MizuTech WebSIPPhone
One minute Guide

What is it?

The WebSIPPhone (or "webphone") is very similar with a regular VoIP softphone software, with the following differences:
- it can be easily embedded in webpages
- it can be easily customized:
  - you can change it after your needs by modifying its parameters including the display language
  - you can create any kind of user interface with simple HTML/CSS
  - you can control the call logic as you wish with the JavaScript API
  - use a skin or run it in the background

It is implemented as a platform independent java applet compatible with all popular OS (Windows, Linux, MAC) and browsers (Chrome, Firefox, IE, Safari and others). Unlike Flash or HTML5 based solutions, it doesn't need any intermediary server. It acts like a traditional softphone, using the open standard SIP and RTP protocols compatible with all SIP software (server/softswitch/softphone) or device (PBX/IP phone/gateway).
All the common VoIP features are supported (udp/tcp/http,reinvite,hold,forward,transfer,conference,call-recording,video,G729,opus,wideband,dtmf,etc).
The webphone covers a wide range of use starting from a simple click to call button to a fully featured softphone on the web. It can be easily integrated with any kind of applications (such as call center software or voice support) or embedded in devices such as a hardware PBX. The webphone is stable, mature software and new versions (with new features and improvements) are released usually twice a year.

How do I use it?

Deployment in short:
1. Copy the webphone.jar to your webserver and refer to it from anywhere from your html with the "applet" tag. Set at least the "serveraddress" applet parameter to the IP or domain name of your VoIP server.
2. That's all. Your users now can initiate or accept calls.
3. Optional:
   a. Change any of the applet parameters after your needs (these are listed in the documentation)
   b. Create your application logic from JavaScript and/or your own skin (be it a simple click to call button or a full featured user interface). For this you can use/modify any of the skin examples from the demo package. Have a look in the documentation, Java Script API section for the details.

Deployment in details:
All you need is to copy the "webphone.jar" near your html files on your webserver and refer to it from your html using the "applet" tag (place it where you wish).
! Make sure that your webserver doesn't block .jar files. You can test it easily by typing the exact URL to the webphone.jar file into your browser address bar. If works ,then you should be able to download the jar file. Otherwise you will have to add the .jar mime type to the webserver allowed list.
Example (copy-paste this into your html):

```
<applet
 archive  = "webphone.jar"
 codebase = "."
 code     = "webphone.webphone.class"
 name     = "webphone"
 width    = "300"
 height   = "330"
 hspace   = "0"
 vspace   = "0"
 align    = "middle"
 mayscript = "true"
 scriptable = "true"
 alt="Enable or install java: http://www.java.com/en/download/index.jsp"
>
<param name = "serveraddress" value = "SERVER_ADDRESS">
<param name = "loglevel" value = "1">
<param name = "MAYSCRIPT" value = "true">
<param name = "scriptable" value = "true">
<param name = "pluginspage" value = "http://java.com/download/">
<param name = "permissions" value = "all-permissions">


<b>You must enable java or install if not already installed from <a href="http://www.java.com/en/download/index.jsp"> here </a> </b>


</applet>
```

You will have to set at least the "serveraddress" applet parameter. If you don't have a VoIP account (or your own server) then you can register to one of the thousands of available VoIP service providers from all around the world (try ours or search google).

In certain situations you might have to preset the webphone with a sip account and a number to call (for example if you wish to implement a simple click-to-call button). In this case you have to set the following applet parameters accordingly: username, password, autocall, callto. Otherwise you can let the users to type their login username/password.

In case if you wish to create a custom skin (as simple as a button or as complex as a full featured softphone) you will need to use the webphone JavaScript API. This means that you will have to embed the webphone as we have described above, but you will hide it's default user interface (set the applet width and height to 1), display your own html user interface and control the webphone instance (running in the background) with JavaScript function calls. For this you can use one of the skins what we include with the webphone or create your own skin from scratch. Also it is possible to modify on of the existing skins after your needs, or just use the wp_common.js to simplify the interaction with the JavaScript API.

Please note that it is perfectly fine to run the webphone without any html skin. You can also run it in the background driven by your logic, embed it in hardware devices or use its API directly from your applications (Java lib/SDK).

For example making a call with the webphone is as simple as this from javascript:
    -insert the webphone applet tag to wherever you wish into your html
    -obtain the webphone handle (see the initcheck function in the skin examples)
    -optionally add a "webphonetojs" javascript function where you can handle notifications from the webphone such as "connecting" or "ringing"
    -call API_Register(server,username,password)   (this step is also optional since you can set the webphone to auto register via the applet parameters)
    -call API_Call(-1,numbertocall) to initiate the outgoing call
    -call API_Hangup(-1) to disconnect the call.

Explore

You should check the webphone homepage if you haven't done it yet for a longer presentation.

Have a look at the demo pages or screenshots to see how it looks like.

Download the demo package and start to use it on your website or from your application. The demo package contains everything you need to get started with the webphone (the webphone.jar itself, some additional modules, the documentation and the examples/skins).

In the documentation you can find:
    -more details about the webphone (features, usage)
    -list of the applet parameters (in case if you wish to change/fine-tune the default settings)
    -JavaScript API details (if you wish to create your own skin)

The demo version has some limitations such as a 100 second call limit after some calls and a trial limit, but it should be perfectly fine during development.

You will need a license to be able to use the webphone in production. The pricing can be found here. On your payment we will send your webphone build within one work-day.

For any enquiries (including help, sales or technical questions) write to us: websipphone@mizu-voip.com.