

RELEASE NOTES IN-APP MESSAGING SDK VERSION 2.8 - Android

These are the main feature releases available in the **In-App Messaging SDK version 2.8 for Android**.

Version 2.8 planned roll-out: October 1st 2017

New functionalities

[Structured content enablement \(GA in SDK\)](#)

[Automatic messages for messaging](#)

[Unread messages badge](#)

New properties

[Structured content](#)

[Unread messages badge](#)

New APIs

[Unread messages badge API](#)

New functionalities

Structured content enablement (GA in SDK)

Type: Feature

Available to all customers? No - early adopters only

The beta version was released in v2.7 (for a full description, refer to the [v2.7 release notes](#)). The SDK delivers structured content enablement only; the feature will be made fully productive in October. In v2.8 the feature is enabled by default in the SDK.

The dictionary of template elements can be found [here](#).

What does enablement mean?

Until rollout is complete, the structured content capability in SDK v2.7 was flagged as a Beta feature. The feature has an enablement toggle in the SDK which was disabled by default. In SDK v2.8, it is enabled by default.

The toggle may be switched on or off as part of the SDK implementation within the host app, however it is highly recommended not to release the SDK in the host app with structured content enabled until end to end flow has been fully tested on the brand's account.

In-app Messaging SDK toggle -

- Android - `enable_structured_content`

Related properties: [Structured content](#)

The following additional conditions and configurations are required:*

Backend update	Backend enablement	Backend configuration	SDK enablement	SDK configuration
Yes	Yes	Yes	Yes	Yes

Automatic messages for messaging

Type: Feature

Available to all customers? No - early adopters only

Description

Automatic Messages (AKA System Messages) are predefined messages about events that occur in the conversation and are sent to the consumer as the events occur. Their purpose is to gain the consumer's trust in the messaging channel, by setting expectations and giving the consumer visibility over the agent's availability.

Auto messages are triggered upon specific events that are detected by the system (e.g. the consumer opens a new conversation, the conversation is transferred to another agent, the time to respond is updated, etc.). When the auto messages are sent, they are displayed to the consumer and the agent within the conversation transcript, and they also appear in the conversation's history both on the consumer's side and in LE.

Notes:

- Messages are supported in all LiveEngage languages.
- The content of each message can be edited by the brand.
- Skill variation is supported, including enabling/disabling the messages for each skill.
- Certain messages can have different parameters, such as the time the conversation is in the queue before the message is sent.
- Dynamic text can be added to the messages, which will be replaced with a runtime value, such as agent name.
- Auto messages do not affect whom the conversation is pending, nor the time to respond.
- They are filtered out of the reports by default (unless manually included).

The following auto messages are supported:

- New conversations
 - A consumer opens a conversation during working hours
 - A consumer opens a conversation for the first time ever, during working hours
- Off hours
 - A consumer opens a conversation during off hours
 - A consumer opens a conversation for the first time ever, during off hours
 - The consumer sends the first message during off hours in an open conversation
- Time to respond
 - The response time is updated manually by the agent
 - The consumer marks the conversation as urgent

- The consumer dismisses the conversation urgent state
- Consumer/Agent non-responsive
 - The consumer has not responded for X seconds/minutes/hours
 - The agent did not respond for X seconds/minutes/hours
 - Conversation is in queue for X mins/hours
- Transfers and connection to agents
 - The conversation is transferred to a different skill
 - The agent returns the conversation to the queue
 - The consumer is connected to an agent
- Conversation participants
 - Agent manager joins the conversation
 - The joined agent manager leaves the conversation

How to enable auto messages

Auto messages will be enabled for early adopters upon release. Please contact your account manager for more information.

IMPORTANT NOTES:

When auto messages are enabled, they are all enabled by default and all have the default text. It is advised to review them immediately and modify them to suit the brand's needs.

Once auto messages are enabled, the SDK does not show toast messages which were presented in the past.

The following messages remain in the SDK :

- Introduction message from the consumer's first ever conversation. Make sure you do not have a collision between that message and auto messages.
- Conversation resolved message

The following additional conditions and configurations are required*:

Backend update	Backend enablement	Backend configuration	SDK enablement	SDK configuration
Yes	Yes	Yes	Yes	Yes

Unread messages badge

When there are unread messages waiting for the consumer within the brand app, this information can be pushed to display in the app's notification badge. Within the app, brands can develop their own visualization of a badge, such as a number, icon or other marker to show unread messages.

The unread messages number is passed to the SDK through LP Push service with every push.

IMPORTANT NOTES :

The number of unread messages are fetched by the API from the pusher regardless of whether it's registered to the LP push service.

How to enable the unread messages counter

There are two options to set up this counter:

1. If the time condition is met, a REST request is performed to get the counter from the pusher
2. Return the cached number on the app

Parameters:

- conversationQuery: conversationQuery: used to identify the related brand
- completion: called once the operation ends successfully
- failure: called once the operation failed

Related properties: [Unread messages badge](#)

Related API: [Unread messages badge API](#)

The following additional conditions and configurations are required*:

Backend update	Backend enablement	Backend configuration	SDK enablement	SDK configuration
Yes	No	No	No	Yes

* Key for items as follows:

Backend update: This feature requires an update to the backend.

Backend enablement: This feature requires items to be toggled on in the backend.

Backend configuration: This feature requires configuration in the backend.

SDK enablement: This feature requires items to be toggled on in the SDK.

SDK configuration: This features requires items to be configured in the SDK.

New properties

Structured content

The following properties for structured content can now be configured:

Name	Description	Default
<bool name="enable_structured_content">	Enable or Disable toggle for Structured Content feature in conversations.	True

New APIs

Unread messages badge API

```
public static void getNumUnreadMessages(String appId, final ICallback<Integer,
Exception> callback) {
    if (!isValidState()) {
        callback.onError(new Exception("SDK not initialized"));
    } else {
        MessagingFactory.getInstance().getController().getUnreadMessagesCount(mBrandId,
appId, callback);
    }
}
```

To use this API, the SDK must be after initialization.

This document, materials or presentation, whether offered online or presented in hard copy ("LivePerson Informational Tools") is for informational purposes only. LIVEPERSON, INC. PROVIDES THESE LIVEPERSON INFORMATIONAL TOOLS "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

The LivePerson Informational Tools contain LivePerson proprietary and confidential materials. No part of the LivePerson Informational Tools may be modified, altered, reproduced, stored in or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), without the prior written permission of LivePerson, Inc., except as otherwise permitted by law. Prior to publication, reasonable effort was made to validate this information. The LivePerson Information Tools may include technical inaccuracies or typographical errors. Actual savings or results achieved may be different from those outlined in the LivePerson Informational Tools. The recipient shall not alter or remove any part of this statement.

Trademarks or service marks of LivePerson may not be used in any manner without LivePerson's express written consent. All other company and product names mentioned are used only for identification purposes and may be trademarks or registered trademarks of their respective companies. LivePerson shall not be liable for any direct, indirect, incidental, special, consequential or exemplary damages, including but not limited to, damages for loss of profits, goodwill, use, data or other intangible losses resulting from the use or the inability to use the LivePerson Information Tools, including any information contained herein.

© 2017 LivePerson, Inc. All rights reserved.