



CHATLY

Chatly

Manejo de Mensajes en Tiempo Real

Versión <1.0.0>



CHATLY

© 2023 Chatly
TODOS LOS DERECHOS RESERVADOS

Queda reservado el derecho de propiedad de este documento, con la facultad de disponer de él, publicarlo, traducirlo o autorizar su traducción, así como reproducirlo total o parcialmente, por cualquier sistema o medio.

No se permite la reproducción total o parcial de este documento, ni su incorporación a un sistema informático, ni su locación, ni su transmisión en cualquier forma o por cualquier medio, sea éste escrito o electrónico, mecánico, por fotocopia, por grabación u otros métodos, sin el permiso previo y escrito de los titulares de los derechos y del copyright.

FOTOCOPIAR ES DELITO.

Otros nombres de compañías y productos mencionados en este documento, pueden ser marcas comerciales o marcas registradas por sus respectivos dueños.



Información General

TÍTULO: Chat Web Grupal - Chatly

SUBTÍTULO: Caso de uso General o de Contexto

VERSIÓN: 1.0.0

AUTOR: Grupo N° 11

Firmas y Aprobaciones

ELABORADO POR: Grupo N°11
Equipo de desarrollo

FECHA: 2024-11-18

Firma: _____

REVISADO POR: Ing. Ángel Marcelo Rea Guamán.
Docente

FECHA: 2024-11-18

Firma: _____

Lista de Cambios

VERSIÓN	FECHA	AUTOR	DESCRIPCIÓN
1.0.0	2024-11-18	Grupo N°11	Emisión Inicial



Contenido

1 Chat Web Grupal - Manejo de Mensajes en Tiempo Real	1
1.1 Breve Descripción	1
2 Definiciones, Acrónimos, Abreviaturas	1
3 Actores	1
4 Precondiciones	1
5 Flujo de Eventos	2
5.1 Flujo Básico	2
5.2 Sub-Flujos	2
5.3 Flujos Alternos	3
6 Reglas del Negocio	3
7 Requerimientos Especiales	3
8 Poscondiciones	3
9 Relaciones	4
10 Modelamiento	5
10.1 Diagrama de Actividades	5
11 Anexos	6
11.1 Anexo 01. Caso de Uso Gestión de Salas Grupales - Chatly.	6



Chat Web Grupal - Chatly

1 Chat Web Grupal - Manejo de Mensajes en Tiempo Real

1.1 Breve Descripción

Este caso de uso describe el proceso de envío y recepción de mensajes entre usuarios dentro de una sala de chat grupal, garantizando la transmisión en tiempo real y las confirmaciones de entrega. El sistema utiliza WebSockets para una comunicación bidireccional instantánea, asegurando que los mensajes se entreguen sin demoras y que los usuarios reciban notificaciones sobre el estado de sus mensajes.

2 Definiciones, Acrónimos, Abreviaturas

- **WebSocket:** Protocolo de comunicación bidireccional que permite el envío y la recepción de mensajes en tiempo real entre cliente y servidor.
- **Usuario A (Remitente):** Usuario que envía el mensaje.
- **Usuario B (Destinatario):** Usuario que recibe el mensaje.
- **Confirmación de Entrega:** Notificación enviada al Usuario A para informar que el mensaje fue entregado correctamente.
- **Servidor:** Sistema encargado de la transmisión de mensajes entre los usuarios.
- **Base de Datos:** Puede almacenar mensajes, pero no es esencial en este caso, ya que el enfoque es en la transmisión en tiempo real.

3 Actores

- **Usuario A (Remitente):** Envía un mensaje a un destinatario dentro de la sala de chat.
- **Usuario B (Destinatario):** Recibe el mensaje enviado por el Usuario A.
- **Sistema (Servidor):** Retransmite el mensaje de forma inmediata a los usuarios conectados.
- **Base de Datos (BDD):** No almacena los mensajes, salvo que se requiera por alguna funcionalidad específica.

4 Precondiciones

- El usuario debe estar autenticado y ser parte de la sala de chat.
- Los usuarios deben tener una conexión activa a través de WebSocket.
- El servidor debe estar operativo y en funcionamiento para gestionar la transmisión de mensajes.



5 Flujo de Eventos

5.1 Flujo Básico

1. **Escribir y Enviar Mensaje:** El Usuario A escribe un mensaje en la sala de chat y lo envía a través del WebSocket.
2. **Recepción por el Sistema:** El sistema recibe el mensaje y verifica si el Usuario B (destinatario) está conectado.
3. **Retransmisión del Mensaje:**
 - Si el Usuario B está conectado, el sistema retransmite el mensaje en tiempo real.
 - El sistema envía una confirmación de entrega al Usuario A para informarle que el mensaje fue entregado.
4. **Notificación de Entrega Fallida:**
 - Si el Usuario B no está conectado, el sistema informa al Usuario A que el mensaje no pudo ser entregado

5.2 Sub-Flujos

SF01: Validación de Conexión del Destinatario

- El sistema verifica si el Usuario B (destinatario) está conectado a través del WebSocket.
 - Si **el Usuario B está conectado**:
 - El sistema procede a retransmitir el mensaje en tiempo real.
 - El sistema envía una **confirmación de entrega** al Usuario A.
 - Si **el Usuario B no está conectado**:
 - El sistema notifica al Usuario A que el mensaje no pudo ser entregado debido a la desconexión del destinatario.

SF02: Retransmisión del Mensaje

- El sistema recibe el mensaje del Usuario A.
- El sistema envía el mensaje al Usuario B, asegurando que la transmisión se realice en tiempo real sin demoras.

SF03: Confirmación de Entrega

- Si el mensaje fue entregado correctamente al Usuario B:
 - El sistema envía una confirmación de entrega al Usuario A para notificarle que el mensaje fue entregado.

SF04: Reintento de Entrega (En caso de fallo de conexión)

- Si hay un fallo en la conexión del Usuario A o del servidor:
 - El sistema intenta reenviar el mensaje.
 - El sistema informa al Usuario A sobre el fallo de conexión.
 - Si la conexión se restablece, el sistema intenta nuevamente enviar el mensaje.
 - Si los reintentos fallan, el sistema notifica al Usuario A que no se pudo entregar el mensaje.



5.3 Flujos Alternos

FA01: Fallos en la Conexión (Envío Fallido): Si ocurre un fallo en la conexión entre el Usuario A y el servidor (por ejemplo, desconexión temporal):

- El sistema informa al Usuario A sobre los problemas de conexión.
- Se realiza un intento de reintento de entrega hasta que se logre enviar el mensaje o se alcance el número máximo de reintentos configurado.

6 Reglas del Negocio

- Los mensajes deben ser enviados y recibidos en tiempo real mediante WebSockets.
- El sistema no almacena los mensajes de forma persistente a menos que se indique lo contrario.
- El sistema debe enviar una confirmación de entrega al remitente si el mensaje se transmite correctamente al destinatario.
- El sistema debe gestionar los errores de conexión y reintentar la entrega de mensajes cuando sea necesario

7 Requerimientos Especiales

- El sistema debe ser capaz de manejar múltiples conexiones WebSocket de forma concurrente.
- Debe implementarse un mecanismo de reintento de entrega en caso de que el destinatario no esté conectado.
- El servidor debe poder gestionar la notificación de fallos de entrega de manera eficiente.
- La confirmación de entrega debe ser enviada de forma rápida y confiable.

8 Poscondiciones

- **Si la acción es exitosa:** El Usuario A recibe una confirmación de entrega y el mensaje llega en tiempo real al Usuario B.
- **Si la acción falla:** El Usuario A es notificado que el mensaje no fue entregado, y el sistema puede intentar reintentar el envío si es necesario.



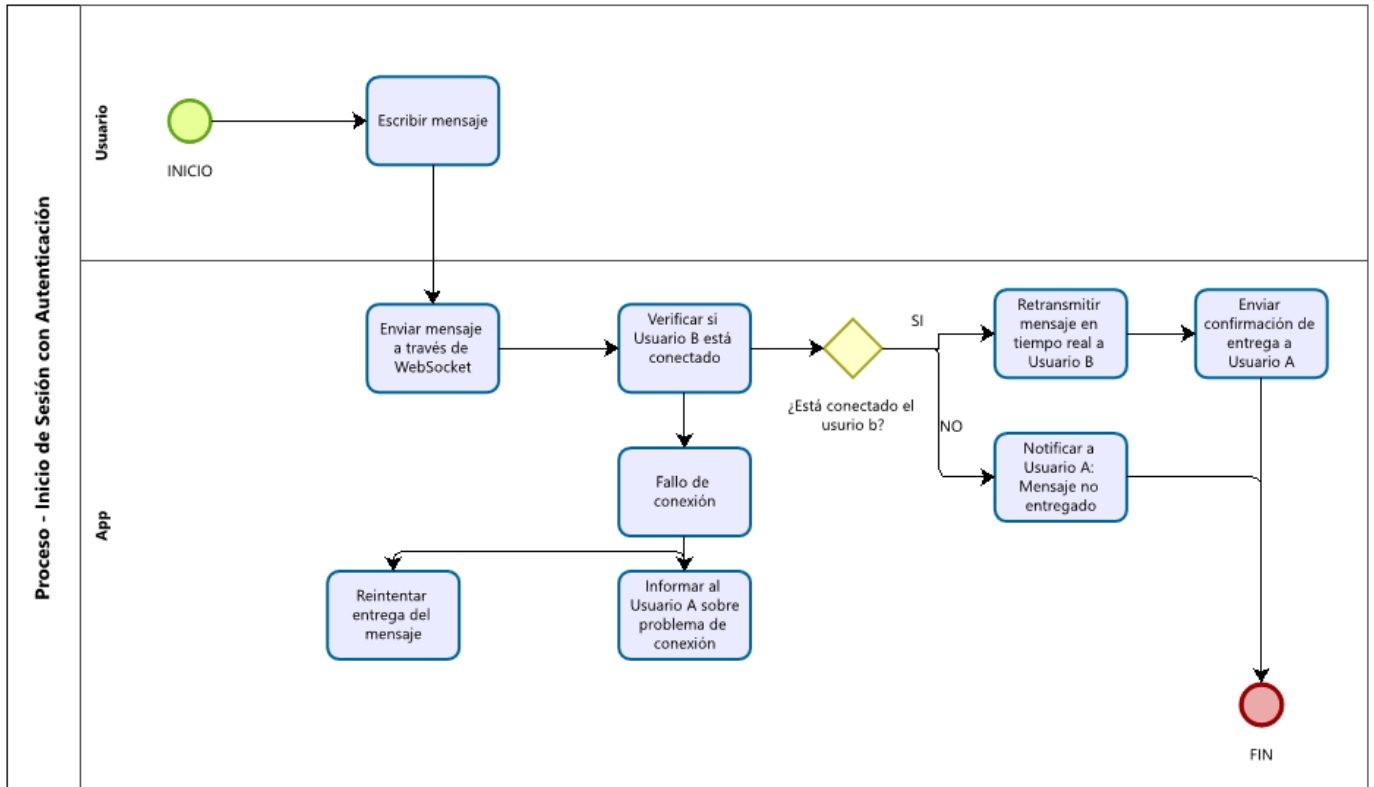
9 Relaciones

Nombre Caso de uso	Obligatoriedad
Gestión de Salas Grupales de Chat	si
Autenticación de Usuarios	si
Manejo de Conexiones WebSocket	si



10 Modelamiento

10.1 Diagrama de Actividades





11 Anexos

11.1 Anexo 01. Caso de Uso Gestión de Salas Grupales - Chatly.

