Web chat a Click to call

The entire client webchat is made up of several elements:

- initialization (always inserted into client pages)
- template (html / css)
- control JS script (from daktela pbx)

Custom template

If you want to use some special design or functionality, you need to add next client initialization variable, "daktelaConfig.template", with the value:
0 = do not download any template, the template is already implemented directly in the site html
1 = default template and behavior downloaded from the pbx
"chat-pattern.tpl" = template file name, including the entire path of its location (relative or absolute)

Therefore, the template file should be saved to the webhosting where the webchat will run or any other webhosting, and it is made up of css and html parts that can be changed depending on the required functionality.

Using on external webhosting
⚠

When using a template from a webhosting other than the client's web site, it is necessary to send these headers to the template

```
header('Access-Control-Allow-Origin: *');
header('Access-Control-Allow-Methods: GET');
```

Template template can be obtained from the daktela pbx:
In the "Manage" > "Queues" section, the "HTML wizard" link is in the chat queue line. In the wizard's popup window, the "Download this template" button is on the right side of the chat preview.

Advantages and disadvantages

- You can solve switching and translating chat templates into your own language versions.
- You can solve online / off-line switching (working time) yourself.
- When updating daktela to the next version where the template changes, you also need to change your own template.

Special parameters

All client script parameters:

```
daktelaConfig.server = ''; // absolute url of our daktela pbx
daktelaConfig.queue = {}; // object with settings of queue, is possible add another properties which are
othervise loaded from /external/web/api/queue.json, the same json object can be inserted in custom template in
element ith id="daktela-web-queuedata"

daktelaConfig.template = 1; // 0 = dont load, template is in website html, 1 = load template from daktela,
'url_string' = read custom template from external file
daktelaConfig.readinfo = 0; // 0 = dont read, data are in template, 1 = read from daktela immediately / from
localStorage
daktelaConfig.autoinputs_chat = {}; // object with strings which will be automaticaly used in starting inputs,
key is name of input, value is value of input, if autoinputs is set, there is no question for title, email and
phone, client can directly write message
daktelaConfig.autoinputs_ctc = {}; // object with strings which will be automaticaly used in starting inputs,
key is name of input, value is value of input
daktelaConfig.templates = {}; // visual templates for javascript inserted html properties are: customInput,
queueOption, nextCallOption, messageUser, messageUserFile, messageSystem, messageTyping, messageImages,
messageFiles, chatUserImage, bubbleImage
daktelaConfig.events = { // object with event functions for customers scripts, properties are:
        template: function(){}, // is after template is loaded and can be used for custom changes of template
/texts,
        running: function(){}, // is after all scripts/html/events of chat are active,
        update: function(data){} // is after every chat update returned from server
};
```