

FORMATO DE SOLICITUD DE DESPLIEGUE

INFORMACION			
NOMBRE	Ramón Darío Rozo Torres	JEFE INMEDIATO	BRAYAN YANEZ
EQUIPO	MONTECHELO	CARGO	DEVELOPMENT AUTOMATION
CORREO	ramon.rozo@montechelo.net		
FECHA SOLICITUD	18/11/2025	AMBIENTE	PRODUCCIÓN
DESPLIEGUE			
PROYECTO	Widget Chat Web ETB - IDARTES		
MODULO		CODIGO MERGE	
RAMAS	master		
TITULO		URL TRELLO	
DESCRIPCION	Este deploy tiene como objetivo desplegar el desarrollo inicial realizado para idartes cuya fecha de entrega se definió para el 21 de noviembre y por temas administrativos se adelantó para el 18 de noviembre.		
PROCEDIMIENTO DESPLIEGUE	<ol style="list-style-type: none"> Documentación: Tener a la mano tanto el conecting string como el README del proyecto, el presente formato. Desplegar el sistema: Según el repositorio (https://dev.azure.com/MontecheloPipelines/SquadMiosV2/_git/WidgetWeb_IDEARTES) git pull origin rama master. Instalar librerías: Dentro de la carpeta V1 ejecutar el comando npm i Variables de entorno: Crear el archivo .env basado en el .env.example, dejo un ejemplo en el apartado de este documento como guía final de como debe quedar. Base de datos: Se debe crear completamente desde cero en el siguiente orden: <ul style="list-style-type: none"> - Primero el script de la tabla que corresponda. - Segundo el script de la tabla histórico que corresponda. - Tercero el script de los trigger que corresponda. En el apartado dejo un ejemplo y se adjuntan los respectivos script en el correo. 		

<div><div></div><div><div><div>6. Levantar servicios:</div><div>Ejecutar según configuración en servidor o por ambiente bajo los comandos especificados.</div></div><div><div>Nota:</div><div>Garantizar permisos totales sobre la carpeta uploads y de ahora en adelante salvarla ya que contendrá información sensible para el sistema.</div></div><div><div>Una vez finalizado el deploy</div><div>abrir el archivo widget_chat_web_etb_idartes_pro.html en un navegador garantizando para probar funcionamiento.</div></div></div></div>				
MENUS				
INDICE	NOMBRE	APLICATIVO	URL	ICONO
	ETB - IDARTES	Widget		
APLICATIVO				
NOMBRE		KEY	ENDPOINT	REQUIERE HORARIO
				No
ROLES				
INDICE	NOMBRE		LLAVE	MENUS ASIGNADOS
VARIABLES DE ENTORNO		<pre># ! ===== ===== ===== # ! CONFIGURACION DE VARIABLES DE ENTORNO # ! ===== ===== ===== # @Autor: Ramón Dario Rozo Torres # @Última Modificación: Dario Rozo # @version 1.0.0 # .env # TODO: CONFIGURACION DEL PROYECTO PROJECT_CLIENT=IDARTES PROJECT_TIPO=WIDGET PROJECT_NAME=widget-chat-web-etb-idartes PROJECT_VERSION=v1 PROJECT_ENV=DEV # TODO: CONFIGURACION DEL SERVIDOR APP_PORT=5006 APP_URL=http://localhost:5006</pre>		

```
# TODO: CONFIGURACION DE LA SESION
SESSION_NAME=WIDGET CHAT WEB ETB IDARTES
APP_SECRET=CONEXION WIDGET CHAT WEB ETB IDARTES

# TODO: CONFIGURACION DE LA BASE DE DATOS
DB_HOST=localhost
DB_USER=root
DB_PASSWORD='MySQL25#09'
DB_NAME=dbd_widget_chat_web_etb_idartes
DB_PORT=3306
DB_POOL_SIZE=100

# TODO: CONFIGURACION URLS DE CORS (URLS PERMITIDAS
SEPARADAS POR COMAS)
# ALLOWED_ORIGINS=file://,null,http://localhost:5006
ALLOWED_ORIGINS=*

# TODO: API SOUL CHAT
# URL_API_SOUL_CHAT=https://eghujizf9df2.share.zrok.io/api
URL_API_SOUL_CHAT=https://messenger-soulchat-
ideartes.mysoul.software/ideartessoftware/api

# TODO: CONFIGURACION DE RATE LIMITING POR IP
# Configuración general (en minutos)
LIMITE_MINUTOS=1
LIMITE_MAX_PETICIONES=1000

# TODO: CONFIGURACION DE TIEMPO LIMITE CHAT ABIERTOS,
EN HORAS
TIEMPO_LIMITE_CHAT_ABIERTOS=24

# TODO: CONFIGURACION DE SCHEDULER PARA CERRAR
CHATS ANTIGUOS, CADA 30 MINUTOS
# Formato CRON: minuto hora día mes día-semana
# Ejemplos:
# '0 * * * *' = Cada hora (en el minuto 0)
# '* /30 * * * *' = Cada 30 minutos
# '0 * /2 * * *' = Cada 2 horas
# '0 0 * * *' = Cada día a las 00:00
CRON_CERRAR_CHATS_ANTIGUOS= */30 * * * *

# TODO: CONFIGURACION AWS S3 PARA LOGS, subir logs a S3
(true/false)
UPLOAD_LOGS_TO_S3=false
AWS_REGION=us-east-1
AWS_ACCESS_KEY_ID=AKIAXYKJTSSH3VRHA7HC
AWS_SECRET_ACCESS_KEY=Vev/ZEpHT/tQXniGvn/DM0c83HERIM
OHAONQt9JQ
AWS_S3_BUCKET_NAME=bucket-logs-wigetetb-idartes

# TODO: CONFIGURACION SCHEDULER SUBIDA DE LOGS A S3
```

	# Formato CRON: minuto hora día mes día-semana # '0 * * * *' = Cada hora (en el minuto 0) # '*/30 * * * *' = Cada 30 minutos # '0 */2 * * *' = Cada 2 horas S3_UPLOAD_CRON_SCHEDULE=0 * * * * # Eliminar archivos locales después de subirlos a S3 (true/false) S3_DELETE_AFTER_UPLOAD=false		
BASE DE DATOS	Orden de ejecución scripts Tabla Chat 1. create_tbl_chat.sql 2. create_tbl_historico_chat.sql 3. create_triggers_tbl_chat.sql Tabla Mensaje 1. create_tbl_mensaje.sql 2. create_tbl_historico_mensaje.sql 3. create_triggers_tbl_mensaje.sql	LIBRERIAS	Comando npm i dentro de la carpeta v1
OBSERVACIONES	Seguir cuidadosamente el presente documento. Quedamos atentos cualquier duda.		