**FLUJOS DEL CHATBOT**

**PROYECTO:**

**DESARROLLO DE UN CHATBOT CON IA GENERATIVA**

ÍNDICE

[1. Flujo de recolección, filtrado, procesamiento y persistencia de mensajes (formación de la base de conocimiento) 3](#_Toc200301076)

[2. Flujo de captura de preguntas y respuestas en tiempo real (chatbot en producción) 5](#_Toc200301077)

[3. Flujo de una consulta en el chatbot 5](#_Toc200301078)

[Glosario de términos y acrónimos 6](#_Toc200301079)

[Referencias bibliográficas o de internet 7](#_Toc200301080)

**Posibles flujos en el sistema**

1. [Flujo de recolección, filtrado, procesamiento y persistencia de mensajes (formación de la base de conocimiento).](#_1._Flujo_de)
2. [Flujo de captura de preguntas y respuestas en tiempo real en Discord (chatbot en producción).](#_2._Flujo_de)
3. [Flujo de una consulta en el chatbot.](#_3._Flujo_de)

# 1. Flujo de recolección, filtrado, procesamiento y persistencia de mensajes (formación de la base de conocimiento)

Se extraen los mensajes intercambiados entre docentes y estudiantes desde el año 2024 hasta marzo de 2025 inclusive, provenientes del canal de consultas de Discord de la asignatura Diseño de Sistemas. Estos mensajes están almacenados en archivos con formato JSON, lo cual permite:

* **Filtrarlos**, eliminando contenido irrelevante (mensajes vacíos, solo stickers, solo emojis, solo GIFs o símbolos).
* **Procesarlos**, clasificando cada mensaje como pregunta o respuesta, y asociándolos cuando corresponda.
* **Persistirlos**, almacenándolos en una base de datos relacional (PostgreSQL) una vez procesados.

**Estructura de la base de datos relacional**

* **Mensajes**
  + id\_mensaje (PK: Primary Key)
  + id\_mensaje\_discord
  + autor\_id (FK: Foreign Key)
  + fecha\_mensaje
  + contenido
  + es\_pregunta (booleano, por defecto FALSE)
  + origen (ruta del archivo JSON original)
* **Preguntas**
  + id\_pregunta (PK)
  + mensaje\_id (FK)
  + texto
  + esta\_cerrada (booleano, FALSE por defecto)
  + sin\_contexto (booleano, FALSE por defecto; preguntas muy breves o ambiguas)
  + es\_administrativa (booleano, FALSE por defecto; preguntas sobre fechas, grupos, etc.)
* **Respuestas**
  + id\_respuesta (PK)
  + mensaje\_id (FK)
  + pregunta\_id (FK)
  + texto
  + orden (permite reconstruir la secuencia de respuestas)
  + es\_validada (booleano, TRUE solo si el autor es docente)
  + es\_corta (booleano, TRUE si tiene ≤14 caracteres)
* **Autores**
  + id\_autor (PK)
  + nombre\_autor
  + es\_docente (booleano, FALSE por defecto; se determina manualmente)
* **Adjuntos**
  + id\_adjunto (PK)
  + mensaje\_id (FK)
  + url
  + tipo
* **Fragmentos de preguntas**
  + id\_fragmento (PK)
  + pregunta\_id (FK)
  + texto\_fragmento
  + orden (para poder reconstruir la pregunta completa si es necesario)
* **Embeddings**
  + id\_embedding (PK)
  + fragmento\_id (FK)
  + id\_chroma\_db (identificador en la base de datos vectorial)

**Pasos del flujo**

1. A partir de los archivos JSON, se utiliza un **DataFrame** (de la librería Pandas de Python) para representar los mensajes como una tabla, lo que facilita su análisis.
2. Se aplican filtros para eliminar mensajes irrelevantes. Se mantienen registros trazables mediante archivos CSV y logs.
3. Se clasifica cada mensaje como pregunta o respuesta. Las preguntas se cierran cuando:
   * Un alumno agradece,
   * Un docente da una respuesta afirmativa,
   * Se intercambian suficientes mensajes,
   * O pasa un determinado tiempo.
4. Finalmente, se almacena toda la información procesada en la base de datos relacional.

# 2. Flujo de captura de preguntas y respuestas en tiempo real (chatbot en producción)

1. Un usuario (alumno o docente) envía un mensaje en el canal “consultas” de Discord.
   * El mensaje es capturado mediante la API de Discord, utilizando discord.py.
2. El mensaje se analiza con un modelo de lenguaje (como Mistral, vía API o local):
   * Si es una pregunta:
     + Se guarda en las tablas mensajes y preguntas.
     + Se evalúa si el autor ya existe; si no, se agrega a autores.
     + Se procesan adjuntos si existen.
     + Se marca como esta\_cerrada = FALSE.
   * Si es una respuesta:
     + Se busca a qué pregunta abierta pertenece. Si hay muchas, se consideran solo las últimas 5.
     + El modelo intenta determinar la relación entre la respuesta y la(s) pregunta(s).
     + Si la respuesta es válida para más de una pregunta, se asocia a todas.
     + Si se trata de un mensaje de cierre de alumno, la pregunta se cierra solo si ya hay al menos una respuesta validada.
     + Si es cierre de docente, se guarda y se cierra la pregunta.
     + En cualquier otro caso, se guarda como una nueva respuesta.
     + También se procesan adjuntos y se verifica si el autor es nuevo.
3. Al cerrarse una pregunta, se generan los embeddings para incorporarla a la base de datos vectorial y que esté disponible para futuras búsquedas semánticas en el chatbot.

# 3. Flujo de una consulta en el chatbot

1. Un usuario envía una pregunta en Discord.
2. Se genera un embedding de esa pregunta y se consulta la base de datos vectorial (ChromaDB).
3. Se recupera la pregunta más similar y sus respuestas desde la base relacional.
   * Si hay varias respuestas: se combinan con ayuda del modelo de lenguaje (Mistral u otro).
   * Si no hay respuestas previas: se informa que la consulta nunca fue realizada y se invita a preguntar en el canal.

Se presenta la respuesta correspondiente al usuario en el chatbot.

# Glosario de términos y acrónimos

API (Application Programming Interface): La interfaz de programación de aplicaciones consiste en un conjunto de mecanismos que permiten a dos componentes de software comunicarse entre sí mediante un conjunto de definiciones y protocolos. Puede considerarse como un contrato de servicio entre dos aplicaciones el cual define cómo se comunican entre sí mediante solicitudes y respuestas.

Base de conocimiento: Es un depósito centralizado de información que está organizada y es de fácil acceso. Es un almacén de conocimientos acumulados, que captura y selecciona información, instrucciones y respuestas valiosas relacionadas con un tema u organización específicos.

Base de datos relacional: Recopilación organizada de información o datos que se basan en el modelo relacional, una forma de representar datos en tablas. Cada fila en una tabla es un registro con una ID única, llamada clave. Las columnas de la tabla contienen los atributos de los datos y cada registro suele tener un valor para cada atributo.

Base de datos vectorial: Tipo específico de base de datos que guarda la información en forma de vectores multidimensionales.

Chatbot: Es un programa o aplicación con el que los usuarios pueden conversar por voz o texto.

ChromaDB: Base de datos vectorial que permite almacenar y buscar información basada en similitud semántica entre embeddings.

CSV (valores separados por comas): Es un tipo especial de archivo que puede crear o editar en Excel. Permite almacenar datos separados por comas.

DataFrame: Es una estructura de datos que organiza los datos en una tabla bidimensional de filas y columnas, similar a una hoja de cálculo.

Discord: Es una plataforma de comunicación de voz, video y texto.

Embeddings: Es una técnica que convierte información compleja (como palabras, imágenes o incluso comportamientos) en números que una máquina puede entender. Lo hace representando esa información en forma de vectores, es decir, conjuntos de números organizados, dentro de un espacio matemático.

FK (Foreign Key): Es una columna o un conjunto de columnas en una tabla, en una base de datos relacional, cuyos valores corresponden a los valores de la clave primaria de otra tabla. Para poder añadir una fila con un valor de clave foránea específico, debe existir una fila en la tabla relacionada con el mismo valor de clave primaria.

IA (Inteligencia Artificial): Tecnología que permite a ordenadores y máquinas simular las capacidades humanas de aprendizaje, comprensión, resolución de problemas, toma de decisiones, creatividad y autonomía.

JSON (JavaScript Object Notation): Es un formato basado en texto para almacenar e intercambiar datos de una manera que es legible por humanos y analizable por máquina.

Logs: Archivos que registran eventos específicos dentro de un sistema.

LLM (Large Language Model): Modelos de lenguaje de gran tamaño son sistemas de Inteligencia Artificial capaces de comprender y generar lenguaje humano procesando enormes cantidades de datos.

Mistral: Un modelo de lenguaje largo desarrollado por la empresa Mistral (por ejemplo, Mistral 7B).

Pandas: Biblioteca de software de código abierto diseñada específicamente para la manipulación y el análisis de datos en el lenguaje Python (se pueden cargar, alinear, manipular o incluso fusionar datos).

PK (Primary Key): la cave primaria es una columna o un conjunto de columnas en una tabla cuyos valores identifican de forma exclusiva una fila de la tabla en una base de datos relacional.

PostgresSQL: es un potente sistema de base de datos relacional de código abierto que utiliza y amplía el lenguaje SQL.

Similitud Semántica: Es una rama de la inteligencia artificial y el procesamiento del lenguaje natural que se ocupa de la comprensión del significado de palabras y frases. Mide la similitud de significado entre dos textos, más allá de las palabras utilizadas. Se aplica para la recuperación de información, la traducción automática y el análisis de sentimientos, entre otros.

# Referencias bibliográficas o de internet

AWS. (2025, 12 de mayo). *¿Qué es una interfaz de programación de aplicaciones (API)?* Recuperado de <https://aws.amazon.com/es/what-is/api/>

StoryLab.ai. (2025,12 de mayo). *¿Qué es una base de conocimientos y por qué es importante?* Recuperado de<https://storylab.ai/es/what-is-knowledge-base-and-importance/>

Oracle. (2025, 12 de mayo). *¿Qué es una base de datos relacional?* Recuperado de <https://www.oracle.com/ar/database/what-is-a-relational-database/>

Datacamp. (2025, 12 de mayo). *Las 5 mejores bases de datos vectoriales*. Recuperado de: <https://www.datacamp.com/es/blog/the-top-5-vector-databases>

AWS. (2025, 12 de mayo). *¿Qué es un chatbot?* Recuperado de <https://aws.amazon.com/es/what-is/chatbot/>

Chroma. (2025, 12 de mayo). *Chroma.* Recuperado de <https://docs.trychroma.com/docs/overview/introduction>

Microsoft. (2025, 12 de mayo). *Crear o editar archivos .csv para importarlos a Outlook*. <https://support.microsoft.com/es-es/office/crear-o-editar-archivos-csv-para-importarlos-a-outlook-4518d70d-8fe9-46ad-94fa-1494247193c7>

Databricks. (2025, 12 de mayo). *What is a DataFrame?* Recuperado de <https://www.databricks.com/glossary/what-are-dataframes>

Discord. (2025, 12 de mayo). *The Discord Story.* Recuperado de <https://discord.com/company>

Novaluz. (2025, 12 de mayo). *¿Qué son los embeddings y cómo puedes aplicarlos a tu pyme?* Recuperado de <https://novaluz.es/blog/embeddings/>

IBM. (2025, 12 de mayo). *¿Qué es la inteligencia artificial (IA)?* Recuperado de <https://www.ibm.com/mx-es/think/topics/large-language-models>

IBM. (2025, 12 de mayo). *Claves primarias y foráneas*. Recuperado de <https://www.ibm.com/docs/es/ida/9.1.2?topic=entities-primary-foreign-keys>

Oracle. (2025, 12 de mayo). *¿Qué es JSON?* Recuperado de <https://www.oracle.com/ar/database/what-is-json/>

Keepcoding. (2025, 12 de mayo). *¿Qué son los logs y para qué sirven?* Recuperado de <https://keepcoding.io/blog/que-son-logs-y-para-que-sirven/>

Builtin. (2025, 12 de mayo). *Mistral AI: What to Know About Europe’s OpenAI Rival*. Recuperado de <https://builtin.com/articles/mistral-ai>

IBM. (2025,12 de mayo). *¿Qué son los LLM?* <https://www.ibm.com/mx-es/think/topics/large-language-models>

DataScientest. (2025, 12 de mayo). *Pandas: La biblioteca de Python dedicada a la Data Science.* <https://datascientest.com/es/pandas-python>

PostgreSQL. (2025, 12 de mayo). *About PostgreSQL.* <https://www.postgresql.org/about/>