

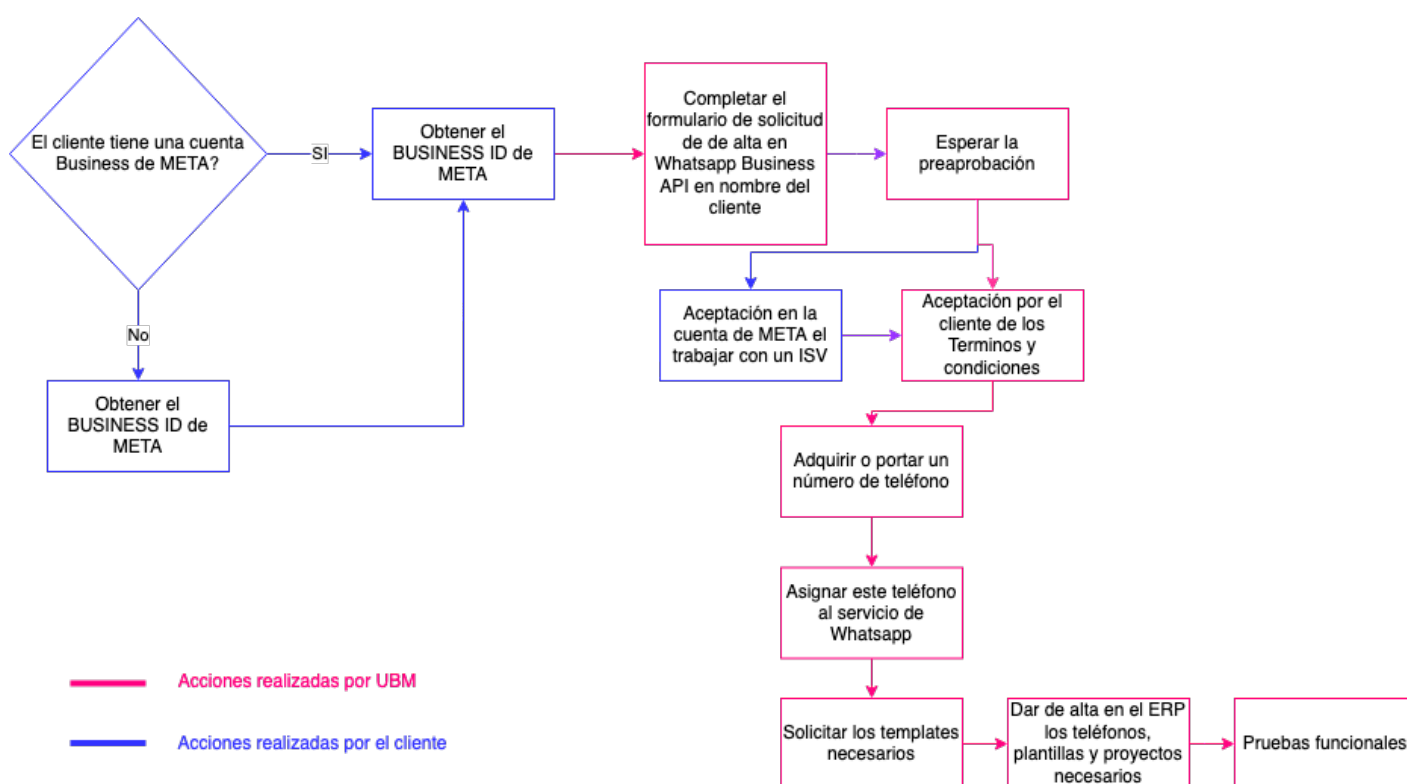
Guías

Resolución a preguntas frecuentes

- Onboarding de nuevos clientes
- Guía de integración del componente web
- Funcionalidades del servicio

Onboarding de nuevos clientes

El proceso de onboarding de nuevos clientes es un proceso no demasiado sencillo. Tiene los siguientes pasos:



- El cliente debe tener una cuenta empresarial de Meta/Facebook, no sirve una cuenta personal a nombre de una empresa. Las cuentas business de Meta/Facebook se encuentran en <https://business.facebook.com/>
- El cliente debe conocer el **Business ID** de la cuenta
 - Una vez conectado al Business de Meta/Facebook hay que acceder a los Settings de la cuenta <https://business.facebook.com/settings/info>
 - Y en esa página pulsar en "Business info" y el ID se encuentra en "Business Information."
- Una vez conocido este Business ID, se debe crear una subcuenta en el panel de Twilio.
 - Cuando se crea una subcuenta nos asignan un ID
- Con el Bussiness ID de Meta y el ID de Twilio UBM procederá a solicitar el acceso a Whatsapp Business API en nombre del cliente.

- Ahora debemos esperar la respuesta de Meta
- El cliente recibirá un mensaje en su cuenta de Meta/Facebook, donde le pedirán que confirmen que nos autorizan, como ISV, a trabajar con ellos.
- Una vez preaprobada la cuenta por parte de Meta/Facebook, el cliente deberá aceptar los terminos y condiciones.
- Una vez completado esto ya está casi.
 - Debemos comprar un teléfono o hacer la portabilidad de uno existente
 - Asignar ese teléfono a un "Whatsapp sender"

Notice: The Sender Display Name must match the name of the end customer's company to avoid rejection by Facebook. Please see the Facebook's [Display Name rules](#).

- La aprobación de los "sender profiles", suele tardar entre 3 y 5 días laborables
- Una vez aprobado pasamos a la definición de la plantilla o plantillas necesarias

Guía de integración del componente web

Para realizar la integración del componente web básicamente tenemos que hacer 4 cosas:

1. Inclusión de las hojas de estilo y javascript necesarios
2. Generación de un div vacío con id **root**
3. Construcción del objeto de inicialización necesario
4. Inicialización del componente

1.- Inclusión de las hojas de estilo y javascript necesarios

Incluir en la/s página/s donde se desee visualizar el componente los siguientes elementos:

```
<link href="https://wa.unir.net/resources/webcomponent/css/2.chunk.css" rel="stylesheet">
<link href="https://wa.unir.net/resources/webcomponent/css/main.chunk.css" rel="stylesheet">
<script src="https://wa.unir.net/resources/webcomponent/js/2.chunk.js"></script>
<script src="https://wa.unir.net/resources/webcomponent/js/main.chunk.js"></script>
<script src="https://wa.unir.net/resources/webcomponent/js/chat-component.js"></script>
```

Una vez incluidas las hojas de estilo y los javascript, debemos añadir un div vacío con nombre root. En este div es donde vamos a "incrustar el componente"

```
<body>
  <!--....-->
  <div id="root"></div>
  <!--....-->
</body>
```

Con esto tenemos todo lo que necesitamos, ahora falta cómo instanciamos el componente. Antes de poder instanciarlo tenemos que crear un objeto donde completamos los datos que vamos a pasar al componente para su configuración. Los elementos soportados en el objeto de instanciación son:

Elemento	Ejemplo	Explicación
phone	" +34659463120"	Teléfono al que queremos mandar el/los mensajes
source	"webcomponent"	Esto es un literal que luego se va a poder usar para filtrar los mensajes, por ejemplo EMPRESA1, EMPRESA2. Este dato es facilitado por UBM
client_name	"María"	Nombre del cliente
client_lastname	"Pérez"	Apellidos del cliente
client_email	"mdlm@email.com"	Mail del cliente
country	"ES"	País del cliente en ISO2
course_id	12311	Código de producto al que hace referencia el mensaje/conversación
assessor_first_name	"Juanito"	Nombre del agente que usa el componente
assessor_last_name	"Rámirez"	Apellido del agente que usa el componente
assessor_email	"jr@mail.com"	Email del asesor que usa el componente
minimized	true	¿El componente debe cargar minimizado?

show_userlist	true	¿Debemos mostrar la lista de conversaciones?
company	"COMPANY"	Nombre de la compañía. Es muy útil cuando el mismo source es usado por compañías distintas.
unit	"Profesorado"	Nombre del departamento. Muy útil cuando un mismo source y company tienen plantillas distintas por departamento.

Elemento	Ejemplo	Tipo	Requerido	Comentario
phone	" +34xxxxxxxxx x"	string	no	Formato internacional
source	"SOURCE"	string	si	Facilitado por UBM
client_name	"Raimundo"	string	si	Nombre del cliente
client_lastname	"Lucas Rodriguez"	string	si	Apellidos del cliente
client_email	"...@..."	string	no	Email del cliente
country	"ES"	string	no	País del cliente. Muy útil para la validación del teléfono
course_id	"249"	string	no	Producto sobre el que muestra interés el cliente
assessor_first_name	"Ana"	string	si	Nombre del empleado que usa el componente
assessor_last_name	"Lucas martín"	string	si	Apellidos del empleado que usa el componente

assessor_email	"...@mycompany.com"	string	si	Email del empleado que usa el componente
minimized	true	bool	no	Si el componente debe arrancar minimizado, por defecto es false
show_userlist	true	bool	no	Si mostramos la lista de conversaciones en el componente, por defecto es false
company	"COMPANY"	string	no	Compañía que usa el source indicado
unit	"Profesorado"	string	no	Departamento del empleado

Un ejemplo de la construcción del objeto sería:

```
var var2init = {
  phone: "+34xxxxxxxx",
  source: "MI_SOURCE",
  client_name: "Raimundo",
  client_lastname: "Lucas Rodríguez",
  client_email: "...@...",
  country: "ES",
  course_id: "249",
  assessor_first_name: "Ana",
  assessor_last_name: "Lucas Martín",
  assessor_email: "...@mycompany.com",
  minimized: true,
  show_userlist: true,
  company: "UBM",
  unit: "Profesorado"
};
```

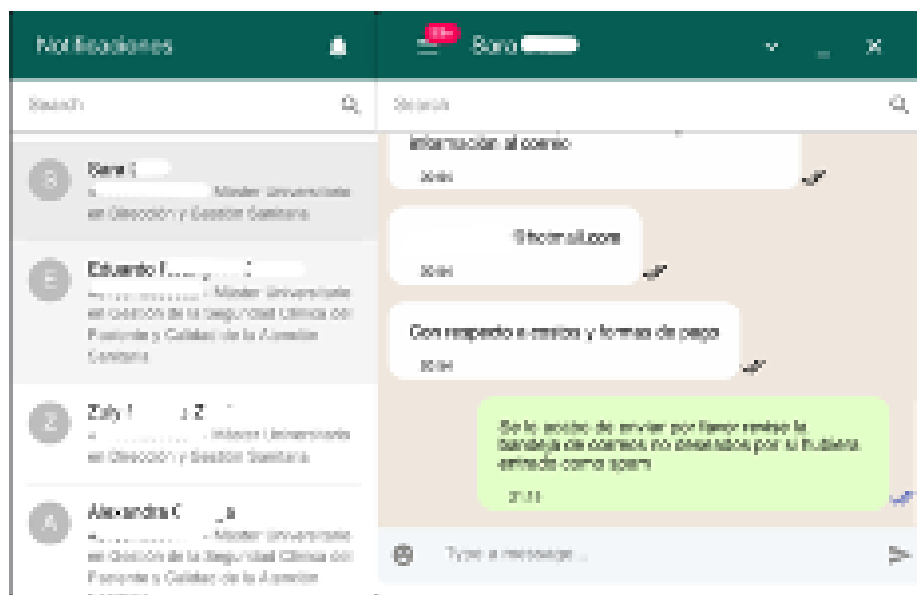
Una vez que tenemos construido el objeto de instancia basta con que iniciemos el componente:

```
<script>
  initChat(var2init);
</script>
```

Otro ejemplo: para instanciar el componente y ver todas las conversaciones de un asesor:

```
var var2init = {
  [1][1]assessor_first_name: "Ana", [1][1]// Nombre del empleado que usa el componente
  [1][1]assessor_last_name: "Lucas Martín", [1]// Apellidos del empleado que usa el componente
  [1][1]assessor_email: "...@mycompany.com", [1]// Email del asesor que manda el/los mensaje
  [1][1]minimized: true, [1][1][1]// True o false, dependiendo si queremos el componente minimizado o no
  [1][1]show_userlist: true,[1][1][1]// True o false, dependiendo si queremos que despliegue la lista de conversaciones
};
```

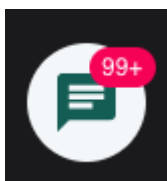
Lo que vais a ver cuando se instancia el objeto chat es, si **minimized** es false y **show_userlist** es true:



Si **minimized** es false y **show_userlist** es false:



Si **minimized** es true:



Para cerrar el componente basta con llamar por javascript al método:
closeChat()

Funcionalidades del servicio

El servicio nos gusta explicarlo como un átomo donde hay funcionalidades que están en el núcleo y que son comunes a todos los clientes y otras, los electrones del átomo, que son intercambiables.

Funcionalidades del núcleo:

1. **Funciones básicas del sistema de mensajería**
 1. Envío
 2. Recepción
 3. Actualización de estados
 4. Gestión de plantillas
 5. Gestión de los "sources", "companies", "unit", etc.
 6. Gestión de horarios comerciales
 7. Gestión de los distintos teléfonos de las empresas
 8. Buscadores generales basados en la transcripción y/o OCR
 9. Activación/desactivación de bots por conversación
2. **Funciones especiales del núcleo que pueden ser reemplazadas para cada cliente:**
 1. Transcripción del audios
 2. OCR de documentos
 3. Análisis de sentimiento

Funciones adicionales: (pueden ser usadas con nuestras herramientas o con las que el cliente decida):

1. Bot transaccionales
2. IA generativa
3. Automatización de campañas/envíos

Funciones específicas para la gestión de mensajes:

1. Todas las funciones específicas en la gestión de mensajes, así como operaciones auxiliares están descritas en el [manual del componente web](#).

Funciones específicas por cliente:

1. Gestión de agendas de contactos
2. Gestión de delegados o sustitutos
3. Personalización de la aplicación móvil
 1. Marca
 2. Login/SSO
4. Integraciones específicas en los procesos de BOTs, IA Generativa, automatizaciones, etc

Los procesos que involucran IA generativa, Bots o procesos de automatización se personalizan para cada caso de uso.