

LiveChat Agent to Visitor Call

Date: 9 June 2021

Approaches

Approach 1

User Flow (Mobile View - only new tab/window)

1. According to the current designs, once the agent initiates a call, a session will be created and stored in the form of a message with the following details.

```
{
  "_id" : "CLcSj3ijxWrc4xd5y",
  "t" : "livechat-call-session",
  "startTime": "", // When the visitor accepts the call
  "endTime": "", // When the visitor declines or any one of them ends the call
  "msg" : "",
  "groupable" : false,
  "ts" : ISODate("2021-06-06T08:27:14.044Z"),
  "u" : {
    "_id" : "jvhMNmT9CPeB8793G",
    "username" : "guest-19",
    "name" : "MobileViewForAgent"
  },
  "rid" : "Hv7PRmGPGkTo464DJ",
  "_updatedAt" : ISODate("2021-06-06T08:27:14.072Z"),
  "mentions" : [],
  "channels" : []
}
```

2. The agent will be automatically redirected to a new tab - `rocket.chat/meet/CLcSj3ijxWrc4xd5y`
[URL format - `rocket.chat/meet/<livechat-call-session_message_id>`]
3. At the same time, the visitor will receive a ring in livechat, and upon acceptance (`startTime` will be updated in the livechat-call-session message), the visitor will be redirected to a new tab - `rocket.chat/meet/CLcSj3ijxWrc4xd5y`.
4. On `rocket.chat/meet/CLcSj3ijxWrc4xd5y` for both the users - agent and visitor, the meet application will be served.
5. Once the meet app is loaded on both the users' end, the webRTC connection establishment process will start.
6. It will be served on `/meet`.

User Flow (Desktop View)

1. For the agent, the desktop view user flow will be the same as the mobile view.
2. For the visitor, the maximized mode will be same as the mobile view(new tab/window redirection). There are two ways to implement minimize mode.
 - a. Iframe inside the livechat.
 - b. Replicate the meet code in livechat.

Challenges

1. No provision to minimize back.
2. Creating another ddp connection for signaling purposes, auth, and session updation purposes which are already implemented in livechat.
3. In desktop view, if I try to maximize in between the call then the meet app will load and my webrtc connection will need to be re-established. In such a case, upon maximizing, the agent will be interrupted with reconnecting alert.
4. For a minimized view, either we'll replicate the meet app in livechat or it will be an iframe(minimized meet) inside an iframe(livechat).

Questions

1. Is there any advantage of going ahead with this approach by creating a separate standalone meet app unless we are targeting/aim to build a call service in long term as a standalone product like gmeet, jitsi, zoom, etc or use beyond just livechat and rc channels?

Approach 2

User Flow (Desktop View)

1. Once the agent initiates a call, a session will be created and stored in the form of a message with the following details.

```
{
  "_id" : "CLcSj3ijxWrc4xd5y",
  "t" : "livechat-call-session",
  "startTime": "", // When the visitor accepts the call
  "endTime": "", // When the visitor declines or any one of them ends the call
  "msg" : "",
  "groupable" : false,
  "ts" : ISODate("2021-06-06T08:27:14.044Z"),
  "u" : {
```

```
    "_id" : "jvhMNmT9CPeB8793G",
    "username" : "guest-19",
    "name" : "MobileViewForAgent"
  },
  "rid" : "Hv7PRmGPGkTo464DJ",
  "_updatedAt" : ISODate("2021-06-06T08:27:14.072Z"),
  "mentions" : [],
  "channels" : []
}
```

2. A call overlay will cover the screen.
3. At the same time, the visitor will receive a ring in livechat, and upon acceptance (startTime will be updated in the livechat-call-session message), the minimized overlay will appear as per the current design.
4. The webRTC connection establishment process will start.

User Flow (Mobile View)

1. For agents, the user flow and experience will remain the same.
2. For visitors, instead of redirecting to a new tab, a call overlay covering the whole screen (Livechat in mobile) will appear with the option to minimize as well.

Advantages

1. Retaining user experience during minimize/maximize won't be a challenge anymore.
2. The ability to resize and drag maximized mode just like a new window/tab will be retained.
(https://www.w3schools.com/howto/howto_js_draggable.asp)
3. No need to create another ddp connection(signaling)/auth as we can use the already implemented livechat solution.

Challenge

1. It will be coupled with the present codebase of livechat and core rocket.chat. It won't be available as a separate standalone service if we wish to expand in the future.