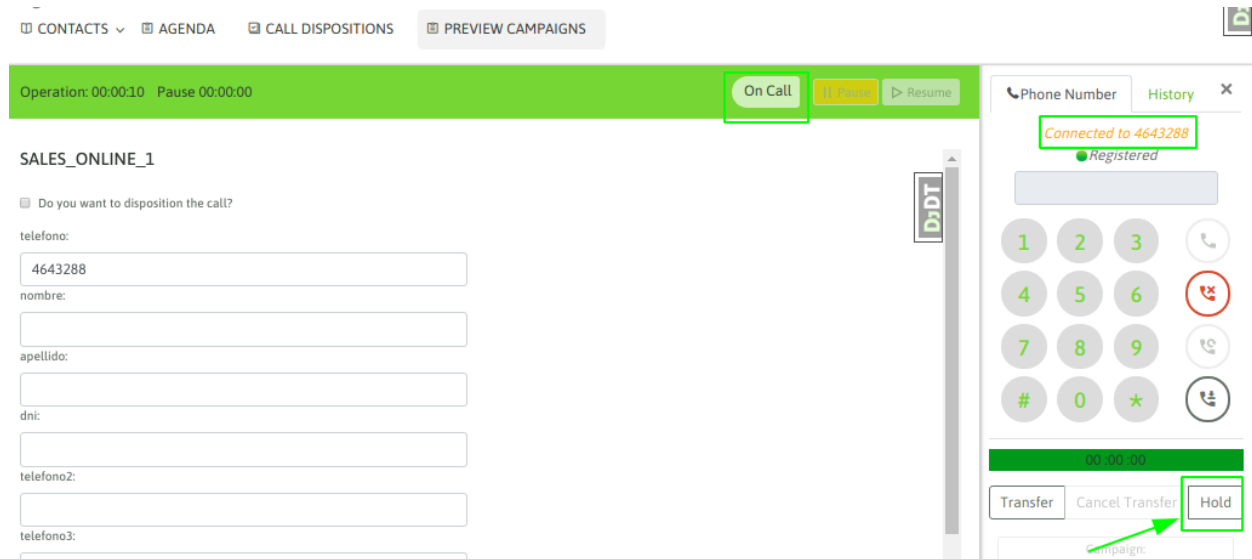


Figure 13: On Hold Call

When hold button is clicked, the customer will listen hold music, while the agent can unhold the call when he wishes, just clicking in unhold, as seen in figure 2.

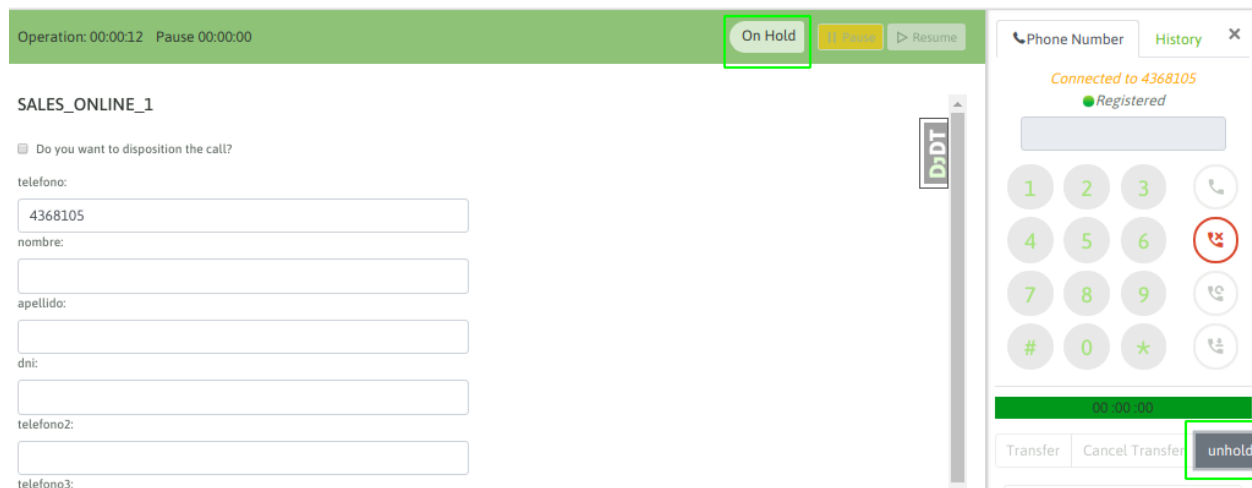


Figure 14: Personal Callback details

This functionality can be used with any type of call.

Call transfer and Conferences

Within the range of possibilities for call transfers that can be made in the system, we have the following:

Blind Transfer to another Agent

Suppose that “agent A” has an active call and she/he wishes to transfer the call to “agent B”, directly. In this case agent clicks the transfer button, inside the webphone and then selects “blind transfer”. After that, she/he can select the agent.

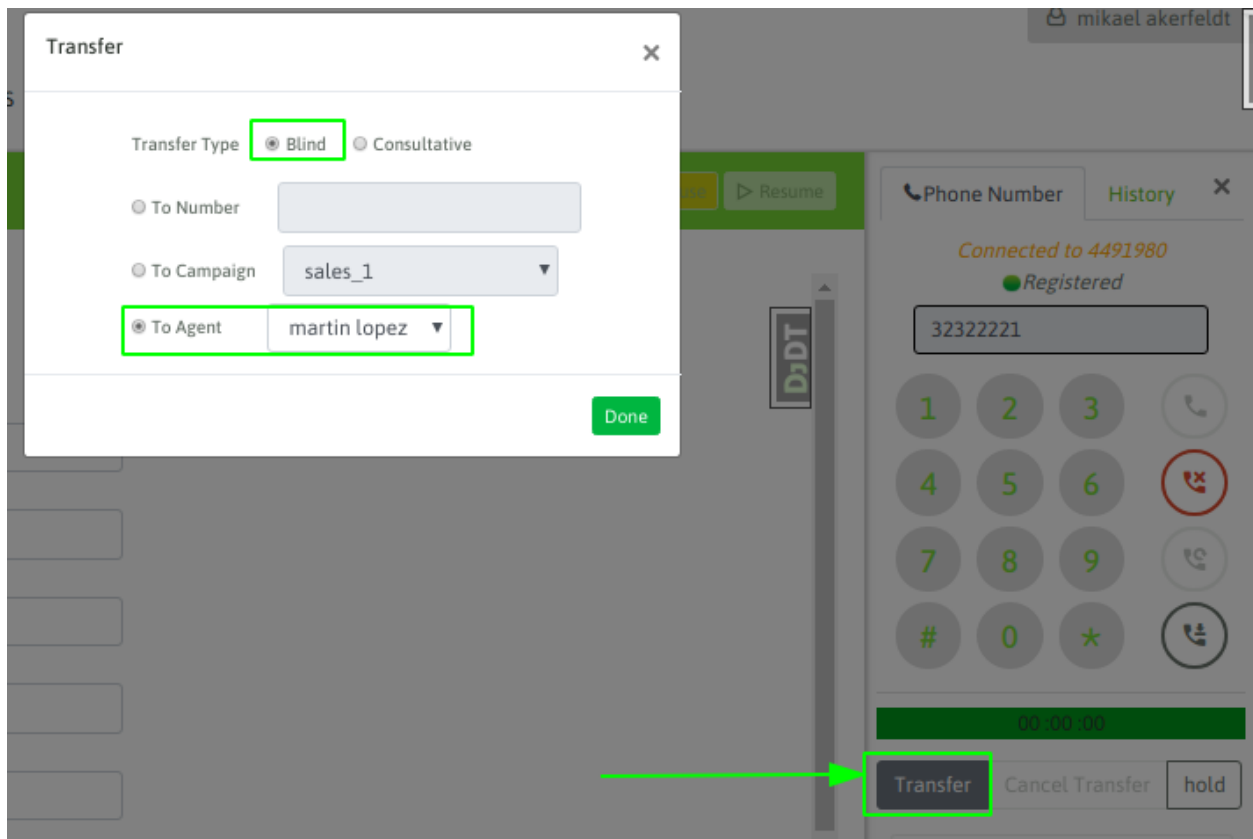


Figure 5: Blind Transfer from “agent A” to “agente B”

In this case the call is automatically transferred to “agent B”, leaving free “agent A”. Once the transfer is made, it cannot be recovered, neither “agent A” can know if the transfer was attended by “agent B”.

Blind transfer to an external number

The “Agent A” is on an active call and wishes to blind transfer the call to an “external number”. When we say external, we mean a call that is generated outside the system. It can be a PBX extension within the company or an external telephone number of the PSTN.

In this case, the transfer button is clicked and then agent can select “blind transfer” as the type of transfer. You must enter the destination number in the box as indicated in the figure 5.

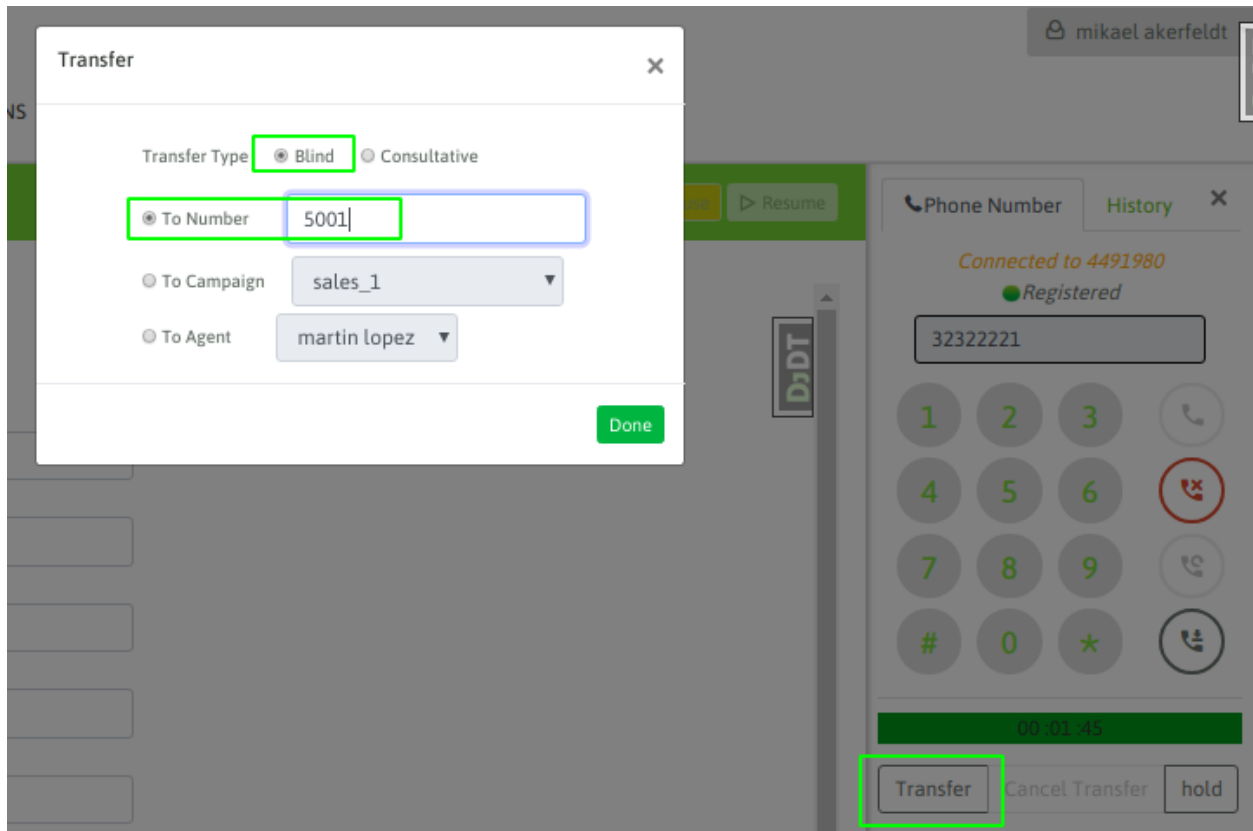


Figure 6: Blind Transfer from “agent A” to “External Number”

In this case, the call is automatically dispatched to the phone destination, and the webphone of the “agent A ” is released. Once this transfer is performed, the call cannot be retrieved to original source, and “agent A” cannot know if the call was attended or not by the transfer destination phone.

Consultative Transfer to another Agent

The “agent A” is on an active call and wishes to transfer the call to the “agent B” in a consultative manner; that is to say that the external telephone is put on hold while the “agent A” opens a new channel to the “agent B”. If the call between them is established and the “agent B” wishes to receive the transfer, then the “agent A” drops the call and automatically the external telephone is bridged with the “agent B”.

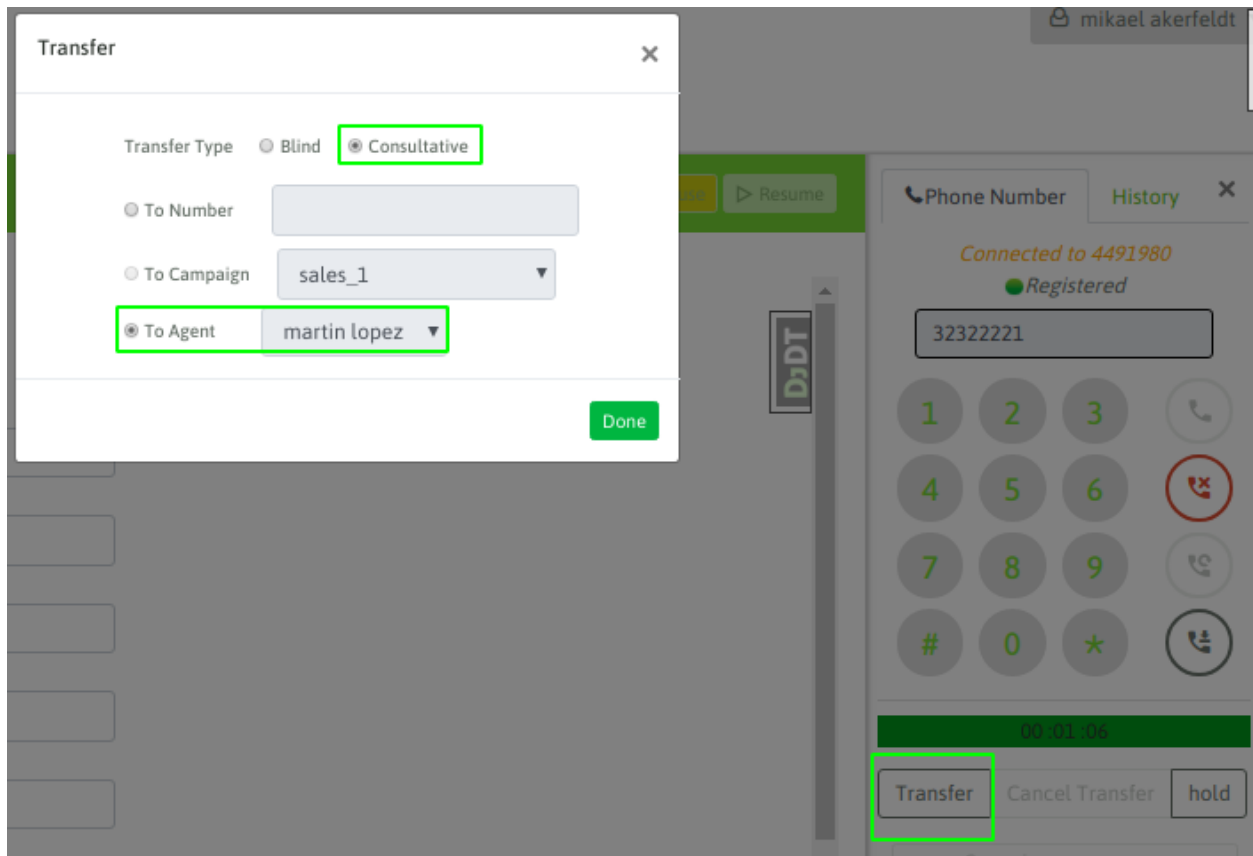


Figure 7: Consultative Transfer from “agent A” to “agent B”

In this scenario, another possibility is:

- That it is not possible to contact the “agent B”, then the “agent A” can cancel the transfer during the ringing phase to the “agent B” using the button “Cancel Transfer” of the webphone.
- The contact with the “agent B” is achieved but the last one cannot (or does not want to) proceed with the transfer. Therefore the “agent B” must drop the call and automatically the “agent A” remains with the connected conversation.

Three Way Conference: external number, Agent A and Agent B

This is a possible scenario within a consultative transfer, since the action to be executed by the agent that drives the conference (agent A), is initially a consultative transfer. At the time of establishing conversation between “agent A” to the “agent B” (while the external person or external number is on hold) the “agent A” must click the “Confer ” button available on the agent Webphone and thus remain the three parties in a conference call.

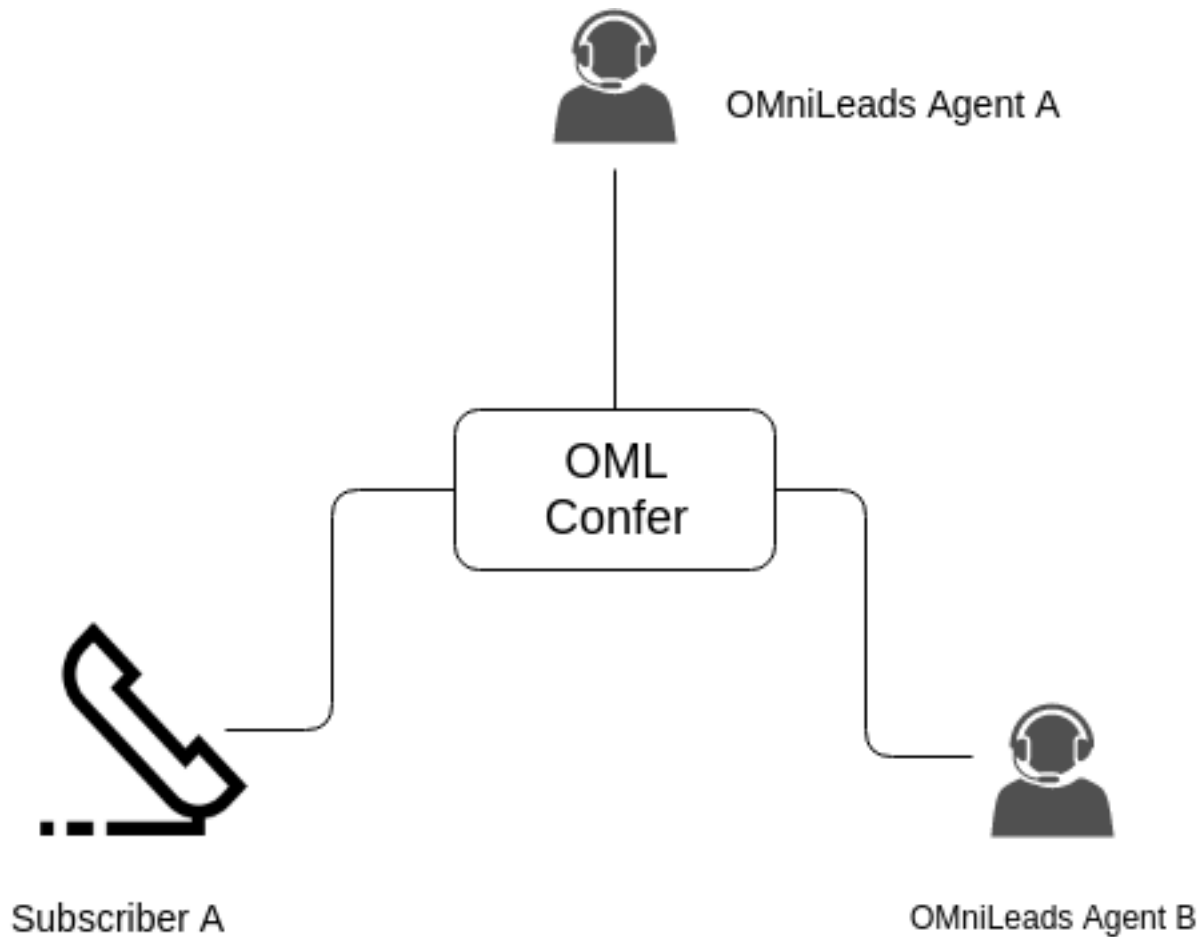
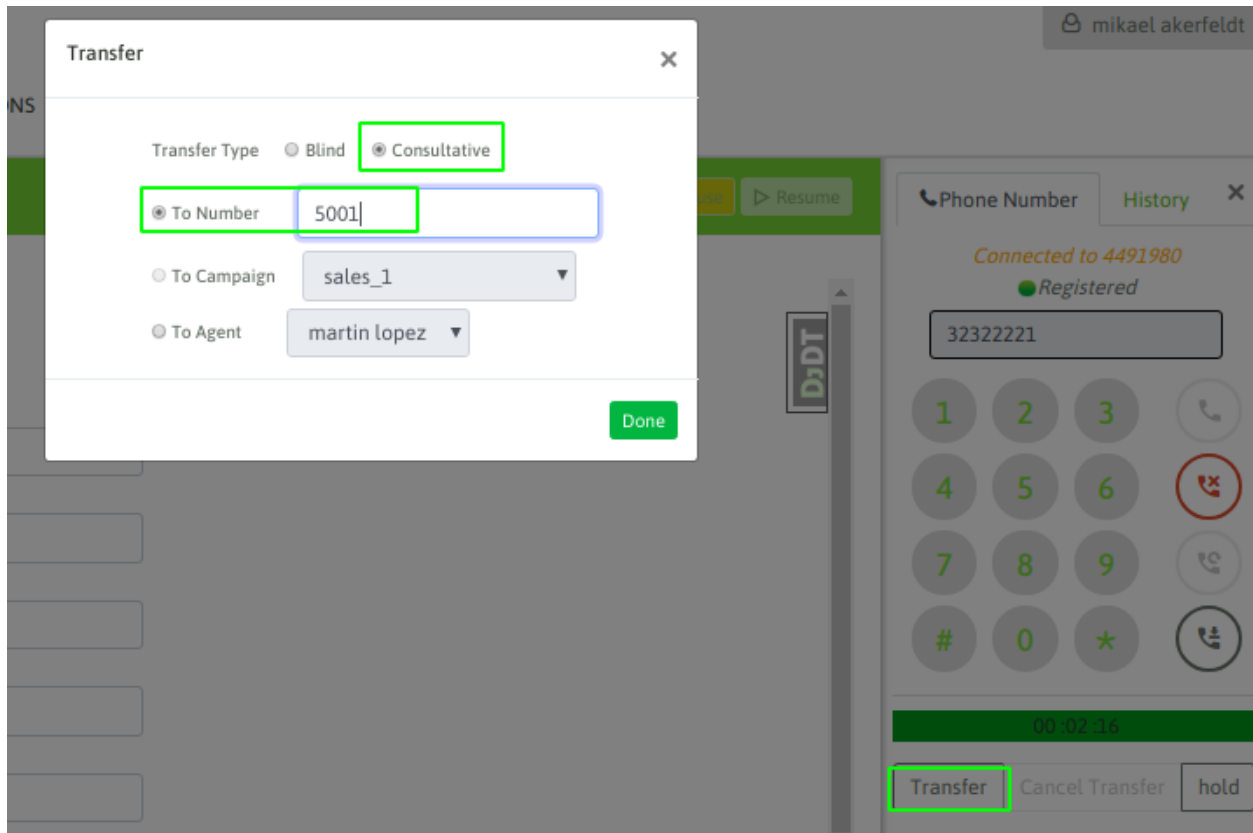


Figure 8: Three-way-Conference between “agent A”, “agent B” and External Number

Consultative Transfer to external number

“Agent A” is on an active call and wants to transfer the call to an external “phone number” in a consultative way, that is making the “external phone A” standby while the “agent A” opens a new channel to “external phone B”. If the call between both is established and the “external phone B” wants to receive the transfer, then “agent A” hangs up the call and automatically “external phone A” is joined in a call with the “external phone B”. This is shown below:



Consultative Transfer to external number

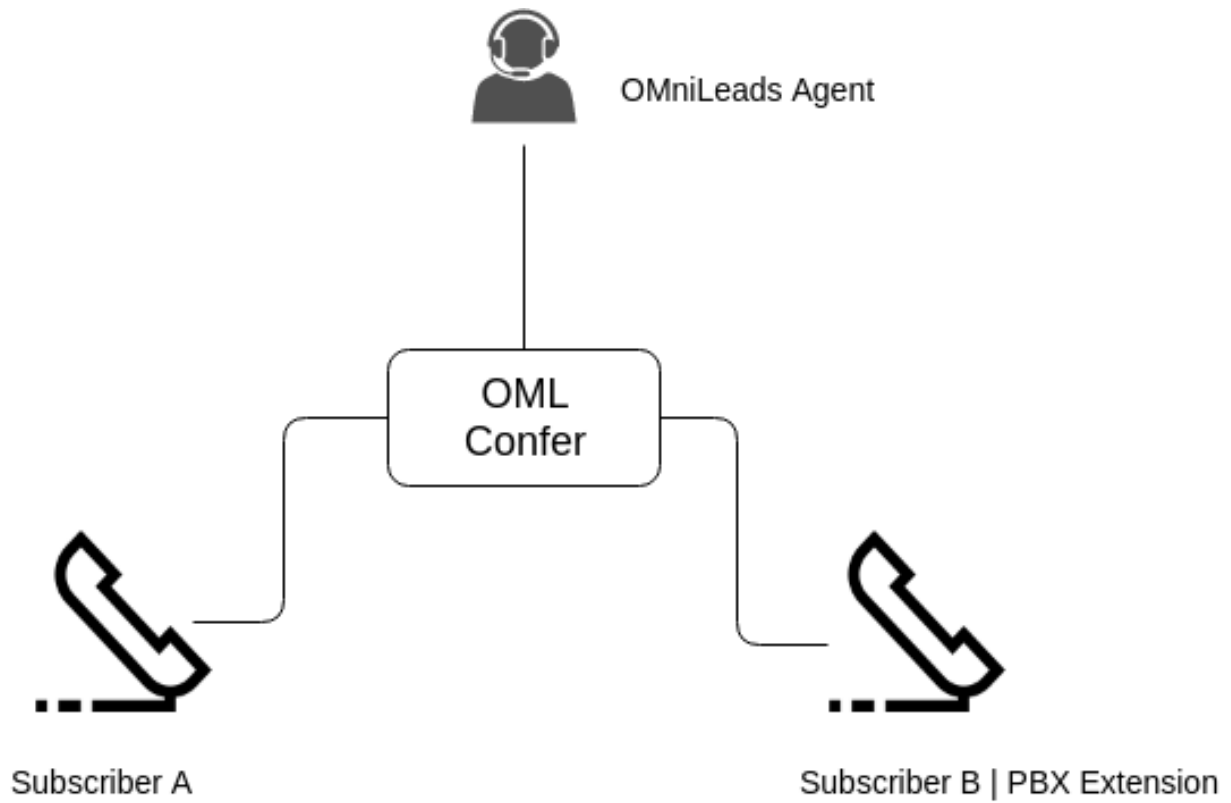
In this scenario, another possibility is:

- That it is not possible to contact the “external telephone B”, then the “agent A” can cancel the transfer during the ringing phase to the “external phone B” using the “Cancel Transfer” button on the webphone.
- The contact with “external telephone B” is achieved but this one cannot (or does not want to) proceed with the transfer, therefore the “external telephone B” hangs up the call and the “agent A” remains connected to “external telephone A”.

Three Way Conference: external telephone A, Agent, external telephone B

Under this scenario the “agent A” can set up a three-way conference between the “external number A”, that is the person who initially established the call with “agent A” and an “external number B”, that can be the extension of a PBX or a subscriber of the PSTN. In such a way, all the parties remain in a conference room.

In order to perform this action, the “agent A” must initiate a *consultative transfer* to the “external number B” and once in call the agent must click on the “Confer ” button of the webphone (Figure 9).



Three Way Conference: external telephone A, Agent, external telephone B

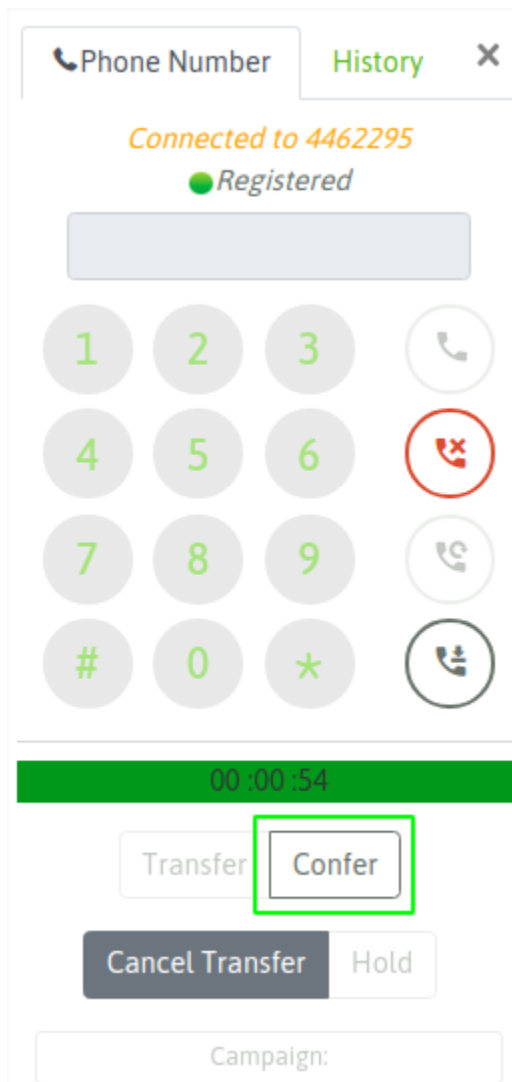


Figure 11: Webphone Conference Button”

Transfer to another Campaign

Under this scenario the “agent A” is in an active call and wants to transfer the call to an Inbound Campaign. Agent must select “blind transfer ” option since the call is transferred to the queue of the target campaign.

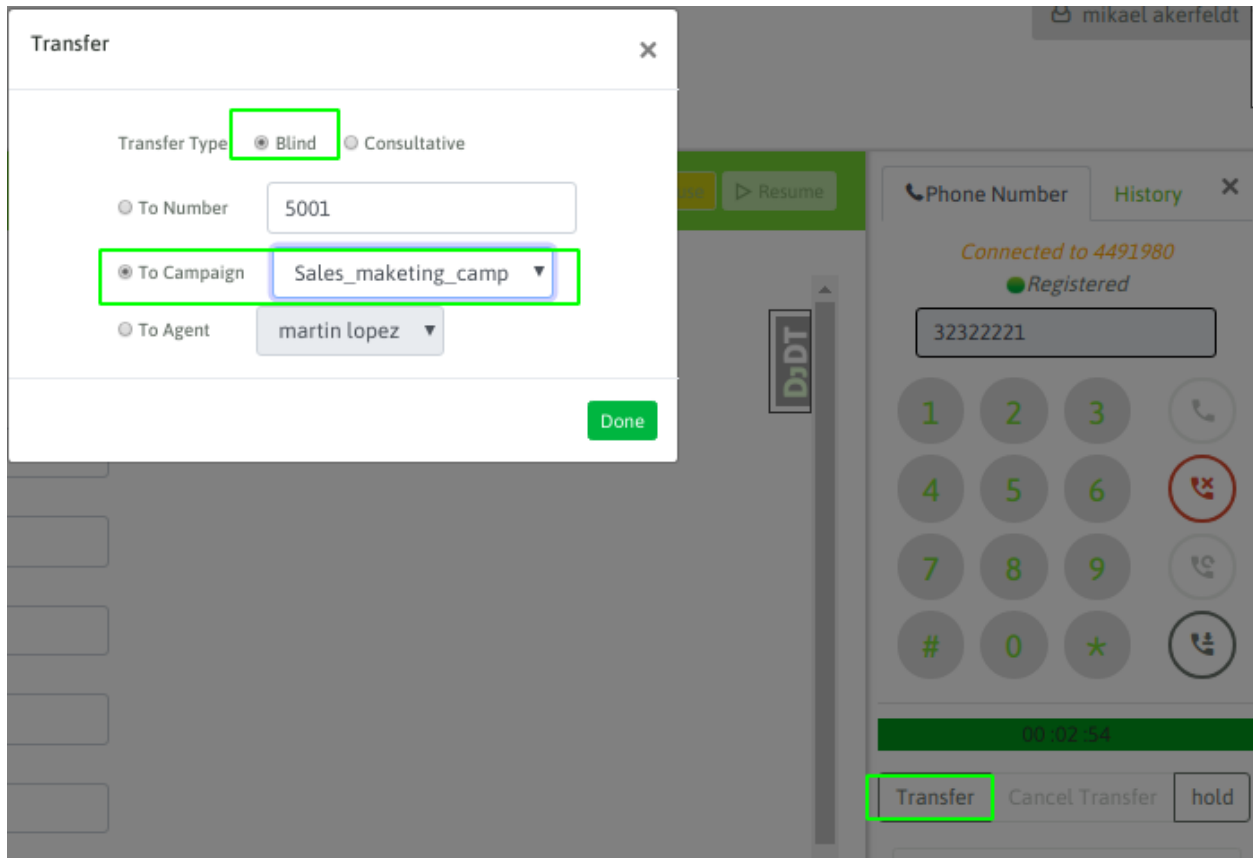
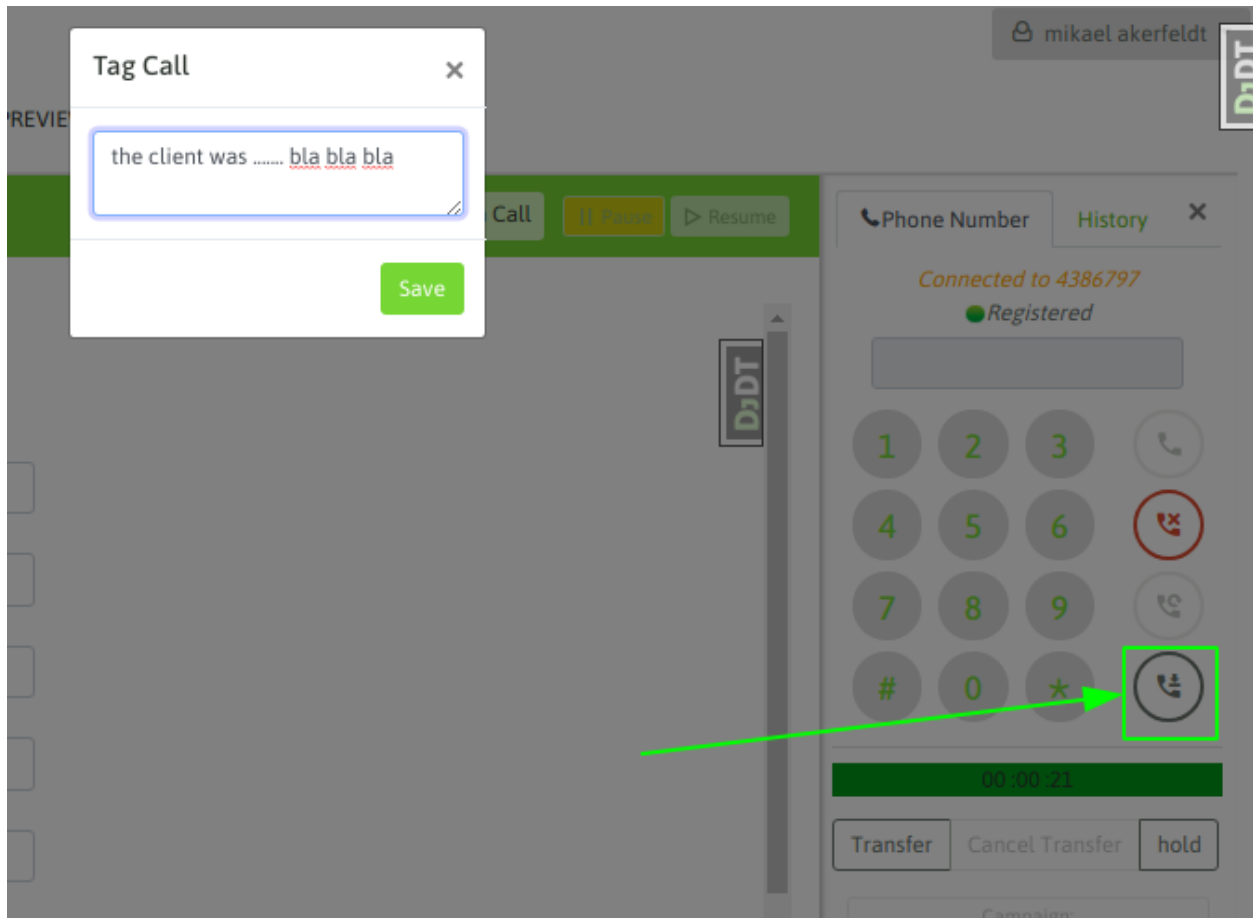


Figure 12: Transfer from “agent A” to “campana entrante”

Como se trata de una transferencia directa, la llamada automáticamente es despachada hacia el teléfono destino, quedando liberado el webphone del “agente A”. Una vez disparada ésta transferencia, no se puede volver a recuperar la llamada original, ni tampoco el “agente A” puede conocer si la llamada fue atendida o no.

Tag a Recording Call

This functionality of the agent webphone allows to generate a mark on a recording call. The idea for this flag is to be reachable from Recording Module.



Tag a Recording Call

As shown in figure 12, after clicking on the button to mark the call, a text field is displayed so that the agent can describe the situation or add a specific comment.

Finally, using the OMniLeads recording module, you can recover that recording and see what the agent has written on it.

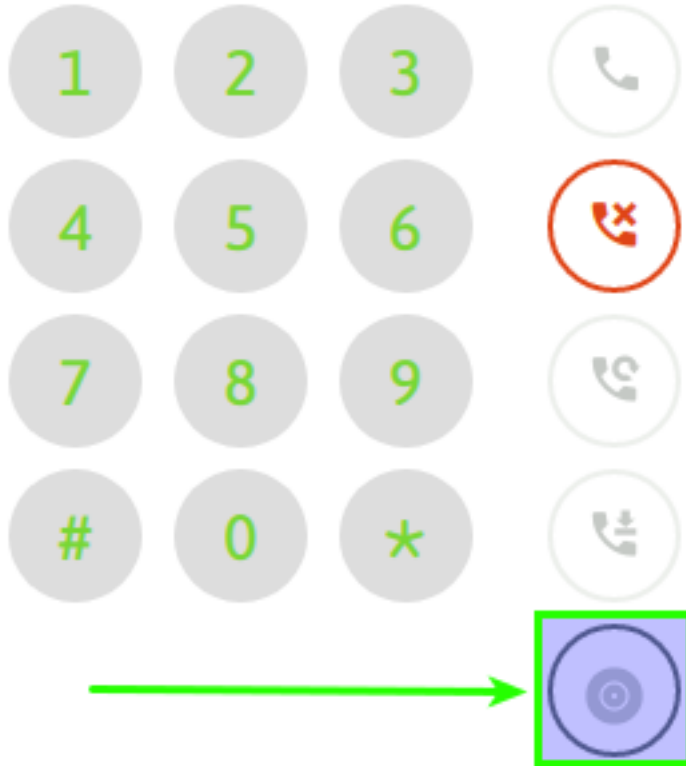
On demand call recording

This feature allows agents to record a call on demand, since clicked the button shown in next picture. This feature is only available for calls related to campaigns with recording calls option disabled

Connected to 342121121111

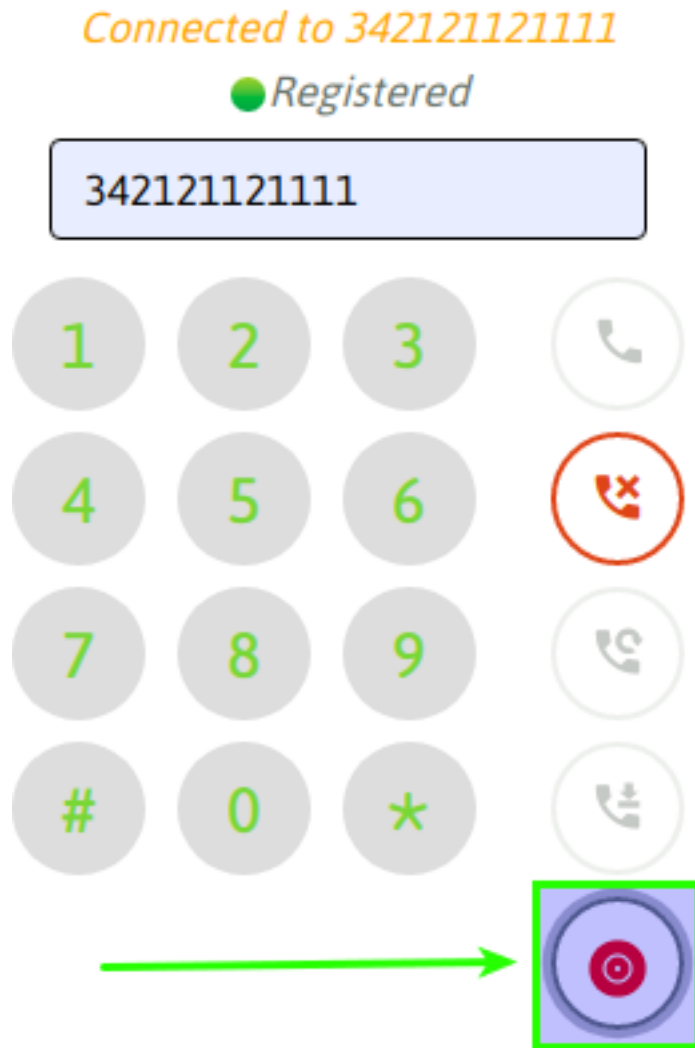
● *Registered*

342121121111



This feature allows agents to record a call on demand, since clicked the button shown in next picture. This feature is only available for calls related to campaigns with recording calls option disabled

The call recording can be stopped if user want it, clicking the same button in the middle of the recording, if the user cannot do that the call will be recorded until the end of the call.



Scheduled Callbacks

Scheduled Callback functionality allows the system to call again a specific contact based on time and date. The idea is not to discard it, but keep managing it.

Personal Callback

When the agent requires calling a specific contact again, she/he is able to generate a reminder in the personal agenda, and then listing them in order to call them back.

Agenda is a default call disposition of the system.



CONTACTS ▾ AGENDA CALL DISPOSITIONS PREVIEW CAMPAIGNS

Operation: 00:00:26 Pause 00:00:00

On Call

Pause

Resume

name:

Terence

surname:

McKenna

id_contact:

30000001

tel2:

33213211

33213211

tel3:

5343431

5343431

Call Disposition:

Agenda

Comments:

D3DT

Figure 14: Personal Callback

After call is dispositioned, a form is displayed for selecting the date, time and reason of the agenda being in place.

Agenda for contact with phone number 4553101

Date:

2019-04-10

Time:

18:13

Type:

PERSONAL ▼

Comments:

llamar nuevamente para conversar cuando esta persona se encuentre fuera de su trabajo.

Back to Call Disposition

Save

Figure 15: Personal Callback Details

Finally, the entry in the Personal Agenda remains available and can be seen by clicking in the Agenda menu.

CONTACTS ▾ **AGENDA** CALL DISPOSITIONS PREVIEW CAMPAIGNS

Operation: 00:09:43 Pause 00:00:21

Connected

|| Pause

▶ Resume

🔍 Search

10/04/2019 - 10/04/2019

Filter using this range

Contact	Date	Time	Type	Comments
4553101	April 10, 2019	6:13 p.m.	PERSONAL	llamar nuevamente para conversar cuando esta persona se encuentre fuera de su trabajo.

Figure 25: Agenda details

Global Callbacks for Outbound Dialer Campaigns

Global Callback is only applicable to Outbound Dialer Campaigns as it is intended to re-insert the contact within the list of numbers to be called by the dialer. In this scenario the dialer simply calls the scheduled number.

This is a functionality that allows agents to schedule a contact, since contact wants agent to call again later, maybe because an impossibility to attend the call in that moment.

To create a Global Callback, you must disposition the call by selecting “Agenda” Call Disposition. After that, agent must select “Global” as the type of agenda.

14.1.7 Agendas, Call Dispositions and Contacts

Agent has different menus to work with campaign contacts.

Campaign Contact List

The agent can list all the contacts of each campaign to which it is assigned. This is achieved by entering the menu item “Contact - Contact list”.

A contact view is displayed, where agent is able to select the campaign to look for contacts.

CONTACTS

Operation: 00:07:53 Pause 00:00:00 Connected Pause Play

CONTACTS

Search

Choose a campaign:

Survey_preview_2

Show contacts

Campaign Contact List

By listing all contacts in the campaign, the agent can go through each of them or perform a search by contact id, telephone, name, surname, etc.

Campaign Contacts: Survey_preview_2

Total contacts 14

Contacts without disposition 13

Dispositioned Contacts 1

Show 10 entries

Search:

Contact ID	Phone Number	Actions
3	4553103	 
5	4553105	 
8	5553103	 
10	5553105	 
14	6553104	 
6	5553101	 
4	4553104	 
11	6553101	 

Figure 2: Contacts Search

For example, agent can search for a phone number in the list of contacts, as shown in figure 3.

Campaign Contacts: Survey_preview_2



Total contacts 14

Contacts without disposition 13

Dispositioned Contacts 1

Show 10 entries

Search:

Contact ID	Phone Number	Actions
10	5553105	 

Showing 1 to 1 of 1 entries (filtering from a total of 14 contactos)

Last 1 Next

Figure 3: Contacts Search by Number

The view allows to edit/modify any of the contacts listed.

telefono:	5553105
name:	Facundo
surname:	Cabral
id_contact:	30000010

Save

Figure 4: Contacts Search

Pending Agendas

The agent can access her/his phonebook of pending calls. This section lists all the entries that the agent has registered during operation.

CONTACTS
AGENDA
CALL DISPOSITIONS
PREVIEW CAMPAIGNS

Operation: 00:55:01
Pause 00:00:00
Connected
Pause
Reconnect

Search

10/04/2019 - 10/04/2019

Filter using this range

Contact	Date	Time	Type	Comments
4553103	March 13, 2019	12:24 p.m.	PERSONAL	the client

Figure 5: Agenda

The agent has each scheduled contact and its description. By clicking on the phone number, it automatically triggers a dial attempt to the contact's phone.

Agent Call Dispositions

In this menu, the agent can list all the dispositioned calls at a historical level. Therefore the agent counts on with a backward control of each managed contact.

10/04/2019 - 10/04/2019

Filter

D3DT

Phone Number	Data	Engaged	Call Disposition	Comments	Actions
4149050	["5149090", "6149010", "Isaac", "Luria", "1000012"]	×	Busy		Options ▾
4149040	["5149092", "6149047", "Herman", "Hesse", "1000005"]	✓	Survey		Options ▾
4149050	["5149090", "6149087", "Facundo", "Cabral", "1000009"]	×	Busy		Options ▾
4149055	["5149095", "6149000", "Carl Gustav", "Jung", "1000011"]	×	No answer		Options ▾

Agent Call Dispositions

The search process can be filtered by date and the agent can also access to the selected contact to review their data or modify the call disposition.

4149050	["5149090", "6149010", "Isaac", "Luria", "1000012"]	×	Busy	Options ▾
4149040	["5149092", "6149047", "Herman", "Hesse", "1000005"]	✓	Survey	Options ▾
4149050	["5149090", "6149087", "Facundo", "Cabral", "1000009"]	×	Busy	<div> Call Disposition Engaged </div>
4149055	["5149095", "6149000", "Carl Gustav", "Jung", "1000011"]	×	No answer	Options ▾
4149050	["5149092", "6149067", "Jiddu", "Krishnamurti", "1000012"]	✓	Survey on demand	Options ▾

Agent Call Dispositions

As you can see, the agent can modify the call disposition of a contact, and the form data in the case the contact is dispositioned with a call disposition of type “Engaged”.

Agent’s recordings search

In this menú the agent can search for the recordings of his calls.

CONTACTS ▾
AGENDA
CALL DISPOSITIONS
PREVIEW CAMPAIGNS
RECORDINGS

Operation: 00:00:47
Pause 00:00:00
Connected
Pause

SEARCH RECORDING

Search

Date:
Call type:

Phone Number:
Call ID:

Figure 8: Call Recording List

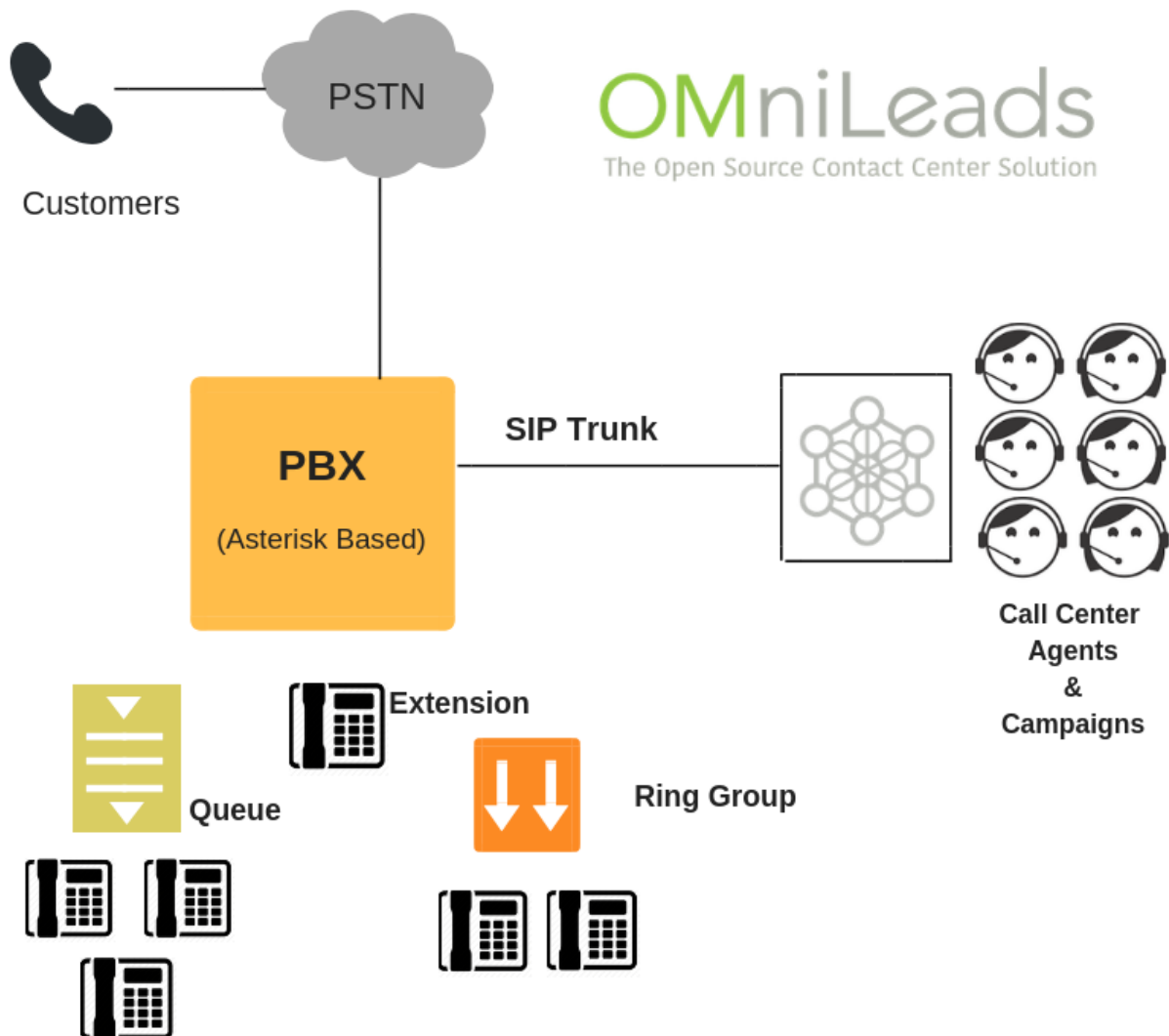
REcordings can be filtered by date, call type, client phone number, call ID, campaign, if the call was tagged, if it has call disposition and call duration

EASY PBX INTEGRATION

Making a few configurations you can establish a complete integration between OMniLeads and any PBX. In this section you can see examples of the configuration necessary to integrate the PBX and OMniLeads Contact Center that live together in same host.

15.1 Integration between OMniLeads & PBXs

Vamos a abordar éste ejemplo, utilizando Issabel-PBX (un proyecto de software libre bien conocido). Sin embargo, todo lo expuesto aquí puede ser extrapolado como configuración para cualquier PBX con soporte SIP:



Note: The steps described in this section are applicable both to the scheme where OMniLeads is on one exclusive host and the PBX on another, as well as to the case where OMniLeads runs in Docker coexisting within the same PBX host.

15.1.1 SIP trunk configuration on PBX

We select the creation of a new SIP trunk y complete the configuration with the following parameters

- En caso de tener OMniLeads en un host y la IP-PBX en otro host dentro de la red LAN:

```

type=friend
host=XXX.XXX.XXX.OML
port=5161
disallow=all
allow=alaw
qualify=yes
secret=omnileads
fromuser=issabel
defaultuser=issabel
context=from-internal

```

- In case you have OMniLeads in the Cloud and the IPPBX in another Host within the LAN network.-

```

type=friend
host=XXX.XXX.XXX.OML
port=5162
disallow=all
allow=alaw
qualify=yes
secret=omnileads
fromuser=issabel
defaultuser=issabel
context=from-internal

```

- In case of executing OMniLeads with Docker inside of IPPBX base operating system

```

type=friend
host=XXX.XXX.XXX.PBX
port=5163
disallow=all
allow=alaw
qualify=yes
secret=issabelOML
fromuser=issabel
defaultuser=issabel
context=from-internal

```

Note that the only thing that changes between the different possibilities is the port parameter. This is related to the fact that in OML a SIP port is used for each type of scenario: LAN, NAT cloud or Docker.

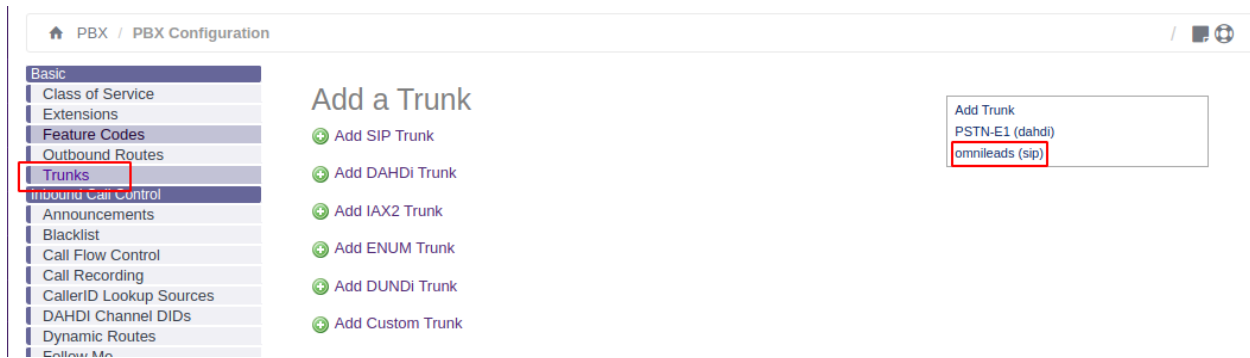
Trunk Name  : omnileads

PEER Details  :

```

type=friend
host=192.168.95.220
port=5161
disallow=all
allow=alaw
qualify=yes
secret=omnileads
defaultuser=issabel
context=from-internal

```



Once our SIP trunk is available, we go to check accessibility using the IPPBX Asterisk IPPBX

```
asterisk -rx 'sip show peers'
```

We should observe OK in the output line corresponding to the new SIP trunk, either with port 5161, 5162 or 5163.

```
[root@issabel ~]# asterisk -rx 'sip show peers'
```

Name/username	Host	Dyn	Forcerport	Comedia	ACL	Port	Status
1001	(Unspecified)	D	No	No	A	0	UNKNOWN
omnileads/issabel	192.168.95.239	Auto	(No)	No		5161	OK (1 ms)

15.1.2 SIP trunk configuration on OMniLeads

Once generated the SIP trunk on IPPBX side, we proceed with the generation with its corresponding part on OMniLeads

- In case of having OMniLeads in a Host and the IPPBX in another Host within the LAN network

```
type=wizard
transport=trunk-transport
accepts_registrations=no
sends_auth=yes
sends_registrations=no
accepts_auth=yes
endpoint/rtp_symmetric=no
endpoint/force_rport=no
endpoint/rewrite_contact=no
endpoint/timers=yes
aor/qualify_frequency=60
endpoint/allow=alaw,ulaw
endpoint/dtmf_mode=rfc4733
endpoint/context=from-pbx
remote_hosts=XXX.XXX.XXX.PBX:5060
inbound_auth/username=issabel
inbound_auth/password=issabelOML
outbound_auth/username=omnileads
outbound_auth/password=issabelOML
```

- In case you have OMniLeads in the Cloud and the IPPBX in another Host within the LAN network.

```
type=wizard
transport=trunk-nat-transport
accepts_registrations=no
sends_auth=yes
sends_registrations=no
accepts_auth=yes
```

(continues on next page)

(continued from previous page)

```

endpoint/rtp_symmetric=yes
endpoint/force_rport=yes
endpoint/rewrite_contact=no
endpoint/timers=yes
aor/qualify_frequency=60
endpoint/allow=alaw,ulaw
endpoint/dtmf_mode=rfc4733
endpoint/context=from-pbx
remote_hosts=XXX.XXX.XXX.PBX:5060
inbound_auth/username=issabel
inbound_auth/password=issabelOML
outbound_auth/username=omnileads
outbound_auth/password=issabelOML

```

- In case of execution of OMniLeas with Docker inside of the IPPBX base operating system, we used:

```

type=wizard
transport=trunk-nat-docker-transport
accepts_registrations=no
sends_auth=yes
sends_registrations=no
accepts_auth=yes
endpoint/rtp_symmetric=yes
endpoint/force_rport=yes
endpoint/rewrite_contact=yes
endpoint/timers=yes
aor/qualify_frequency=60
endpoint/allow=alaw,ulaw
endpoint/dtmf_mode=rfc4733
endpoint/context=from-pbx
endpoint/rtp_symmetric=yes
remote_hosts=XXX.XXX.XXX.PBX:5060
inbound_auth/username=issabel
inbound_auth/password=issabelOML
outbound_auth/username=omnileads
outbound_auth/password=issabelOML

```

Once effective our trunk, we pass to check if Issable is accessible from OMniLeads, using OMniLeads Asterisk CLI.

```
asterisk -rx 'pjsip show endpoints'
```

Note: If we are executing OMniLeads in Docker, for access the container executing we must execute the following command: **docker exec -it oml-asterisk-prodenv**, then we can start the CLI

The command output should be similar to the figure:

```

root@5466a47b7e48:/etc/asterisk# asterisk -rx 'pjsip show endpoints'

Endpoint: <Endpoint/CID.....> <State.....> <Channels.>
I/OAuth: <AuthId/UserName.....>
Aor: <Aor.....> <MaxContact>
Contact: <Aor/ContactUri.....> <Hash.....> <Status> <RTT(ms).....>
Transport: <TransportId.....> <Type> <cos> <tos> <BindAddress.....>
Identify: <Identify/Endpoint.....>
Match: <criteria.....>
Channel: <ChannelId.....> <State.....> <Time.....>
Exten: <DialedExten.....> CLCID: <ConnectedLineCID.....>
=====

Endpoint: 1006/1006 Unavailable 0 of 1
Aor: 1006 2
Transport: agent-transport udp 0 0 0.0.0.0:5160

Endpoint: 1007/1007 Unavailable 0 of 1
Aor: 1007 2
Transport: agent-transport udp 0 0 0.0.0.0:5160

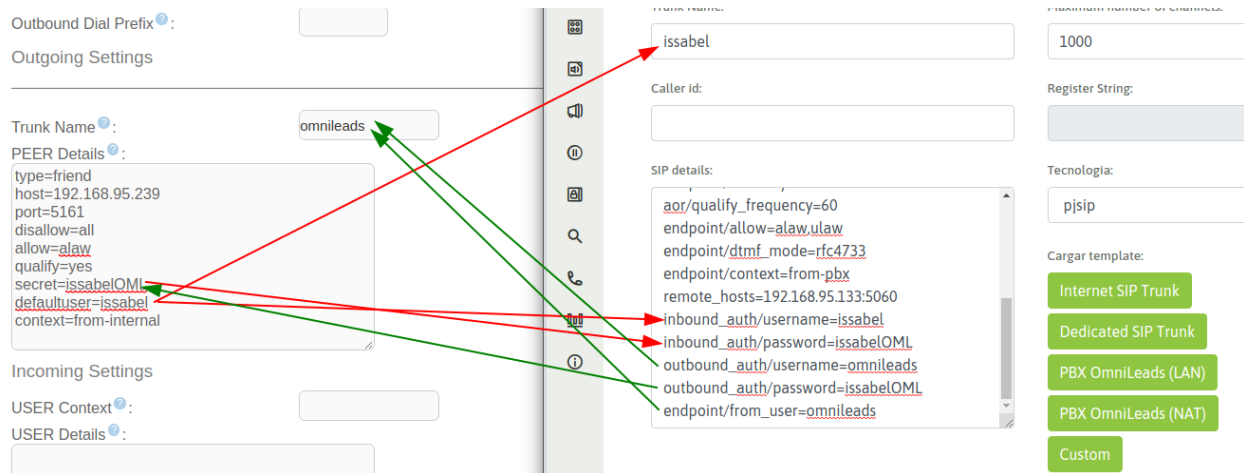
Endpoint: issabel Not in use 0 of inf
OutAuth: issabel-oauth/omnileads
Aor: issabel 0
Contact: issabel/sip:192.168.95.133:5060 c971541761 Avail 0.775
Transport: trunk-nat-docker-transport udp 0 0 0.0.0.0:5163
Identify: issabel-identify/issabel
Match: 192.168.95.133/32

```

At this point, exists a SIP trunk between bot phone systems, pending the calls routing configuration between both systems

Finally we emphasize on make relations between parameters on Issabel and OMniLeads SIP trunks

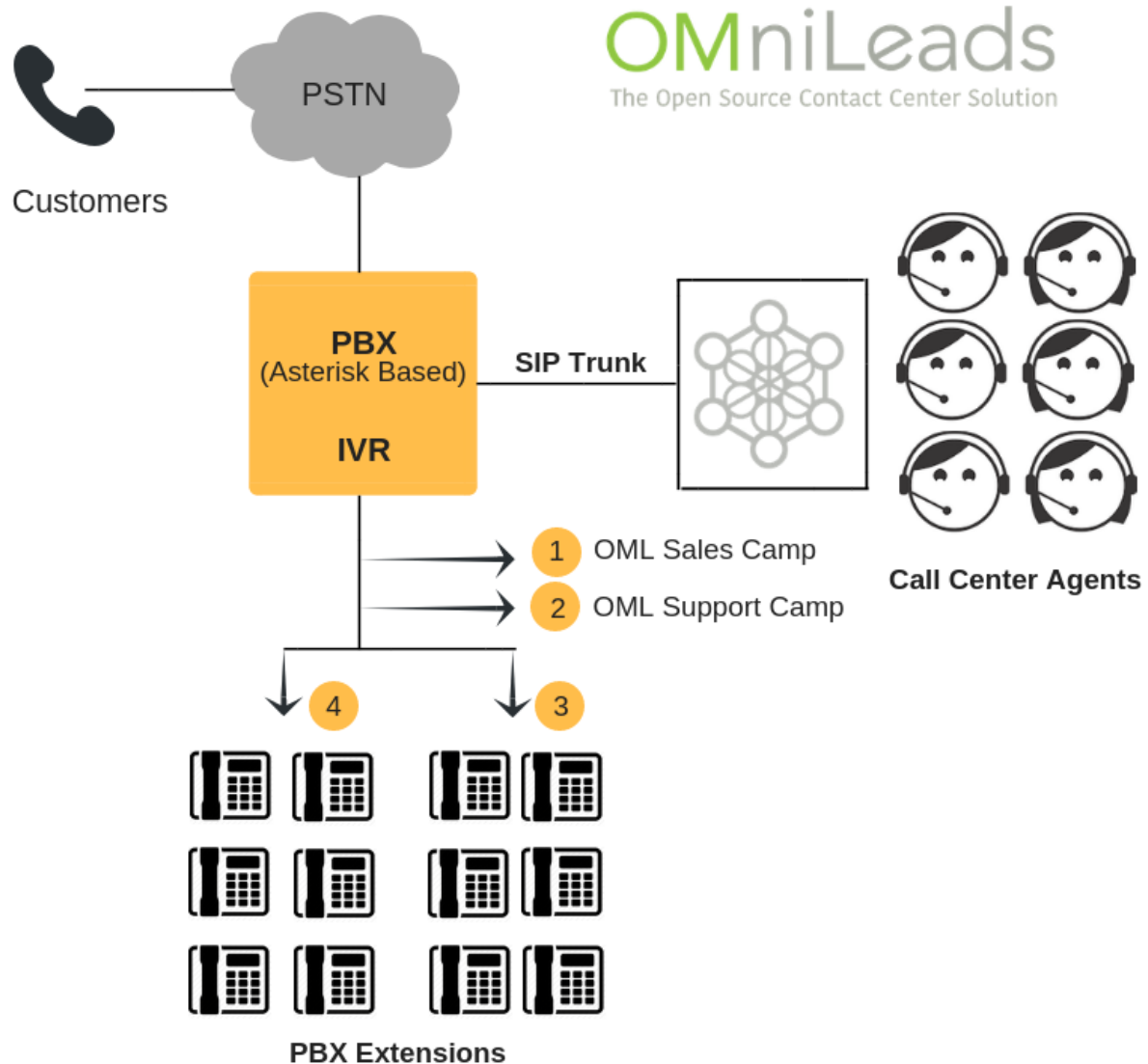
A picture is worth a thousand words:



15.1.3 How send calls from IP-PBX to OMniLeads

Now we exposed a way to connect the IP-PBX resources (inbound routes, IVRs, announcements, extensions, etc.) with OMniLeads. It means, that, for example from one option of the company main IVR can derive to an OMniLeads inbound campaign, or that one extension can contact or transfer a call to an OMniLeads inbound campaign or OMniLeads agent.

This is completely viable using the IP-PBX *custom extensions*, in our example case: Issabel-PBX



15.1.4 Calls to OMniLeads inbound routes



Now we present the example where the user want to create a *custom extension* in which, when call it from another extension or invoke it from some PBX object (IVR, inbound route, announcement, etc) create a channel against OMniLeads, particularly pointing to an *inbound route* which can at the same time, send the call to an inbound campaign.

For one side, we have an inbound route in OMniLeads, pointing for instance, to an inbound campaign:

Name:	DID number:
<input type="text" value="camp1_from_pbx"/>	<input type="text" value="098098"/>
Caller id prefix:	Language:
<input type="text"/>	<input type="text" value="English"/>
Destination type:	Destination:
<input type="text" value="Inbound campaign"/>	<input type="text" value="Inbound campaign: in1"/>
<input type="button" value="Save"/>	

Having in mind that the DID choosen was *098098*, in the IPPBX we must generate an *extension* with type *custom*, where the *Dial* strings should point to the OMniLeads SIP trunk, and the sent number should be *098098*.

Extension: **098098** (1)

 Delete Extension 098098
 Add Follow Me Settings
 Add Gabcast Settings
 - Edit Extension

Display Name ?
 CID Num Alias ?
 SIP Alias ?
 +Extension Options

+Assigned DID/CID

- Device Options

This device uses custom technology.

dial ?

(2) (3)

In the figure we mark three elements:

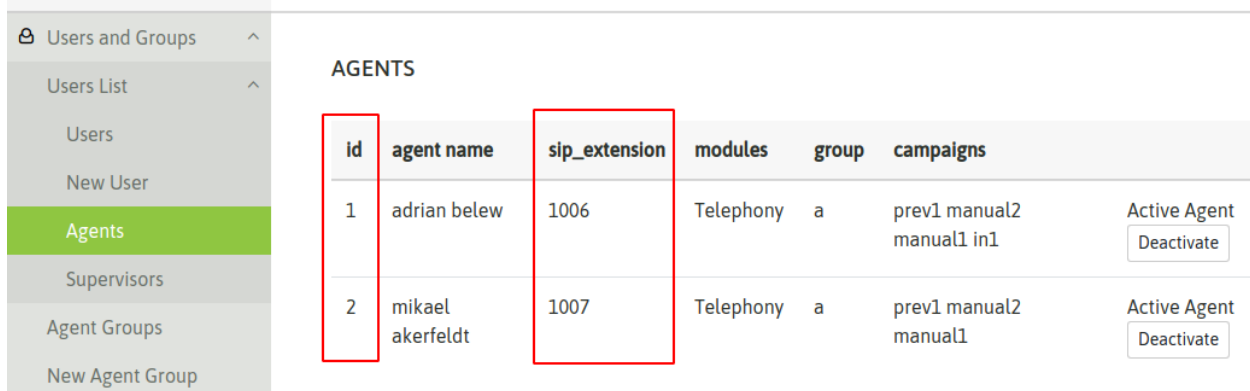
- (1) The extension number, could be different to the number sent to OMniLeads(3) . It can be every number, only making sure the Dial extension from the custom extension matches with the DID number from the OML inbound route (098098 for our example)
- (2) The *trunk* where the custom extension points. This value must match with the field *Trunk Name* on the SIP trunk against OMniLeads generated on the IP-IPBX
- (3) The number to send by the trunk must match with the DID on the OMniLeads inbound route

This way every IPPBX extension could call or transfer a call to this *custom extension* and will be sent to the related inbound route on OMniLeads for finally connect over an inbound campaign or the assigned element as destination of the OMniLeads inbound route

As a final mention, we want to make clear that we can have in the IPPBX many custom extensions pointing to different OMniLeads inbound routes as we like !

15.1.5 Calls to OMniLeads agents

From the figure, let's take the agent *Adrian Belew*. Note that its ID is 1 and its SIP number is 1006. For that reason at the moment to make the number to send in the Dial string of the IPPBX custom extension we must concatenate the *SIP Number* with its *Agent ID*; in our example will be **10061** for agent *Adrian Belew* and *10072* for agent *Mikael Akerfeldt*



The screenshot shows a web interface for managing agents. On the left is a sidebar menu with options: Users and Groups, Users List, Users, New User, Agents (highlighted in green), Supervisors, Agent Groups, and New Agent Group. The main area is titled 'AGENTS' and contains a table with the following data:

id	agent name	sip_extension	modules	group	campaigns	
1	adrian belew	1006	Telephony	a	prev1 manual2 manual1 in1	Active Agent Deactivate
2	mikael akerfeldt	1007	Telephony	a	prev1 manual2 manual1	Active Agent Deactivate


When generating the configuration in the PBX to be able to send calls to the agents, we have two alternatives

- (1) 1- Use an outgoing route from the PBX to OMniLeads as indicated in the following figure


Edit Route


 Delete Route to_oml


Route Settings


Route Name 

to_oml


Route CID: 

☐ Override Extension 


Route Password: 

Route Type: 


☐ Emergency ☐ Intra-Company

Music On Hold? 

default ▾

Time Group: 

---Permanent Route--- ▾

Route Position 

---No Change--- ▾

Additional Settings


Call Recording 


Allow ▾

PIN Set 

None ▾

Dial Patterns that will use this Route

+ | / 

+ | / 



+ Add More Dial Pattern Fields

In this case, any extension of the PBX can generate a call to an agent by dialing the combination mentioned in the previous paragraph.

- (2) Generate a custom extensions for each OML agent, that is, the Dial chain of the custom extension will no longer be made up of an OMniLeads incoming route DID as it was in the case of linking with incoming routes, but it will be a combination of the ID of the agent and his SIP number.

As indicated in the figure

Extension: **9001** ⁽¹⁾

 Delete Extension 9001
 Add Follow Me Settings
Add Gabcast Settings
- Edit Extension

Display Name [?]

CID Num Alias [?]

SIP Alias [?]

+Extension Options

+Assigned DID/CID

- Device Options

This device uses custom technology ⁽²⁾
dial [?] ⁽³⁾

In the figure we mark three elements:

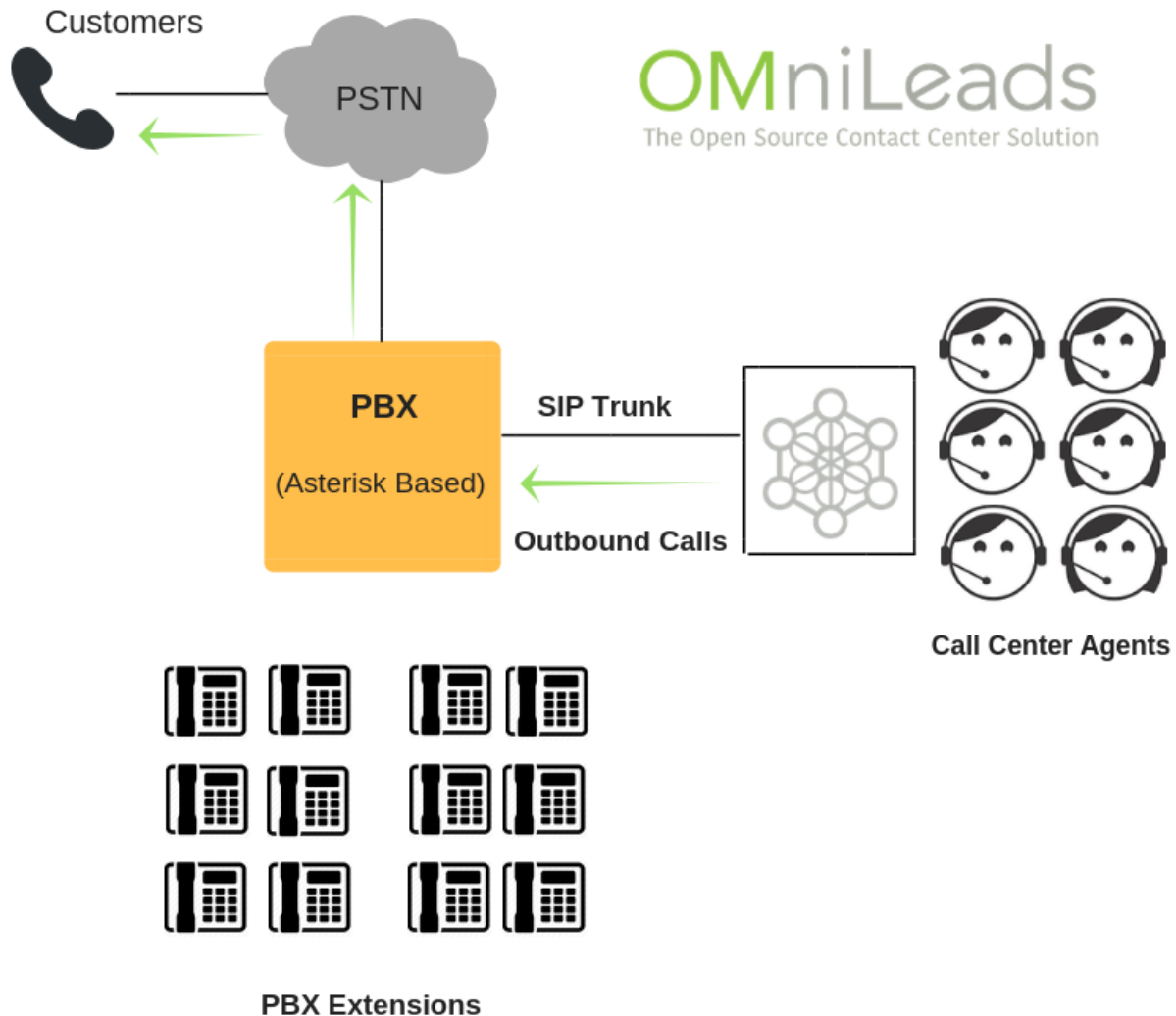
- (1) The extension number, could be different to the number sent to OMniLeads(3) . It can be every number, only making sure the Dial extension from the custom extension matches with the concatenation of the ID agent and its SIP number (10061 for our example)
- (2) The *trunk* where the custom extension points. This value must match with the field *Trunk Name* on the SIP trunk against OMniLeads generated on the IP-IPBX
- (3) The number to send by the trunk must match with the concatenation of the ID agent and its SIP number (10061 for our example)

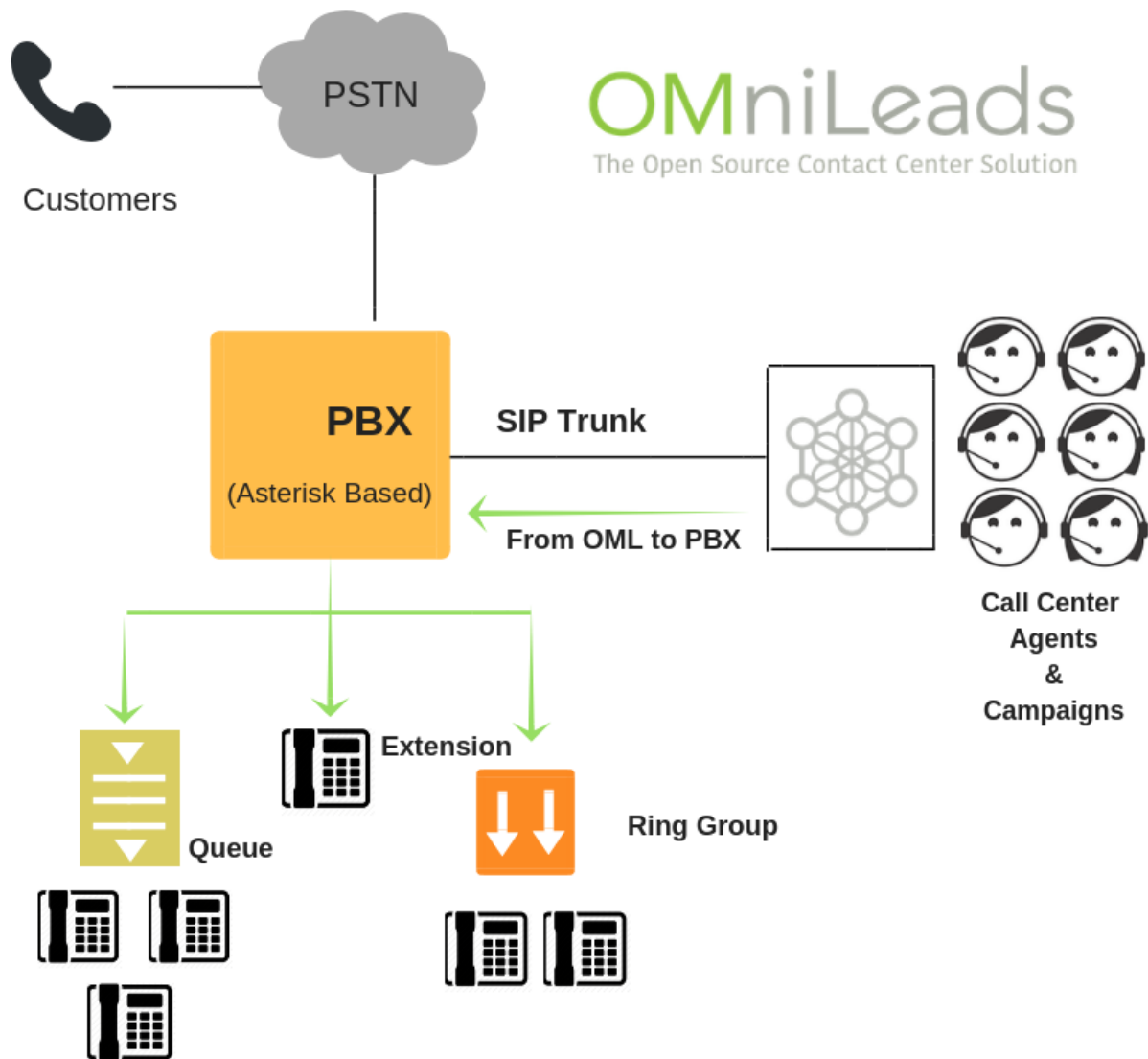
We must repeat the procedure for every OMniLeads agent that wants to customize their extension in the PBX

Either of the two alternatives are viable and you will get the desired result.

15.1.6 Calls from OMniLeads to the PSTN and the IPPBX resources

Finally we are going to generate the outbound routing inside OMniLeads that allow agents and dials raise calls to the PSTN, and , at the same time allow agents to call or transfer calls to the IPPBX resources like extensions, ring, groups, queues calls, etc





Simply we must add a new outbound route that points to the trunk to the IPPBX

Name:

all

Ring time:

25

In seconds

Dial options:

Tt

Dial patterns

Prepend:	Prefix:	Dial pattern:	
		XXXXXXX	Remove
Prepend:	Prefix:	Dial pattern:	
		XXXXXXXX	Remove
Prepend:	Prefix:	Dial pattern:	
	9	1XXX	Remove
Prepend:	Prefix:	Dial pattern:	
		XXXXXXXXXX	Remove
Prepend:	Prefix:	Dial pattern:	
		XXXXXXXXXXXX	Remove

Add dial pattern

Trunks sequence

issabel

Add trunk

This way the integration is completely functional and both systems can make all types of calls and interactions

IT ADMINISTRATOR MANAGEMENT

In this chapter is covered some tasks made from the IT administrator of OMniLeads. Referring to: configuration of the dialer platform, upgrade of software management, backup & restore and network parameters change.

16.1 IT administrator managment

16.1.1 Environment variables

Through this section is going to be commented the procedures that needs the use of *inventory* file. As known the file is edited bofer installation and after that it becomes default. But the variables and their values stayes as environment variables.

To check this variables open the file `/etc/profile.d/omnileads_envvars.sh`.

```
cat /etc/profile.d/omnileads_envvars.sh

AMI_USER=omnileadsami
AMI_PASSWORD=5_MeO_DMT
ASTERISK_IP=192.168.95.163
ASTERISK_HOSTNAME=localhost.localdomain
ASTERISK_LOCATION=/opt/omnileads/asterisk
CALIFICACION_REAGENDA=Agenda
DJANGO_PASS=098098ZZZ
DJANGO_SETTINGS_MODULE=ominicontacto.settings.production
EPHEMERAL_USER_TTL=28800
EXTERNAL_PORT=443
INSTALL_PREFIX=/opt/omnileads/
KAMAILIO_IP=192.168.95.163
KAMAILIO_HOSTNAME=localhost.localdomain
KAMAILIO_LOCATION=/opt/omnileads/kamailio
MONITORFORMAT=mp3
NGINX_HOSTNAME=localhost.localdomain
OMNILEADS_HOSTNAME=localhost.localdomain
```

(continues on next page)

(continued from previous page)

```
PGHOST=localhost.localdomain
PGDATABASE=omnileads
PGUSER=omnileads
PGPASSWORD=my_very_strong_pass
PYTHONPATH=$INSTALL_PREFIX
REDIS_HOSTNAME=localhost
SESSION_COOKIE_AGE=3600
TZ=America/Argentina/Cordoba
WOMBAT_HOSTNAME=localhost.localdomain
WOMBAT_USER=demoadmin
WOMBAT_PASSWORD=demo

export AMI_USER AMI_PASSWORD ASTERISK_IP ASTERISK_HOSTNAME ASTERISK_LOCATION_
↪CALIFICACION_REAGENDA DJANGO_SETTINGS_MODULE DJANGO_PASS EPHEMERAL_USER_TTL_
↪EXTERNAL_PORT INSTALL_PREFIX KAMAILIO_IP KAMAILIO_HOSTNAME KAMAILIO_LOCATION_
↪MONITORFORMAT NGINX_HOSTNAME OMNILEADS_HOSTNAME PGHOST PGDATABASE PGUSER PGPASSWORD_
↪PYTHONPATH REDIS_HOSTNAME SESSION_COOKIE_AGE TZ WOMBAT_HOSTNAME WOMBAT_USER WOMBAT_
↪PASSWORD
```

This way the administrator can see them when he/she wants

Important: Do not edit this file.

16.1.2 Configuration of the *Predictive Dialer* module

Note: First of all, we notify that if the OML instance deployed in the previous steps does not contemplate the use of campaigns with predictive outbound dialer, this step can be omitted.

OMniLeads needs a third-party tool to implement the campaigns with predictive outbound dialer. This tool is based on commercial software licenses that must be managed with the manufacturer.

In any case the system can be used with a test channel that grants as a demo. Therefore we can configure the component and run concept tests before acquiring licenses for a real operation.

If running predictive campaigns is desired, we must generate the following basic Wobal Dialer settings. To generate this configuration we must follow some steps that begin with the access to the corresponding URL.

<http://omnileads.yourdomain:8080/wombat> or <http://XXX.XXX.XXX.OML:8080/wombat>

Note: In case of running OMniLeads Dockerized the URL is:<http://XXX.XXX.XXX.OML:442/wombat>

Where XXX.XXX.XXX.OML is the IP of docker engine host

Creation of database

When enter for the first time, we must proceed with the creation of a MariaDB database that uses Wombat Dialer. Click on the highlighted button in figure 2.

WombatDialer JDBC connection tester

It looks like you don't have a working JDBC connection.

WombatDialer requires a working JDBC connection to a MariaDB database server in order to work properly.

If you have not already done so:

- Create a database for WombatDialer and manually import the sample database
- Edit the file `WEB-INF/tpf.properties` to enter the database server, user and password WombatDialer will use to connect

These steps are explained in greater detail in WombatDialer user manual, available at <http://wombatdialer.com>.

Create WombatDialer database now >>>

This is the current configuration found:

- XML configuration file located at: `/usr/local/queuemetrics/webapps/wombat-18.08.3-232/WEB-INF/web.xml`
- JDBC URI configured as: `jdbc:mariadb://127.0.0.1/wombat?user=wombat&password=dials&autoReconnect=true`

Most common JDBC errors are explained in the FAQs, available at <http://wombatdialer.com>.

Figure 1: DB create

Then is the moment to ingress root MySQL user password and then click on the button shown in figure 2.

Note: Since OMniLeads 1.6.0 version a MySQL root password is not set, so you can leave this field empty.

We proceed then with the creation of the MariaDB database that will use from now on the Wombat Dialer component.

Create the WombatDialer database

This procedure may delete all data if you already have an existing WombatDialer database. **If you already have a database, do not run this.** Parameters that are not editable come from the `tpf.properties` file. If you need to change them, go edit your `tpf.properties` file.

MySQL root information	
MySQL database address	127.0.0.1
MySQL root user	root
MySQL root password	098098ZZZ
WombatDialer database information	
QueueMetrics database	wombat
QueueMetrics database user	wombat
QueueMetrics database password	dials
<input type="button" value="Update now"/>	

Figure 2: MariaDB root password

First login

Once created the MariaDB database that Wombat Dialer uses, we proceed with the first login.

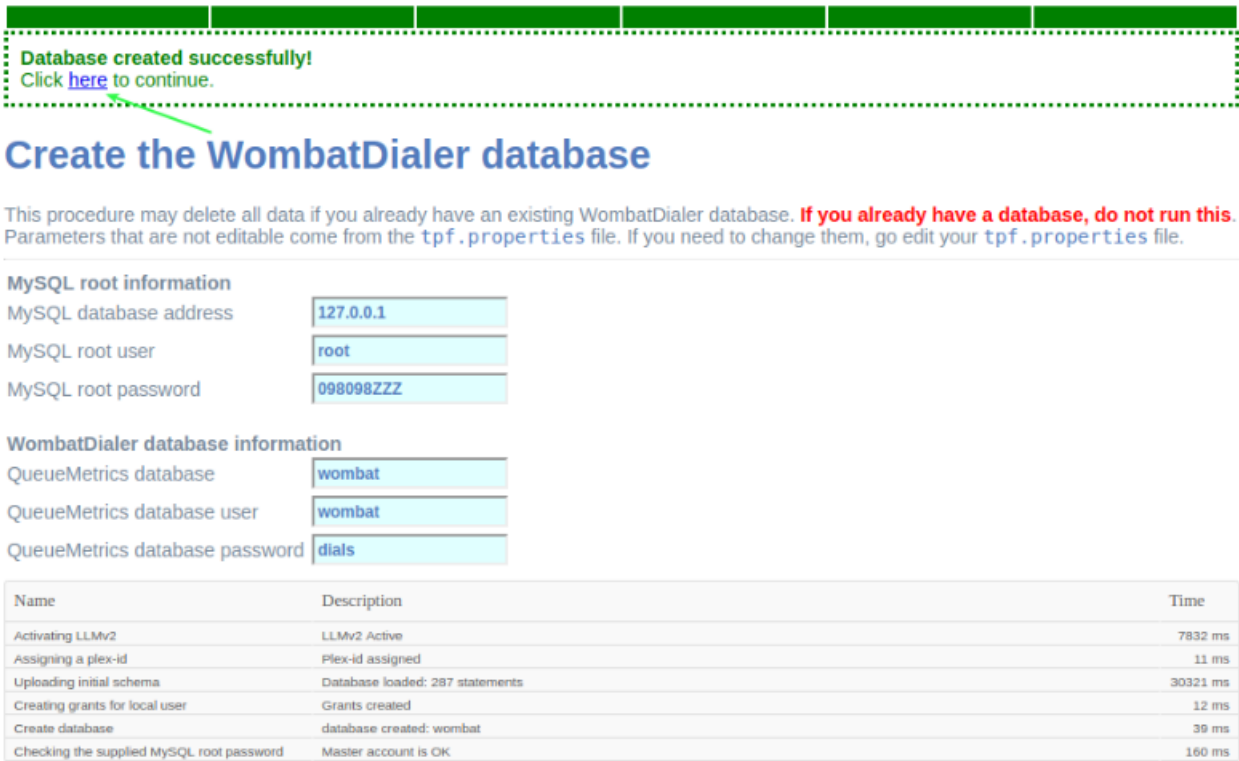


Figure 3: Login post db create

A login must be performed then in the Wombat Dialer administration interface to continue with the configuration of the necessary parameters for the OML interaction.

Upon entering a screen like the following is displayed, where we must access with the user and passwords generated within the installation.

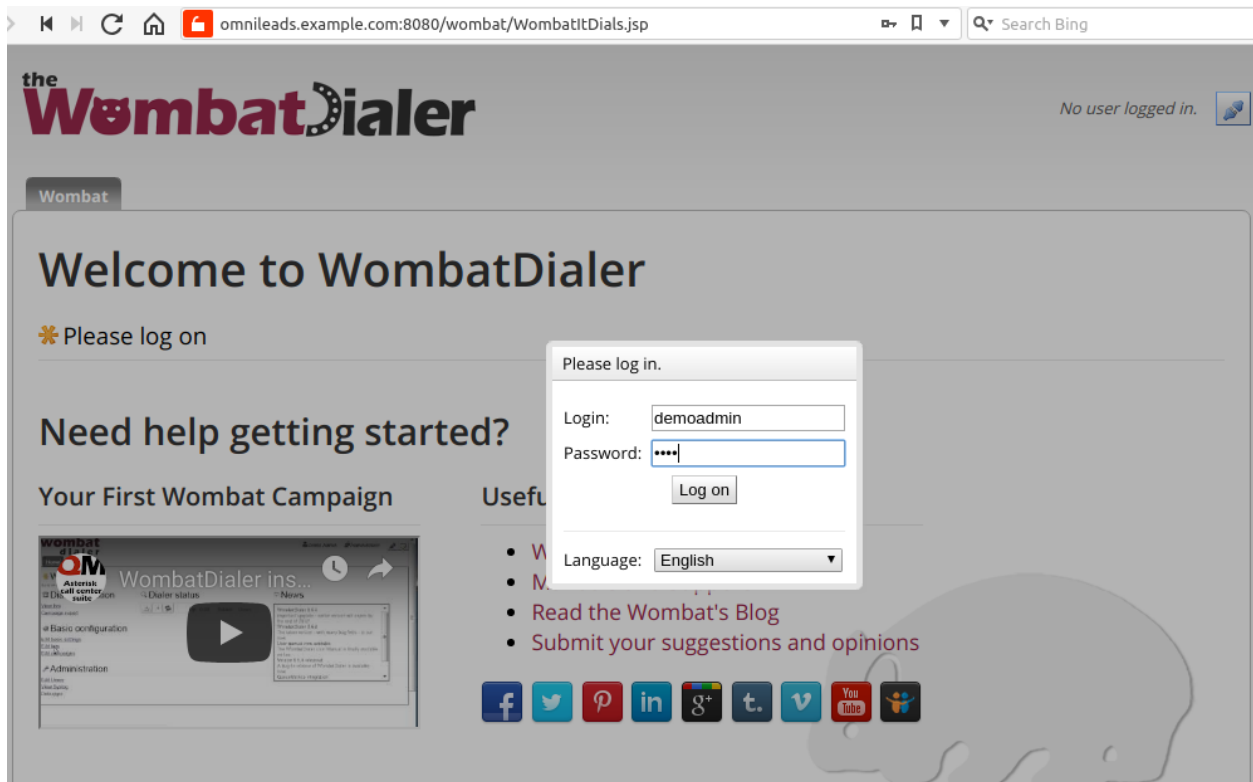


Figure 4: Access to WD

Change of web credentials

By default Wombat Dialer comes with the web credentials *demoadmin* as user and *demo* as password. These credentials can be changed:

- Firstly, the credentials must be defined in the inventory file, these are the variables **dialer_user** and **dialer_password**. See `about_install_inventory_vars`.
- Go to Wombat Dialer web, and go to Administration -> Edit users.

WombatDialer 20.02.1

Funciones del marcador

- Vista in tempo real
- Estado Marcador
- Reporte de campaña
- Copiar Campaña

Estado del marcador

Tiempo de actividad: 4:45:18 / Estado: **READY**

Marcador: READY - Rec. 8 Env. 0
 Canales con licencia: 1
 Se ruega de actualizar manualmente las licencias.

Novedades

- Ghostcat security update**
Check if your system is affected by the Ghostcat vulnerability and what to do about it.
- Introducing WombatDialer 20.02**
Black-list superpowers, effective transfers and improved GUI make outbound easier and more productive than ever.
- QueueMetrics 19.10 released**
QueueMetrics 19.10 is available now. Improved UI, new "synchronizer" module, improved security and efficiency and much more. It's time to upgrade!
- QueueMetrics 19.04 released**
QueueMetrics 19.04 is available now. Lots of new wallboard goodies and improved...

Configuración de base

- Configuraciones de base
- Editar listas
- Horarios de abertura
- Editar campañas

Administración

- Editar usuarios
- Ver log de sistema

Figure 5: WD change credentials

- There you can see the user demoadmin, click on the pencil icon, change the user in Login field with **the same user you ingressed in the variables dialer user**, for password **use the same password ingressed in dialer_password**.
- Finally click on Save button.

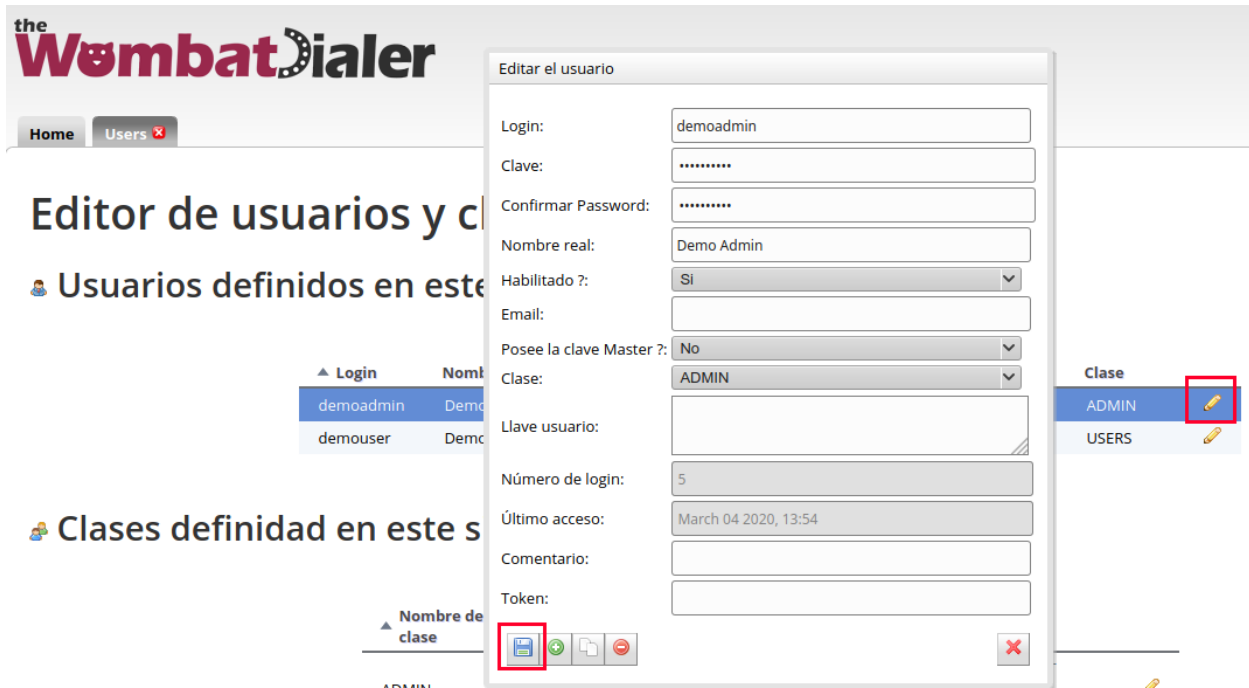


Figure 6: WD change credentials 2

- Once finished, reload the page and ingress with the new credentials.

AMI credentials

Note: From release-1.8.0 take care of this


Wombat Dialer will use different AMI credentials that OMniLeads uses, for that reason, the AMI user **wombatami** is created inside the file `oml_manager.conf`. The password for this AMI user changes depending of what inserted the user in inventory file parameter **ami_password**. By default the user comes this way:


```
[wombatami]
secret = 5_MeO_DMT
deny = 0.0.0.0/0.0.0.0
permit = 127.0.0.1/255.255.255.255
read = all
write = all
```

Basic parameters

Once inside the system, we proceed with the configuration of the two basic parameters to make the integration with OMniLeads ready. To do this we must access the “Basic configuration” menu as indicated in figure 7.


WombatDialer 18.08.1


 **Dialer operation**
[View live](#)
[Dialer Status](#)
[Campaign report](#)
[Copy Campaign](#)


 **Basic configuration**


[Edit basic settings](#)


[Edit lists](#)
[Edit Opening Hours](#)
[Edit campaigns](#)


 **Administration**
[Edit Users](#)
[View Syslog](#)

 **Dialer status**











Uptime:
54:15 /

State:
READY

 Dialer: **READY** - Rcv 5 Snt 0
Licensed channels: 1

 **News**

- Introducing WombatDialer 18.08**
Better visibility, powerful call rate controls and improved QueueMetrics integration.
- Asterisk in the Contact Centre 2018**
How happy are you with your Contact Centre? let your voice be heard - take part in our new survey.
- Official training and free webinars available for WombatDialer**
The official training and certification program is now available for WombatDialer. And we also have free weekly webinars.
- Introducing WombatDialer 16.06**
Improved GUI, stellar performance and easy QueueMetrics integration.
- Loway unveils its Referral Program**
You can earn a consistent commission when a referred customer purchases a license or training

Figure 7: WD basic config

In this menu, first of all, a new conection instance must be generated inside the *Asterisk Servers* section as exposed in figure 8.

Basic Settings

Description

asterisk

Edit Asterisk server

Server description: asterisk

Server Type: Asterisk AMI

Server address: 127.0.0.1

AMI port: 5,038

Login: omnileadsami






Password: 5_MeO_DMT


Max msg. per unit: 5

Unit length (ms): 50

Security key:


100 Messages per Second





Max msg.

dsami 5 / 50 ms





Trunk name	Asterisk server	Dial string	Calls per Second	Capacity
todo	asterisk	Local/\${num}@from-internal/n	100	100
		<input type="text"/>		

Figure 8: WD basic config - AMI Asterisk

In the next point, a Trunk is configured using an arbitrary “Trunk name”, but with the calling chain marked in figure 9. `Local/${num}@from-oml/n`

16.1. IT administrator managment

231

Figure 9: WD basic config - Asterisk Trunk

At last, remember to “play” the dialer service, as indicated in figure 10.

WombatDialer 18.08.1

Dialer operation

View live
Dialer Status
Campaign report
Copy Campaign

Basic configuration

Edit basic settings
Edit lists
Edit Opening Hours
Edit campaigns

Dialer status

News

- **Introducing WombatDialer 18.08**
Better visibility, powerful call rate controls and improved QueueMetrics integration.
- **Asterisk in the Contact Centre 2018**
How happy are you with your Contact Centre? let your voice be heard - take part in our new survey.
- **Official training and free webinars available for WombatDialer**
The official training and certification program is now available for WombatDialer. And we also have free weekly webinars.
- **Introducing WombatDialer 16.06**
Improved GUI, stellar performance and easy QueueMetrics integration.
- **Loway unveils its Referral Program**
You can earn a consistent commission when a referred customer purchases a license or training

Figure 10: WD activate

Finally the platform is enabled to manage predictive calls. The default installation has a Wombat Dialer demo license for a channel.

Backup/restore of database

You can make a backup/restore of MariaDB dialer database, executing the following commands in the host where Wombat/MariaDB is running.

To perform a Backup:

```
# mysqldump -B wombat -u root -p > $ubicacion_archivo_dump/dump.sql
Enter password:
```


Where `$ubicacion_archivo_dump` is the path where you are going to place the dump file

To make the restore, in a new MariaDB server:

```
mysql -u root -p qstats < dump.sql
```

You must have the backup file in the new server to restore the database

16.1.3 Change SSL certificates

If you want to change the SSL certificates you need to have yours in **.pem** format. Rename the files, they must be ***cert.pem** and **key.pem**. Then place them in these folders:

```
/opt/omnileads/nginx_certs/  
/opt/omnileads/kamailio/etc/certs
```

After that, restart the following services:

```
service nginx restart  
service kamailio restart
```

16.1.4 Reset admin web password

If you have forgotten the password for admin user you can reset the same to its default values (admin/admin), for that SSH into OMniLeads and execute this command:

```
/opt/omnileads/bin/manage.sh cambiar_admin_password
```

16.1.5 Backup & Restore

OMniLeads has a script to perform the Backup/restore tasks.

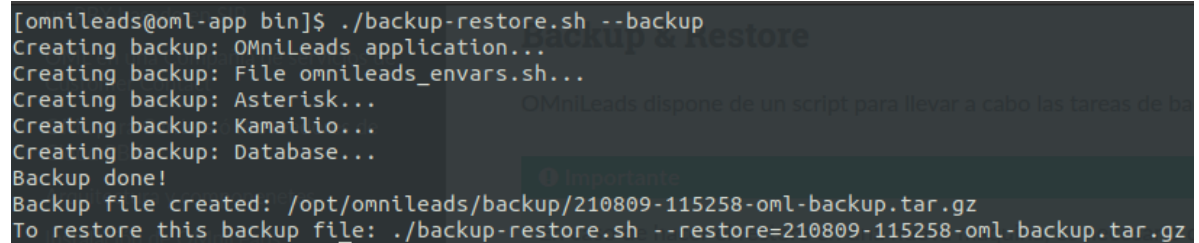
- **Backup**

To perform a Backup:

We must access the host where OMniLeads is running through ssh. Once inside the host we execute the following commands.

```
su omnileads -  
cd /opt/omnileads/bin  
./backup-restore.sh --backup
```

The execution of the script throws an output similar to the one on figure 11.



```
[omnileads@oml-app bin]$ ./backup-restore.sh --backup  
Creating backup: OMniLeads application...  
Creating backup: File omnileads_envvars.sh...  
Creating backup: Asterisk...  
Creating backup: Kamailio...  
Creating backup: Database...  
Backup done!  
Backup file created: /opt/omnileads/backup/210809-115258-oml-backup.tar.gz  
To restore this backup file: ./backup-restore.sh --restore=210809-115258-oml-backup.tar.gz
```

As it can be seen, the command output tells us how to do the restore procedure. Inside the path **/opt/omnileads/backup**, the files “.tar.gz” that contain the backups are generated.

If you don't want to make a database backup you can run script with the **--no-database** option

```
su omnileads -  
cd /opt/omnileads/bin  
./backup-restore.sh --backup --no-database
```

• Restore

If the restore is performed in a new host, then the file generated in the Backup within the path **/opt/omnileads/backup** must be left available.

Important: In case of doing the restore in a new instance, it is necessary that the machine has a full OMniLeads running in the same version backup file has been generated from.

To perform a restore, we must execute:

```
su omnileads  
cd /opt/omnileads/bin/  
./backup-restore.sh --restore=nombre_del_archivo_de_backup.tar.gz
```

It is not needed to add the full location path of the backup.

```
[omnileads@oml-app-restore bin]$ ./backup-restore.sh --restore=210809-115258-oml-backup.tar.gz  
Restoring backup: OMniLeads application...  
Restoring backup: Asterisk...  
Restoring backup: Kamailio...  
Restoring backup: Database...  
Running 'regenerar_asterisk'...  
The file 'omnileads_envars.sh.backup' was found on your backup file, so it was copied to /opt/omnileads/backup, in case you need to check variables from your previous OMniLeads instance.  
Restore done!
```

Note:

- If there is no database backup, the restore of database isn't done
 - A copy of omnileads_envars.sh file will be placed in **/opt/omnileads/bin/omnileads_envars.backup**
 - If the backup includes addons, the restore is going to execute the reinstall of these addons
-

16.1.6 Upgrades

OMniLeads create continuously releases, so it's necessary that you upgrade the software periodically.

Important: Upgrade under release-1.3.1 to release-1.3.1 (including it)

- Is **ESSENTIAL** to know the passwords for *postgresql*, *mysql* and *django admin* that were set during installation. You can see these passwords in *my_inventory* file and you will have to type them again in *inventory* file. If the same passwords are not used, the upgrade will set up the passwords you typed in inventory file
- If you don't use the same MySQL password you set during install, the upgrade will fail

Upgrade after release-1.3.1

- If you don't want to change variable values, just define the installation type.

- If you want to change some variable, ingress it in inventory file and the upgrade process will change it for you.

****Upgrade from release-1.15.0 (or any previous version) to release-1.16.0 ****

- In this delivery, the components are extracted from the main repository: rtpengine, asterisk, redis, websocket, kamailio, postgres, and nginx.

They all reside in isolated and dedicated repositories, included as GIT submodules of the main repository of the App. Therefore the upgrade procedure should involve the submodules initialization as indicated below.

Below are the steps to follow in order to perform a new platform update. This task is also performed with the script “deploy.sh”.

Self-Hosted Upgrade

In order to proceed:

- Access as root user in the machine with OMniLeads installed
- Go over root directory *ominicontacto* and execute:

```
git fetch
git checkout release-V.V.V
git submodule init
git submodule update --remote
```

Note: Remember that the *Tab* key when pressing more than once, autocompletes the command displaying all releases.

- Next we should position on the directory

```
cd install/onpremise/deploy/ansible
```

- Now we proceed with the edition of the inventory file, where only the variables should be adjusted:

```
#####
↪####
# If you are installing a prodenv (PE) AIO y bare-metal, change the IP and hostname_
↪here #
#####
↪####
[prodenv-aio]
localhost ansible_connection=local ansible_user=root #(this line is for self-hosted_
↪installation)
#10.10.1

extern_ip=auto
```

- Then the script is executed with the -u (update) parameter. This execution will take some minutes and implies applying all the updates downloaded with the “git pull origin master” on our OMniLeads instance.

```
./deploy.sh -u --iface=$NETWORK_INTERFACE
```

- If everything flows correctly, at the end of the task execution we will see a screen as shown in figure 13.

[illegible]

Figure 13: updates OK

Upgrades using Deployer-Nodes method

- The cloned repository should be accessed on our Workstation machine, to run the update on the Linux OM-niLeads host.

```
cd PATH_repo_OML
cd ominicontacto/deploy/ansible
git fetch
git checkout release-V.V.V
git submodule init
git submodule update --remote
```

Remember that the *Tab* key when pressing more than once, autocompletes the command displaying all releases. Once the release is selected:

- Uncomment the line for self-hosted installation in the inventory file

```
#####
↪###
# If you are installing a prodenv (PE) AIO y bare-metal, change the IP and hostname_
↪here #
#####
↪###
[prodenv-aio]
#localhost ansible_connection=local ansible_user=root #(this line is for self-hosted_
↪installation)
10.10.10.100 ansible_ssh_port=22 ansible_user=root #(this line is for node-host_
↪installation)

extern_ip=auto
```

- Then the script is executed with the -u (update) parameter. This execution will take some minutes and implies applying all the updates downloaded with the “git pull origin master” on our OMniLeads instance.

```
sudo ./deploy.sh -u
```

- Finally, the platform is updated to the last stable version “master”

[illegible]

Figure 14: updates from ansible remote OK

Note: AIO installations will not be supported in future for Debian and Ubuntu so, is recommended to use CentOS.

16.1.7 Changes of network parameters (Hostname and/or IP Address) and changes of passwords of services

- To do these tasks, you must execute again the `deploy.sh` script
- **If you want to change IP/hostname** you must inside the server with root user, change the IP/hostname and be sure the system got the changes. A **reboot** of the system is recommended.
- **If you want to change passwords** edit the password you wish, see `about_install_inventory_vars` to check the variables related to passwords.

You must execute the `deploy.sh` script, this way:

```
./deploy.sh -u
```

Important: You must be sure you run the deploy script in the same release installed in the system, otherwise an upgrade of software will be done.

16.1.8 Users unblock

OMniLeads count with a block users system, when someone enter the wrong password three times. This is the security measure implemented to avoid brute force attacks in the platform's Login console. The administrator user has the possibility of unblocking any user who has been blocked by entering the wrong password unintentionally.

To unblock it you enter the following URL: <https://omnileads-hostname/admin>, this URL displays the **Django Administrator Console**.

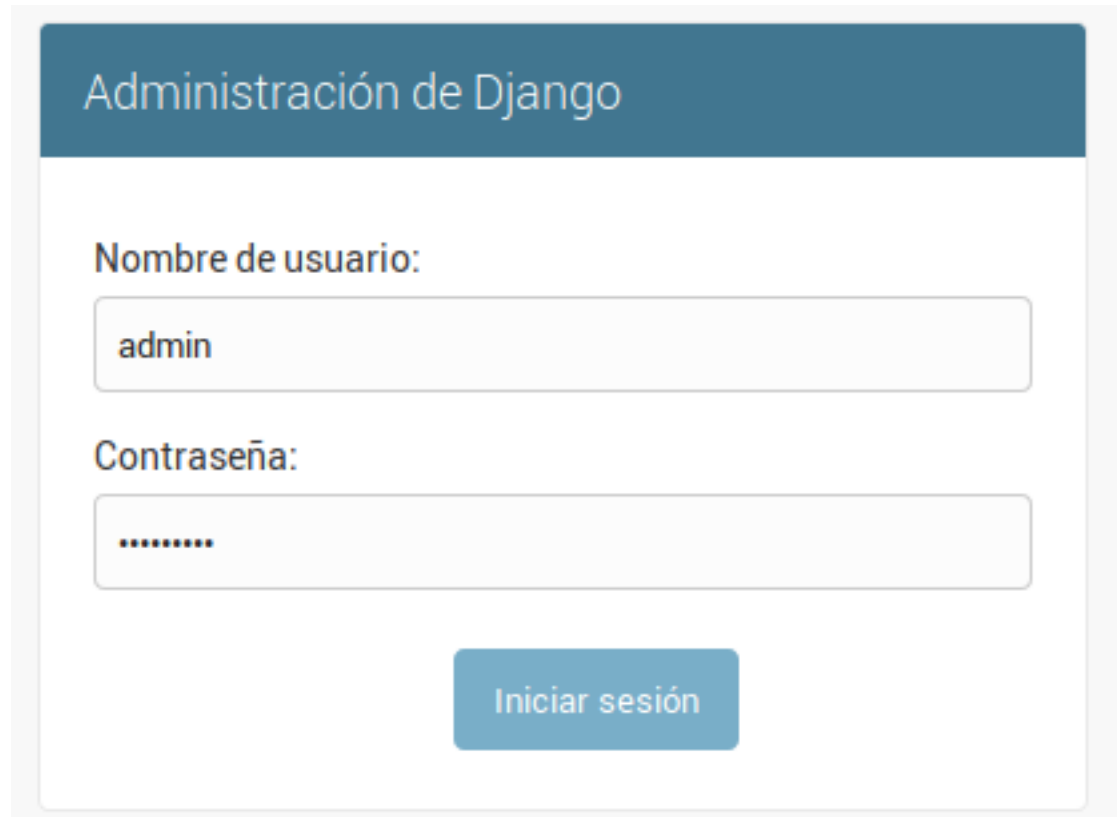
The image shows the Django Administrator Console login interface. It has a blue header with the text "Administración de Django". Below the header, there are two input fields: "Nombre de usuario:" with the text "admin" entered, and "Contraseña:" with a masked password represented by dots. At the bottom of the form is a blue button labeled "Iniciar sesión".

Figure 15: Django admin console

There, enter the admin user credentials. Then click on the button **Defend**



Figure 16: Defender in django admin

This opens the **Django Defender** administrator (<https://github.com/kencochrane/django-defend>) which is the Django plugin used to manage this. Click on **Blocked Users**

Administración de Defender

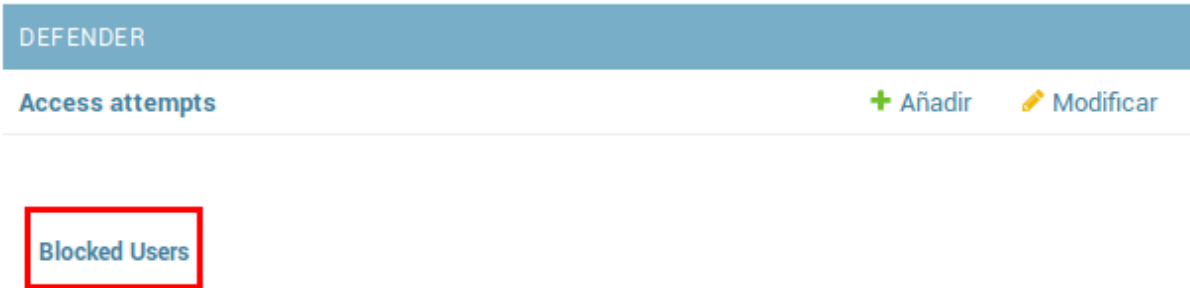


Figure 17: Blocked users view

You will see the blocked user. Just click on **Unblock** to unblock it.

Blocked Logins

Here is a list of IP's and usernames that are blocked

BLOCKED IP'S	
IP	ACTION
No IP's	

BLOCKED USERNAMES	
USERNAMES	ACTION
agentetest	<div>unblock</div>

Figure 18: Unblock user view

Now the user can login without problem.

```
<<<<<<< HEAD Recover & Takeover nodo PostgreSQL HA ***** ===== ..
_about_recovery_pgsql_ha:
```

16.1.9 Recovery & Takeover nodo PostgreSQL HA

```
>>>>>>> develop
```

Para realizar la recuperacion de un nodo principal que por algun tipo de falla quedo fuera de servicio y por lo tanto el nodo de backup pasa a ser principal. Para volver a unir nuestro nodo al cluster, se deben ejecutar los siguientes comandos:

```
<<<<<<< HEAD main_pgsql: systemctl stop postgresql-11 main_pgsql: su postgres - main_pgsql: cd ~ main_pgsql:
repmgr -h replica_pgsql -U repmgr -d repmgr -f repmgr.conf standby clone --dry-run main_pgsql: repmgr -h
```



```
replica_pgsql -U repmgr -d repmgr -f repmgr.conf standby clone -F main_pgsql: systemctl start postgresql-11
main_pgsql: repmgr -f repmgr.conf standby register -F
```

```
main_pgsql: repmgr -h ip_backup_node -U repmgr -d repmgr -f repmgr.conf standby clone --dry-run
main_pgsql: repmgr -h ip_backup_node -U repmgr -d repmgr -f repmgr.conf standby clone -F
main_pgsql: systemctl start postgresql-11 main_pgsql: systemctl start repmgr11 main_pgsql: repmgr
-f repmgr.conf standby register -F
```

>>>>>> develop Ahora bien, para dejar nuestro cluster en el estado inicial, es decir en la configuración de main - backup explicitada en la instalación, se deben ejecutar la siguiente lista de comandos:

```
replica_pgsql: systemctl stop postgresql-11
```

```
<<<<<<< HEAD replica_pgsql: repmgr -h main_pgsql -U repmgr -d repmgr -f repmgr.conf standby clone --dry-run
replica_pgsql: repmgr -h main_pgsql -U repmgr -d repmgr -f repmgr.conf standby clone -F

replica_pgsql: repmgr -h ip_main_node -U repmgr -d repmgr -f repmgr.conf standby clone --dry-run
replica_pgsql: repmgr -h ip_main_node -U repmgr -d repmgr -f repmgr.conf standby clone -F

>>>>>>> develop replica_pgsql: systemctl start postgresql-11 replica_pgsql: repmgr -f repmgr.conf standby register
-F
```

16.1.10 OMniLeads uninstall

If by any reason you want to uninstall OMniLeads from your machine or VM, there is a script for that. It is already incorporated in the install process, just execute it:

```
oml-uninstall
```

This script:

- Uninstall the essential services of omnileads: asterisk, kamailio, rtpengine, mariadb, postgresSQL, wombat dialer, redis, nginx and omniapp.
- Delete the file /opt/omnileads (including recordings)
- Remove the databases

Note: The script does not uninstall the dependency packages used to install the services.

Important: Be careful when executing it, once executed there is no way to recover the system.

OMniLeads allows integration with Web CRM systems, allowing to configure the software to send notifications and requests from OMniLeads to the CRM system and vice vers through the system API.

17.1 CRM integration

As well mentioned in the official documentation, OMniLeads allows us a bi-directional interaction with a CRM system. That's why we go to divide the configurations in two parts.

From one hand:

17.1.1 OMniLeads to CRM Interaction

Each campaign can invoke a CRM or particular view

OMniLeads will execute program calls or notifications to the CRM when using campaigns with external URL set from the administration level. This can occur in incoming campaigns, preview or predictive dialer.

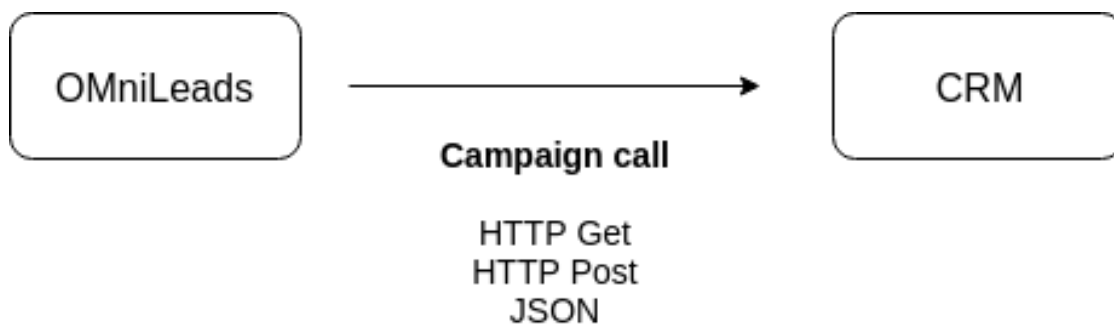


Figure 1: campaign calls and crm

The idea of this interaction is that the agent counts on with a view of the contact information in the CRM. OMniLeads allows a CRM URL invocation involving parameters of the call, contact and / or custom parameters within the context of the campaign running the program call to the CRM.

Depending on the settings applied in the campaign configuration, the invoked URL may be embedded within the agent console , or a new browser tab can be opened for each program call. Another action is to simply send an HTTP-Post JSON to the CRM.

Figure 2: CRM and agent console

Finally we must consider that the execution of the CRM URL by OMniLeads can be done automatically (i.e. when call attempt is performed) or triggered by the agent after pressing a button in the agent console.

All configuration details associated with this scenario of CRM Integration are covered within this section.

Activate a new External CRM entity

The first step to follow is to register the entity " External CRM" with the related web address (URL) and the interaction settings we expect. For that, access to the menu *Campaigns - External Sites - New Site*.

Figure 3: new crm

As stated in figure 3, in this step we need to simply complete the resource name, define the URL of the resource to invoke, and the type of interaction (GET, POST or JSON) with external CRM. Finally we set if the system will open a new tab per connected call, make the request and embed the result within the agent console or send a notification (JSON) to the CRM with the call parameters.

Let's list each form field (figure 3):

- **Name:** reference name
- **URL:** this is the web address to invoke on each call. Here we only declare the web resource to invoke. As we will see later, the parameters are customized per campaign.
- **Trigger:** here you can select the way you are going to invoke CRM URL web address.
 - When “Agent” is selected, then when the call connects with an agent, this is the one who triggers the execution of the CRM URL through an AJAX request from the browser.
 - When “Automatic” is selected, the execution of CRM URL is performed at the time the call is delivered to the agent console, through an AJAX request from the browser.
 - When “Server” is selected, then a HTTP POST request to CRM is generated.
- **Method:** the execution of the CRM URL can be done through GET/POST request.
- **Format:** in case of using HTTP-POST, HTML format can be defined here.
- **Scope:** if the “trigger” is set as Automatic or Agent, then the result of the request made to the CRM URL can be displayed “embedded” in the agent console or by opening a new “tab” in the agent browser.

Once the CRM with all the configuration parameters is generated, we can affect it to different campaigns so that the CRM can be invoked in each call delivered to an agent.

Campaign configuration with CRM Interaction

All OMniLeads campaign types are able to activate this CRM interaction per agent call. In his point we will see some examples on how to perform this configuration using the campaign wizard (*figure 3*).

All campaigns have the ability to trigger a form or a CRM Interaction at the time a call is connected to an agent. In this configuration you may indicate CRM Interaction and then define the External CRM entity that will take place.

Nombre:

Base de Datos de Contactos:

Los espacios, tildes, ñ-es y otros caracteres no ASCII no están permitidos.

Sistema externo:

ID en sistema externo:

Tipo de Interacción

☐ Formulario

☒ Url externa

Url Externa

Sitio: Sales CRM 2 - url: https://sales-crm.example

Grabar llamados: ☐

Objetivo:

Tiempo de desconexión:

A partir de este tiempo en minutos, un agente será desconectado de cualquier contacto que tenga asignado y no haya atendido.

Figure 4: CRM campaign activate

Then it is necessary to assign the parameters to be sent to the CRM. This is also set in the campaign configuration. (figure 5).

The screenshot shows a web interface for configuring a CRM campaign. At the top, there is a horizontal navigation bar with several tabs: 'Campaign', 'Queue Basic Information', 'Call Disposition Options', 'External Site interaction' (which is highlighted with a green underline), 'Add Supervisors', and 'Add Agents'. Below the navigation bar, there is a form area. On the left, there is a dropdown menu with a downward arrow and a placeholder text '-----'. To the right of the dropdown are two input fields: 'Value' and 'Name'. To the right of the 'Name' field is a red button labeled 'Remove'. Below these fields is a green button labeled 'Add parameter'.

Figure 5: CRM campaign params

At this configuration phase, we can indicate the expected parameters available in OMniLeads System that must be sent to the CRM every time a call is connected to an agent. Those available parameters are grouped by four families:

- **Campaign data, conformed by the parameters:**
 - **id:** id of the campaign.
 - **name:** represents the campaign name.
 - **type:** represents the campaign type.
- **Call data, conformed by the parameters:**
 - **call_id:** is the transaction identifier within OMniLeads.
 - **agent_id:** this is the id of the agent that is processing the call in charged of triggering the request to the CRM.
 - **telephone:** is the contact telephone number.
 - **contact_id:** is the internal id of the contact in the campaign.
 - **rec_filename:** the name of the audio file that contains the recording of the call connected to the agent.
- **Contact Data,** parameters available from the columns of the current campaign contact database. Any column can be chosen as a parameter to send to the CRM.
- **Fixed Parameter,** you can set custom parameters to be sent on each call.
- **Dialplan data,** it is a parameter that the Dialplan should send in SIP originate header.

With all parameters defined above, in the figure 5 we need to note that three fields still need to be completed *per parameter* in order to be sent.

- **1st Field:** corresponds to the parameter type (campaign data, call data, database data or fixed data).
- **2nd Field:** corresponds to the specific name of the parameter to be sent (for example “name” if it is a campaign data).
- **3rd Field:** is the name of each parameter, expected from the CRM side.

Example 1: CRM interaction using GET

Let's assume the following URL needs to be executed: **https://my_crm.domain.com?idClient=321321321&idCamp=11&lang=115-20190604-2-4149014-1559667982.424.wav**

As you can note in our example of URL, each execution must provide:

- Contact ID
- Campaign ID that will invoke external CRM
- Parameter “lang=es”
- Recording of the current connected call

How would we implement this requirement from what we have covered in this chapter?

Generate the new CRM

Figure 6 shows the implementation of the proposed CRM seen in our example.

NEW EXTERNAL SITE

Name:	URL:
<input type="text" value="crm_example_1"/>	<input type="text" value="https://my_crm.domain.com"/>
Disparador:	Method:
<input type="text" value="Automático"/>	<input type="text" value="GET"/>
Formato:	Scope:
<input type="text" value="-----"/>	<input type="text" value="Embedded"/>

Figure 6: CRM definition

Therefore, campaign configuration now will proceed in order to invoke the CRM with the parameters specified above.

In Figure 7, we explain how to configure the campaign to work with the CRM of this example.

Campaign Call Disposition Options Add Supervisors Add Agents

Name: Contact Database:

Only ASCII characters are allowed.

Sistema externo: ID en sistema externo:

Type of Interaction:
☐ Campaign Form
☒ External url

External url:

Enable Recordings: ☐

Scope:

Disconnection Time:

In minutes. Once this timer is reached, Assigned Contact will be disconnected from agent if call is not dispositioned.

[Next step](#)

Figure 7: Campaign and CRM

The last step has to do with the assignment of the related parameters for CRM interaction. In *figure 8* we see an example of this step.

Campaign Call Disposition Options **External Site interaction** Add Supervisors Add Agents

Call data	id_contacto	idClient	Remove
Campaign field	id	idCamp	Remove
Fixed	lang	es	Remove
Call data	rec_filename	recordingFile	Remove

[Add parameter](#)

Figure 8: Campaign CRM parameters

Finally we highlight the relationship between columns 2 and 3 for each parameter, since they make the assignment between system parameters and the expected parameter names on the CRM side.

Example 2: CRM Interaction using GET and Clean URLs

Let's assume that you want to run a Clean URL : **https://my_crm.domain.com/idClient/idCamp/lang/recordingFile**

For instance: https://my_crm.domain.com/321321321/11/es/prev-115-20190604-2-4149014-1559667982.424.wav

As you can note in our example of URL, each execution must provide:

- Contact ID
- Campaign ID that will invoke external CRM
- Parameter “lang=es”
- Recording of the current connected call

How would we implement this requirement from what we have covered in this chapter?

Generate the new CRM

The implementation of the proposed CRM is shown as an example in *Figure 9*.

The screenshot shows a form for defining a CRM interaction. It has two columns of fields. The left column contains: 'Name:' with the value 'crm_example_2'; 'Disparador:' with a dropdown menu showing 'Automático'; and 'Formato:' with a dropdown menu showing '-----'. The right column contains: 'URL:' with the value 'https://my_crm.domain.com/{1}/{2}/{3}/{4}' where the placeholders are highlighted in a green box; 'Method:' with a dropdown menu showing 'GET'; and 'Scope:' with a dropdown menu showing 'Embedded'. At the bottom left of the form is a green 'Save' button.

Figure 9: CRM definition with clean URL

The figure highlights the “holders” needed to work with Clean URLs. When generating the URL to be executed, parameters must be specified between brackets. Those Parameters will then be used when generating the campaign for that CRM interaction.

Therefore, campaign configuration now will proceed in order to invoke the CRM with the parameters specified above.

The main difference from the standard URLs (HTTP GET) that is exposed in the *example 1*, is that when assigning parameters in the campaign, “holders” need to be used instead of “Parameter Names”, as shown in *figure 10*.

Call data	id_contacto	{1}	Remove
Campaign field	id	{2}	Remove
Fixed	lang	{3}	Remove
Call data	rec_filename	{4}	Remove
Add parameter			

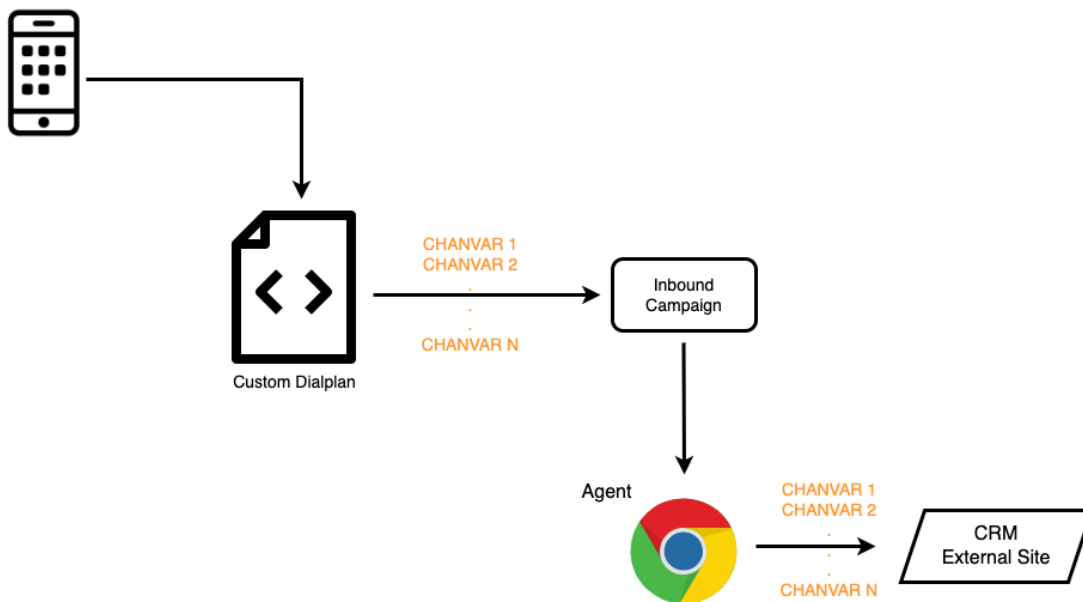
Figure 10: Campaign and CRM parameters

Finally we highlight the relationship between columns 2 and 3 of each parameter, within the scenario of “clean URLs”.

Levantando variables de canal personalizadas de Asterisk y pasándolas al CRM

Hay veces en las que el *modelo de negocio* implica la necesidad de ejecutar un plan de discado de Asterisk personalizado, en este solicitar el ingreso de algún valor DTMF o bien acudir a la posibilidad que nos brinda Asterisk de utilizar servicios de *reconocimiento de voz* para interactuar con la persona que llama al centro de contacto.

Vamos a intentar ejemplificar un workflow con un ejemplo.



Entonces supongamos que se desea solicitar a un llamante que ingrese su número de cliente. Por lo que se podría tomar un DID de nuestro troncal SIP y enviarlo a un *dialplan personalizado*.

Edit inbound route

Name: *	DID number: *
<input type="text" value="Ruta entrante 0"/>	<input type="text" value="41004100"/>
Caller id prefix:	Language: *
<input type="text"/>	<input type="text" value="English"/>
Destination type: *	Destination: *
<input type="text" value="Custom destination:"/>	<input type="text" value="Custom destination:: custom"/>
<input type="button" value="Save"/>	

Luego vamos a trabajar en dar de alta el destino personalizado y su código.

Update Custom Destination

Name: *	<input type="text" value="custom"/>
Custom destination: *	<input type="text" value="customerdata,41004100,1"/>
Failover destination	
Destination type:	Destination:
<input type="text" value="Inbound campaign"/>	<input type="text" value="Inbound campaign: in_crm_dialplan"/>
<input type="button" value="Save"/>	

Respecto al dialplan como tal, se recuerda que el mismo debe ser generado en el archivo: `oml_extensions_custom.conf`.

```
[customerdata]
exten => 41004100,1,Verbose(custom dialplan customer data)
same => n,Answer()
same => n,Read(Omlcrmcustomerid,customerid,1,,1,5)
same => n,Read(Omlcrmcustomerplan,customerplan,1,,1,5)
same => n,Set(OMLCUSTOMDIALPLANVARS=Omlcrmcustomerid&Omlcrmcustomerplan)
same => n,Gosub(sub-oml-dst-switch,s,1(1,5))
```

Vamos a analizar el dialplan en cuestión:

En primer lugar y como es de esperar habrá una parte en donde el desarrollador captura la interacción del llamante, en el caso del ejemplo es a través de la aplicación de dialplan *READ*.

Important:

- El nombre de las variables TIENE que contener el prefijo **Omlcrm**
- Se TIENE que crear la variable **OMLCUSTOMDIALPLANVARS** que contenga TODAS las variables que se desean pasar al CRM concatenadas con el carácter `&`
- Para enviar la llamada a un *nodo* de OMniLeads se debe invocar en el dialplan a: `Gosub(sub-oml-dst-switch,s,1(X,Y))` donde **X** es el tipo de destino:

- 1 campanana entrante
- 2 condicional de tiempo
- 3 IVR
- 6 Encuesta

mientras que el valor **Y** hace referencia al *id* del objeto en cuestion. En nuestro ejemplo (1,5) el 1 hace referencia a una campana entrante cuyo id es el 5.

Finalmente en la campana en cuestion, se configura la integracion con CRM de la siguiente manera:

INBOUND CAMPAIGN: EXTERNAL SITE INTERACTION

Campaign Queue Basic Information Call Disposition Options **External Site interaction**

Call data	telefono	tel	Remove
Call data	rec_filename	callrec	Remove
Dialplan data	Omlcrmcustomerid	name	Remove
Dialplan data	Omlcrmcustomerplan	plan	Remove
-----	Value	Name	Remove

Add parameter

first step prev step Finish

And for the other hand, we get:

17.1.2 CRM to OMniLeads Interaction

It is also possible from a CRM system to execute requests once accessing to the endpoints of the `about_api`. Throughout this section we will go to explain how to activate the actions available from an external CRM.

- **Click to call:** a CRM user who is available in the contact view can make a call to the contact number clicking on that number within the CRM. The click action triggers a program call to an OMniLeads API method in order for the system to route the call accordingly.
- **Call Disposition:** each call connected to an agent can execute a request to the CRM passing usable call parameters. When the user finishes the call within the CRM, she/he proceeds with the “Call Dis-

position”. This action allows CRM to access an OMniLeads API method establishing a correlation between the OMniLeads campaign level and CRM contacts.

In order to implement the listed actions, CRM developer needs to implement these functionalities consuming the `about_api`. Once the functionalities from CRM side are available, the following configurations need to be set as to start operating with this integration level.

CRM and OMniLeads entities Relationships

As we well know, each call processed by OMniLeads implies a relationship between a Campaign, an Agent and in most cases a Contact. We could talk about the Trinity: “Agent - Contact - Campaign”.

In the CRM universe we also know that we have the same relationship between the CRM User, the campaign she/he is operating and the contacts associated to the campaign. Therefore, the relationship stated here has to do with the fact of being able to relate each OMniLeads agent, campaign and contact with the expected CRM entity side. This concept will allow a perfect correlation between CRM and OMniLeads systems when executing a “click to call” or “disposition a contact”.

OMniLeads has its own identifiers (agent, campaign and contacts of the campaign) self-generated. It is also common to find the same scenario in CRM, therefore it is desired to make a mapping process between OMniLeads and CRM in this aspect. We will explain below how to synchronize these identifiers between both systems.

OMniLeads Agents and CRM Users relationship

The first step is to register an External System entity and associate the OMniLeads agents with CRM users by entering the ID (identifier) of the user in the CRM, as indicated in Figure 1.

Nombre:

Open CRM

Agents

Agente:	Id externo agente:	
Adrian Belew ▼	1001	Remove
mikael akerfeldt ▼	1002	Remove

Add agent

Save

Figure 1: new external crm

That being said, each identified agent will be able to interact from the CRM.

Relationship between OMniLeads and CRM Contact Database

In order to create a Contact Database relationship between systems, the database upload process will offer an external ID selector field, as shown in figure 2.

tel1	name	surname	id_contact	tel2	tel3
4553101	Terence	McKenna	30000001	33213211	5343431
4553102	Timothy	Leary	30000002	33213211	5343431

Phone fields:

- ☒ tel1
- ☐ name
- ☐ surname
- ☐ id_contact
- ☒ tel2
- ☒ tel3

Id externo:

Figure 2: new crm contact database

This value must be unique for each contact in the database, so there should be no contacts with the same value. Each contact can only have one external identifier.

In Figure 2, the example assumes that column “id_contact” is the one reserved for external ID. Therefore, OMniLeads will consider this column/value when interacting with CRM system.

Relationship between OMniLeads and CRM Campaigns

When a Campaign Wiozard is in process, there are fields related to CRM interaction. Remember that in the section of OML to CRM Interaction we present the field Type of interaction “External URL”, to be able to launch a program call to the CRM for each call connected to the agent. In this section we present the Interactions from the CRM to OML, so let’s work with the fields “External system” and “external system ID” respectively (Figure 3).

Campaña Opciones de calificación Dar permiso a supervisores Adicionar agentes

Nombre:

Base de Datos de Contactos:

Los espacios, tildes, ñ-es y otros caracteres no ASCII no están permitidos.

Sistema externo:

ID en sistema externo:

Tipo de Interacción:

☐ Formulario

☒ Url externa

Url Externa:

Grabar llamados: ☒

Objetivo:

Tiempo de desconexión:

A partir de este tiempo en minutos, un agente será desconectado de cualquier contacto que tenga asignado y no haya atendido.

Figure 3: new camp crm to oml interaction

Therefore, the wizard process selects the external CRM that will execute the requests against OML (External System). And on the other hand, the ID of the corresponding campaign we wish to link to (external system ID). Each campaign can only be linked to one External CRM campaign.

Then we move forward with the creation of the campaign with all the steps that we already know.

Note: When the exposed linkage is processed, the following exceptions can exist.

- When assigning agents that do not have an external identifier in the selected External System.
- When assigning a Contact Database that is already assigned to a campaign with another External System configured.
- When assigning an External URL that is already assigned to a campaign with another External System.

Notifications will also appear when editing an External System , if there are missing external identifiers in agents assigned to campaigns where External System is configured.

In this section you will find all the information about the Rest API of the system

18.1 OMniLeads RESTful API

This section is destined to developers that want to execute an integration between its CRM system and OMniLeads. For that reason, the terminology and information provided here has software developers as its public target.

OMnileads offers a RESTful API base on HTTPS / JSON. This API allows access to system resources and services by outside of the user web interface, allowing this way that external systems could communicate in a simple way with OMniLeads.

The authentication methods available for this API are: Session (the agent must be logged on the system via web interface) and Token (using its credentials to obtaining a token)” and then passing it on the headers of the request:

“”Authorization: Bearer <token value>””

For example:

```
GET https://<omnileads-addr>/api/v1/disposition/  
Authorization: Bearer 22bab1b107251a6771bdf3dbeb5184a7f4b5c182  
Content-Type: application/json
```

Note: In new releases this section will increase, in order to add new endpoints.

Then when describe the available endpoints

18.1.1 Login Endpoint

This endpoint allows to authenticate as a system user, in case of success, it allows access to others available endpoints depending of the user profile

URL: POST https://<omnileads_addr>/api/v1/login

```
POST https://<omnileads_addr>/api/v1/login
Content-Type: application/json
{
    "username": "agente_username",
    "password": "agente_password"
}
```

figure 1: endpoint login request

filed name	type	description
username	string	username value generated from the OML users creation menu
password	string	password value generated from the OML users creation menu

Successful authentication

If the login is successful, the endpoint shows the following output:

```
HTTP 200 OK
Allow: POST, OPTIONS
Content-Type: application/json
Vary: Accept

{
    "token": "6c8a7b970996480adcbaf17acff053d145b039ff",
    "expires_in": "359.997618",
    "user": {
        "first_name": "Alessio",
        "last_name": "Tachinardo",
        "email": "a@a.it"
    }
}
```

figure 2: endpoint login request ok

As shown in the picture, a successful login, returns fields like security “token”. This token must be used on next requests to the API from the authenticated user. Also, in the field “expires_in” indicates the token lifetime

In case that the system makes a request and the security token has expired, then a new authentication request must be done.

Note: The security token’s lifetime can be configured modifying the “TOKEN_EXPIRED_AFTER_SECONDS” parameter located on “/opt/omnileads/omnicontrato/omnicontrato/settings/production.py”

Authentication failed

If the login failed, the endpoint returns the following output:

```
HTTP 404 Not Found
Allow: POST, OPTIONS
Content-Type: application/json
Vary: Accept

{
  "detail": "Invalid Credentials or activate account"
}
```

figure 3: endpoint login request fail

18.1.2 Endpoint to obtain Contact database structure

This endpoint makes possible to obtain information fields information of contacts database related to a campaign. With this information is possible then to create a new contact. The credentials must belong to an Agent (*Users*) or a Supervisor (*Users*) associated to a campaign

URL: POST https://<omnileads_addr>/api/v1/campaign/database_metadata/

filed name	type	description
idEx-ternal-System	integer	Optional parameter, if sent the system tries to locate the contact as 'external_id' on the Campaign contacts database. If not sent, the system will assume that the value of the paramater 'idContact' is the intern id on OML
idCam-paign	string	Contact identifier to tag, its value depends on if 'idExternalSystem' parameter is sent

In case of no errors occurred it will show an output like this, with data of the new disposition created

```
{u'external_id': u'idext',
 u'fields': [u'nombre',
             u'idext',
             u'tel',
             u'tel2'],
 u'main_phone': u'tel',
 u'status': u'OK'}
```

The 'fields' field indicate the list of all the fields of a contact database. The field 'main_phone' indicates which is the field correspondent to the main number. The field 'external_id' indicates which field correspond to the external id of contact. When the database doesn't have external id, the 'external_id' field will be None.

In case of errors occurred, the endpoint returns a JSON with the field 'status': 'ERROR' and the detailed information of the error on the field 'errors'. On other case the 'status' field value will be 'OK'

18.1.3 Endpoint to create contact

This endpoint allows to add a contact in a database referred to a campaign. The credentials must belong to an Agent (*Users*) or a Supervisor (*Users*) associated to a campaign

URL: POST https://<omnileads_addr>/api/v1/new_contact/

filed name	type	description
idEx-ternal-System	in- te- ger	Optional parameter, if sent the system tries to locate the contact as 'external_id' on the Campaign contacts database. If not sent, the system will assume that the value of the paramater 'idContact' is the intern id on OML
idCam- paign	string	Contact identifier to tag, its value depends on if 'idExternalSystem' parameter is sent

Also, it must send the values of the fields correspondent to the contact database, and their names can be obtained with using the endpoint: Obtain Contact database structure (*Endpoint to obtain Contact database structure*). Is mandatory to send the value to the field 'main_phone', and in case the database has external id, the field's value 'external_id' mustn't exist previously in other contact of database.

In case of no errors occurred it will show an output like this, with data of the new disposition created

```
{u'contacto': {u'idext': u'id7',
               u'nombre': u'Juan',
               u'tel': u'351222221',
               u'tel2': u''},
 u'id': 30,
 u'message': u'Contacto agregado',
 u'status': u'OK'}
```

In case of errors occurred, the endpoint returns a JSON with the field 'status': 'ERROR' and the detailed information of the error on the field 'errors'. On other case the 'status' field value will be 'OK'

18.1.4 Call generator endpoint

Allows to generate calls (click to call) from an External CRM System. The credentials must belong to an Agent (*Users*).

URL: POST https://<omnileads_addr>/api/v1/makeCall

```
{
  'idExternalSystem': <idExternalSystem> :String,
  'idAgent': <idAgent> :String,
  'idContact': <idContact> :String,
  'idCampaign': <idCampaign> :String,
  'phone': <phoneContact> :String
}
```

figure 4: endpoint new call request

filed name	type	description
idEx-ternal-Sys-tem	string	Optional parameter, it must be sent if needed to link campaign with the external CRM system
id-Cam-paign	string	Required parameter, must match with an OML campaign identifier. If the parameter 'idExternalSystem' is sent, it must match with the field "external identifier" of a campaign associated to the External System specified
idA-gent	string	Required parameter, must match to a system Agent identifier. If the parameter 'idExternalSystem' is sent must match to the field "external identifier" of an Agent associated to the external CRM System
id-Con-tact	string	Optional parameter, if is not sent the system assumes that is a new contact. If sent must match with an identifier of the campaign databasecontact. If the 'idExternalSystem' is sent, it must match with the contacts database field marked as an external identifier

In case of errors occurred, the endpoint returns a JSON with the field 'status': 'ERROR' and the detailed information of the error on the field 'errors'. On other case the 'status' field value will be 'OK'

18.1.5 Disposition options list endpoint

URL GET https://<omnileads_addr>/api/v1/campaign/<idc:integer>/dispositionOptions/ (1)

URL GET https://<omnileads_addr>/api/v1/campaign/<idc:string>/dispositionOptions/<ids:integer>/ (2)

This endpoint allows to get a disposition options list available for tag a contact on a campaign. The credentials must belong to an Agent (*Users*).

The parameters for this endpoint must be specified on the url. It has two modes to use, if it uses the (1) mode, with a single parameter, the 'idc' parameter value must be an integer specifying the OML intern campaign identifier

The mode (2) is for using the endpoint from an external CRM system to OML and in this case the parameter 'ids' must indicates an external CRM system id and the 'idc' parameter must indicates the identifier of one campaign in this external system

In case of the execution without errors the endpoint will return a disposition options list like the following:

```
HTTP 200 OK
Allow: GET
Content-Type: application/json
Vary: Accept

[
  {
    "id": 1,
    "name": "Success",
    "hidden": false
  },
  {
    "id": 6,
    "name": "Agenda",
    "hidden": false
  },
  {
    "id": 2,
    "name": "Hangs up angry",
    "hidden": true
  },
]
```

In case that the id does not match with an id of a campaign or CRM system the endpoint will return an output like:

```
HTTP 404 Not Found
Allow: GET
Content-Type: application/json
Vary: Accept

{
  "detail": "No encontrado."
}
```

18.1.6 Dispositions list endpoint

This endpoint allows get a dispositions list made by the agent who make the request. (*Users*)

URL: GET https://<omnileads_addr>/api/v1/disposition/

In case of no errors occurred, it returns the dispositions list made it by the agent

```

HTTP 200 OK
Allow: GET, POST
Content-Type: application/json
Vary: Accept

[
  {
    "id": 35,
    "idContact": 48,
    "callid": "96855041",
    "idDispositionOption": 131,
    "comments": "Create peace."
  },
  {
    "id": 36,
    "idContact": 49,
    "callid": "93178525",
    "idDispositionOption": 132,
    "comments": "Me indeed open."
  }
]

```

18.1.7 Create new disposition endpoint

This endpoint allows to “tag” the result of a management about a contact. When a CRM user ends a management, it is normal that management closes with a disposition made, and using this endpoint an External CRM System can integrate this action to OML. The credentials used must belong to an Agent (*Users*).

URL: POST https://<omnileads_addr>/api/v1/disposition/

field name	type	description
idExternalSystem	integer	Optional parameter, if sent the system tries to locate the contact as ‘external_id’ on the Campaign contacts database. If not sent, the system will assume that the value of the parameter ‘idContact’ is the intern id on OML
idContact	string	Contact identifier to tag, its value depends on if ‘idExternalSystem’ parameter is sent
idDispositionOption	integer	The disposition option campaign id that will be used to tag the contact, each campaign defines its disposition options. See the endpoint that allows to obtain that values
callid	string	Optional parameter, call identifier
comments	string	The agent observations in the disposition

In case of no errors occurred it will show an output like this, with data of the new disposition created

```
HTTP 201 Created
Allow: GET, POST
Content-Type: application/json
Vary: Accept

{
  "id": 37,
  "idContact": 25,
  "callid": "93178525",
  "idDispositionOption": 60,
  "comments": "Me indeed open."
}
```

If an attempt to create a new disposition instance is made, to a contact already tagged on the campaign, the endpoint will return the following error:

```
HTTP 400 Bad Request
Allow: GET, POST
Content-Type: application/json
Vary: Accept

{
  "status": "ERROR",
  "msg": "There is another disposition for this contact on this campaign"
}
```

If the contact id on the campaign database is not found the endpoint will return the following error:

```
HTTP 400 Bad Request
Allow: GET, POST
Content-Type: application/json
Vary: Accept

{
  "contact": "Contact id not found"
}
```

If the disposition option id is not found the endpoint will return the following error:

HTTP 400 Bad Request**Allow:** GET, POST**Content-Type:** application/json**Vary:** Accept

```
{
  "idDispositionOption": "Disposition option id not found"
}
```

18.1.8 Create new contact and assign it a new disposition endpoint

This endpoint allows to ‘tag’ a management and, at a same time, to create a contact, it means that it creates the contact and the disposition is linked to this new contact. The credentials used must belong to an Agent (*Users*).

URL: POST https://<omnileads_addr>/api/v1/new_contact/disposition/

filed name	type	description
phone	string	The contact phone number
idExternal-Contact	string	Optional parameter, the contact id on an external CRM system
idDispositionOption	integer	The disposition option campaign id that will be used to tag the contact, each campaign defines its disposition options. See the endpoint that allows to obtain that values
comments	string	The agent observations in the disposition
callid	string	Optional parameter, call identifier
<optional_bd_field>	string	Optional parameters, they can define values to fill the custom data” of the contact that will be created, the field names must match with the fields of campaign database

In case of no errors occurred it will show an output like this, with data of the new disposition created

HTTP 201 Created**Allow:** GET, POST**Content-Type:** application/json**Vary:** Accept

```
{
  "id": 37,
  "idContact": 25,
  "callid": "93178525",
  "idDispositionOption": 60,
  "comments": "Me indeed open."
}
```

If the disposition option id is not found the endpoint will return the following error:

```
HTTP 400 Bad Request
Allow: GET, POST
Content-Type: application/json
Vary: Accept

{
  "idDispositionOption": "Disposition option id not found"
}
```

18.1.9 Disposition update endpoint

This endpoint allows to update an existent disposition in OMniLeads

The credentials must belong to an Agent (*Users*)

URL: PUT https://<omnileads_addr>/api/v1/disposition/<idDisposition>

filed name	type	description
idExternalSystem	integer	Optional parameter, if sent the system tries to locate the contact as 'external_id' on the Campaign contacts database. If not sent, the system will assume that the value of the paramater 'idContact' is the intern id on OML
idContact	string	Contact identifier to tag, its value depends on if 'idExternalSystem' parameter is sent
idDispositionOption	integer	The disposition option campaign id that will be used to tag the contact, each campaign defines its disposition options. See the endpoint that allows to obtain that values
callid	string	Optional parameter, call identifier
comments	string	The agent observations in the disposition

If doesn't exist, the endpoint returns the following output:


```
HTTP 200 OK
Allow: GET, PUT
Content-Type: application/json
Vary: Accept

{
  "id": 35,
  "idContact": 48,
  "callid": "96855041",
  "idDispositionOption": 131,
  "comments": "Create peace. asdasd"
}
```

If in the url a non-existent disposition id is specified, the endpoint will return the following output error:

```
HTTP 404 Not Found
Allow: GET, PUT
Content-Type: application/json
Vary: Accept

{
  "detail": "No encontrado."
}
```

If the disposition instance is tried to be modified, changing the parameters 'idContact' and 'idDispositionOption' the system detects that this would make to disposition for the oen contact on the same campaing the endpoint will return the following error output:

```
HTTP 400 Bad Request
Allow: GET, POST
Content-Type: application/json
Vary: Accept

{
  "status": "ERROR",
  "msg": "There is another disposition for this contact on this campaign"
}
```

If the contact id on the campaign database is not found the endpoint will return the following error:

```
HTTP 400 Bad Request
Allow: GET, POST
Content-Type: application/json
Vary: Accept

{
  "contact": "Contact id not found"
}
```

If the disposition option id is not found the endpoint will return the following error:

```
HTTP 400 Bad Request
Allow: GET, POST
Content-Type: application/json
Vary: Accept

{
  "idDispositionOption": "Disposition option id not found"
}
```

18.1.10 Asterisk Agent Session API

API endpoints used by the WebPhone to control the Asterisk's Agent sessions.

Asterisk's Agent Sessions API

Endpoints to control the Asterisk's Agent Session to be used by the WebPhone.

Asterisk agent session login

Establece el estado de sesión del Agente (*Users*) en Asterisk como iniciada. Las credenciales deberán pertenecer al Agente, y no hace falta enviar ningún parámetro extra.

URL: POST https://<omnileads_addr>/api/v1/asterisk_login/

Asterisk agent session logout

Establece el estado de sesión del Agente (*Users*) en Asterisk como finalizada. Las credenciales deberán pertenecer al Agente, y no hace falta enviar ningún parámetro extra.

URL: POST https://<omnileads_addr>/api/v1/asterisk_logout/

Agent Pause start

Establece el estado de sesión del Agente (*Users*) en Asterisk como en Pausa (*Pauses*). Las credenciales deberán pertenecer al Agente.

URL: POST https://<omnileads_addr>/api/v1/asterisk_pause/

field name	type	description
pause_id	string	ID of the pause the agent starts.

Agent Pause End

Establece el estado de sesión del Agente (*Users*) en Asterisk como “Disponible” e indica la finalización de una Pausa (*Pauses*). Las credenciales deberán pertenecer al Agente.

URL: POST https://<omnileads_addr>/api/v1/asterisk_unpause/

field name	type	description
pause_id	string	ID of the pause the agent ends.

18.1.11 Agent SIP credentials endpoint

This endpoint provides the necessary credentials for the agent’s authentication in the SIP server using a Webphone.

URL: GET https://<omnileads_addr>/api/v1/sip/credentials/agent/

```
{
  "status": "OK",
  "sip_user": "1589240643:1003",
  "sip_password": "IvS9X8kgPfY8PfIfiVSsAiAZXW4="
}
```


19.1 Known Issues

- disposition form not showing up when receiving a transfer call
- Pause times are not separated by date
- Supervision calls capture is broken
- Recycle over recycled campaign duplicates records to call
- The system does not reproduce some temporary voice messages produced by some carriers on the webphone
- The mysql password cannot be changed using `–change-passwords`
- The restore of the system doesn't include the Wombat Dialer database so the dialer campaigns are useless after restore

20.1 Release Notes 1.25.0

30-09-2022

20.1.1 New Features

- OML-434: Se agrega soporte de Validación de Duplicados a API de Interacción a CRM.
- OML-1328: Se diseña API para obtener URL de grabación basado en CallID
- OML-2273: Se agrega autenticación de Sitio Externo para interacción con CRM en métodos POST.
- OML-1688: Traducciones i18n de Front-End
- OML-2274: Ahora PostgreSQL permite separar operaciones Read-Only/Read-Write en modo Cluster
- OML-1522: Migración del ABM de Rutas Salientes a VueJS
- OML-1864: Migración del ABM de Rutas Entrantes a VueJS
- OML-2255: Migración del ABM de Pausas a VueJS
- OML-2269: Ahora es posible filtrar grabaciones por Calificaciones.
- OML-2252: Se agrega un nuevo Dashboard de Supervisión y Administrador.
- OML-1975: Soporte Notificaciones via Email: un Nuevo Usuario recibe un correo tan pronto es creado, o cuando su contraseña es modificada.
- OML-2063: Migración del ABM de Formularios a VueJS.
- OML-2209: Migración del ABM de Sistemas Externos a VueJS.
- OML-2250: Se agrega paginación a Resultados de Base.
- OML-2285: Se agrega nuevo EULA en proceso de despliegue de Addons.

20.1.2 Fixes and Improvements

- OML-2276: Fix de Build de VueJS en Develop
- OML-482: Ajustes del Devenv para Redis Sentinel
- OML-2270: Fix de formato de Exportación de Grabaciones en escenarios de múltiples calificaciones
- OML-2275: Refactor del componente de Websocket para las vistas de supervisión
- OML-2294: Fix en reportes de grabaciones para timestamps inferiores al rango del filtro.
- OML-2295: Fix menores en Gitlab Pipelines.
- OML-2256: Optimización en Reportes de No Contactados.
- OML-2254: Optimización en Reportes de Contactados.
- OML-2297: Fix en vista de Adherir Agentes a Campaña cuando las campañas están en Pausa/Inactiva.