|  |  |  |  |
| --- | --- | --- | --- |
| gsma_logo_colour_rgb_sm | | RCSJTA TT CR\_012  IM/Chat API calling flow | |
| Meeting Information | | | |
| Meeting Name and Number | | TT | |
| Meeting Date | | May 3, 2013 | |
| Meeting Location | |  | |
| Document Information | | | |
| Document Author(s) | | Qualcomm Incorporated | |
| Document Creation Date | | May 21, 2013 | |
|  | | Approval | X |
| This document is for: *(mark X as appropriate)* | | Discussion |  |
|  | | Information only |  |
| Security Classification – Non Confidential / Confidential GSMA Material (Delete as appropriate and delete the rows below that do not apply) | | | Can be distributed to: (mark X as appropriate or specify group |
| Non Confidential | | Public |  |
| Confidential | | Project Team or Group | X |
| Confidential | | GSMA HQ Staff | X |
| Confidential | | GSMA Full Members | X |
| Confidential | | GSMA Associate Members | X |
| Confidential | | GSMA Rapporteur Members | X |
| Document Summary | | | |
| Desc | | | |
| Document History | | | |
| Date | Version | Author / Comments and for updates description of changes | |
| May 3, 2013 | 1.0 | 1. Initial version | |

© GSMA 2010. The GSM Association (“Association”) makes no representation, warranty or undertaking (express or implied) with respect to and does not accept any responsibility for, and disclaims liability for the accuracy or completeness or timeliness of the information contained in this document. The information contained in this document may be subject to change without prior notice. This document has been classified according to the GSMA [Document Confidentiality Policy](https://infocentre.gsm.org/cgi-bin/prddets.cgi?274175). GSMA meetings are conducted in full compliance with the GSMA [Antitrust Policy](https://infocentre.gsm.org/cgi-bin/docdisp.cgi?275305).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| RCC TF Change Request Form | | | | |
| Title: |  | | | |
| Type of Change Request: Mark the appropriate box by “X” | New Feature |  | Major   |  | | --- | | X |   Change Errata |  |
| Targeted Document | joyn Terminal API Specification | | | |
| Impact on Other Official Documents: | None | | | |

**--------------------------------------Changes below-----------------------------------**

### IM/Chat API

This API exposed all functionality for the Instant messaging/chat Service. It allows:

* Sending messages to a contact.
* Starting group chats with a predefined participants list with an optional subject.
* Joining existing group chats.
* Rejoining existing group chats.
* Restarting a previous group chat.
* Extends a 1-1 chat to a group chat.
* Sending messages in a group chat.
* Leaving a group chat.
* Adding participants to a group chat.
* Retrieving information about a group chat (status, participants and their status)
* Receiving notifications about incoming messages, “is-composing” events, group chat invitations and group chat events.
* Accept/reject an incoming chat invitation.
* Displaying chat history (messages and group chats).
* Erasing chat history by user, by group chat, or by single messages.
* Marking messages as displayed.
* Receiving message delivery reports.
* Read configuration elements affecting IM.

Note: a chat (single/group) is identified by a unique conversation ID which corresponds to the “Contribution-ID” header in the signalling flow. This permits to have a permanent chat or group chat like user experience.

#### IM/Chat API calling flow

The figures in this section contain basic call flows of the IM/Chat service API.

##### Session establishment

Figure 4.5.5.1.1-1 is an example that shows the flow for a RCS client establishing a IM/Chat session with a remote contact.



**Figure 4.5.5.1.1-1: Establish a one-to-one chat session with remote party**

1. The RCS client instantiates a service instance of the IM/Chat Service, establishes a connection with the IM/Chat Service and associates the listener with this RCS client.
2. By selecting a user, opening the IM window and typing a message, the user initiates a IM/Chat session request to the remote party
3. RCS service sends the session request to remote party
4. The session request is delivered to the remote party and a confirmation of delivery of the message is received by the RCS service. This delivery confirmation is provided to the RCS client.
5. By opening the chat window and viewing the incoming message, the remote party accepts the chat session and sends a confirmation of message display
6. RCS service provides the message displayed notification to the RCS client
7. Using procedures described in Figure 4.5.5.1-y, the two users exchange IM / Chat messages

##### Incoming session request

Figure 4.5.5.1.2-1 is an example that shows the flow for a RCS client receiving a IM/Chat session request from a remote contact.



**Figure 4.5.5.1.2-1: Establish a one-to-one chat session with remote party**

1. The RCS client instantiates a service instance of the IM/Chat Service, establishes a connection with the IM/Chat Service and associates the listener with this RCS client.
2. IM/Chat Service receives a request for a session from a remote party
3. IM/Chat Service notifies the RCS client through the invocation of the appropriate listener function
4. RCS client notifies the user by displaying the message received in the incoming session invitation to the user. The user accepts the session request,

OPEN ISSUE: How does the RCS client get the message in the received SIP INVITE request?

1. RCS client retrieves the chat object associated with this session.
2. RCS client installs the listener function for this chat session.
3. RCS client requests the IM/Chat service to send to the remote party, an indication of display of the message to the user.
4. Using procedures described in Figure 4.5.5.1-y, the two users exchange IM / Chat messages

##### Message exchange after chat session establishment

Figure 4.5.5.1.3-1 is an example that shows the flow for a RCS client receiving a IM/Chat session request from a remote contact.

 **Figure 4.5.5.1.2-1: Establish a one-to-one chat session with remote party**

1. A party and B party have already established a IM/Chat session
2. Party A starts to type a new chat / IM message
3. Client A provides an indication of composing event to the IM/Chat service
4. IM/Chat service of party A provides indication of composing toward party B.
5. IM/Chat service of party B provides the indication to the IM client of party B
6. Party A completes typing the message and presses enter
7. Client A provides the message to the IM/Chat service A
8. IM/Chat service A sends the message to IM/Chat service B
9. IM/Chat service B provides an indication to the client B
10. IM/Chat service B provides delivery indication to IM/Chat service A
11. IM/Chat service A provides the delivery indication to IM client A
12. IM Client B displays the message to user
13. IM client B provides the displayed indication to the IM/Chat service B
14. IM/Chat service B provides the displayed indication to IM/Chat service A
15. IM/Chat service A provides the indicaation to the client A

#### Package

Package name **org.gsma.joyn.chat**