



DOCUMENTACIÓN – Sistema de Boletería con IA – Artechito

ÍNDICE

- 1. Introducción General**
 - 2. Contexto y Problemática del Teatro**
 - 3. Solución Propuesta**
 - 4. Pilares Tecnológicos**
 - 4.1 Supa Base**
 - 4.2 Python**
 - 4.3 n8n (Explicación narrativa integrada)**
 - 5. Arquitectura del Sistema**
 - 5.1 Diseño de Base de Datos**
 - 5.2 Flujo n8n Detallado (narrado)**
 - 6. Integración con Python**
 - 7. Equipo Artechito**
 - 8. Conclusión Final**
 - 9. Presentación del Proyecto**
-

★ 1. Introducción General

El proyecto Sistema de Boletería con IA, desarrollado por el Equipo Artechito en el marco del Centro Pescar ARTECH 2025, busca modernizar la gestión teatral mediante el uso de Inteligencia Artificial, automatización y análisis de datos.

El sistema permite que un usuario realice consultas en lenguaje natural —como “¿cuántas entradas vendimos en septiembre?” o “mostrame la cartelera de la semana que viene”— y reciba una respuesta precisa, un archivo generado o incluso un correo enviado automáticamente.

Todo esto gracias a la integración entre Supabase, n8n, Python y modelos Gemini.

El resultado es una solución moderna, automatizada y eficiente que replica herramientas reales utilizadas en organizaciones profesionales.

🤖 2. Contexto y Problemática del Teatro

Los teatros manejan diariamente información compleja: obras, funciones, ubicaciones, salas, precios, actores y compradores. La administración debe responder consultas internas, generar informes y realizar análisis.

Sin embargo, muchos procesos siguen siendo manuales, dispersos y propensos a errores:

- Consultas que requieren revisar múltiples registros
- Informes que se arman “a mano”
- Falta de automatización para correo, archivos o reportes
- Dificultad para acceder a información histórica

Nuestro sistema aborda esa problemática convirtiendo estas tareas en procesos automáticos, integrados y accesibles mediante lenguaje natural.

3. Solución Propuesta

Implementamos una arquitectura donde el usuario interactúa con el sistema a través de un chat en Python.

Esa consulta se envía a un flujo en n8n que interpreta la intención con IA, ejecuta consultas SQL seguras, genera archivos si es necesario y devuelve una respuesta clara y lista para usar.

La interacción fluye así:

Python → n8n (IA + Automatización) → Supabase →
Respuesta/Reporte

Todo el sistema está diseñado para que el usuario no necesite conocimientos técnicos, sino simplemente formular una pregunta en lenguaje natural.

4. Pilares Tecnológicos

4.1 Supabase – Base de Datos Escalable

Supabase fue seleccionado como motor principal debido a su estructura basada en PostgreSQL, su facilidad de acceso remoto y sus capacidades avanzadas:

- Seguridad integrada
- Relaciones complejas
- Vistas optimizadas
- Alto rendimiento
- Integración perfecta con n8n

En esta base se almacena todo: obras, funciones, salas, ubicaciones, clientes y entradas.

4.2 Python – Consola Inteligente

Python es el punto de interacción con el usuario.

Desde una consola simple, el usuario:

- Escribe su consulta
- Python la envía por POST al flujo de n8n
- Recibe y muestra la respuesta estructurada

El usuario nunca ve SQL, no necesita navegar una base de datos, ni acceder a sistemas externos.

4.3 n8n – Motor de Automatización e Inteligencia

El flujo de n8n es el núcleo lógico del sistema.

Su función es interpretar la intención, ejecutar consultas, generar archivos, enviar correos y devolver resultados limpios.

A continuación se describe su funcionamiento de manera narrativa.

5. Arquitectura del Sistema

5.1 Diseño de Base de Datos

La base de datos se diseñó bajo normas de normalización (3FN), garantizando coherencia y consistencia.

Incluye:

- Obras
- Actores
- Salas
- Ubicaciones
- Funciones
- Precios
- Clientes
- Entradas

Esta estructura permite responder preguntas sobre cartelera, ventas, clientes, funciones y más.

(Aquí podés insertar el diagrama de Supabase)

5.2 Flujo n8n – Explicación Técnica Narrada

El flujo de n8n funciona como un **orquestador inteligente**. Su rol principal es recibir consultas provenientes de Python, interpretarlas mediante un agente de IA, conectarse con la base de datos en Supabase y generar respuestas procesadas para el usuario.

El Agente de IA como intérprete del sistema

Todo comienza cuando el Webhook recibe un mensaje del usuario. Esa información se envía al **Agente de IA**, configurado para actuar no solo como un chatbot, sino también como un generador de SQL seguro. Este agente traduce lenguaje natural a consultas válidas, limitadas exclusivamente a operaciones de lectura (SELECT, COUNT, SUM, AVG).

También detecta la intención del usuario: si necesita un archivo descargable, una planilla interna, enviar por email o subir información a Google Drive.

Ejecución segura sobre Supabase

El SQL generado por la IA pasa luego al nodo de PostgreSQL. Este nodo está configurado únicamente para lectura, evitando cualquier modificación accidental o maliciosa del sistema.

El flujo rechaza tablas inexistentes o campos incorrectos, asegurando integridad.

Automatización inteligente de archivos y reportes

Dependiendo de la intención detectada por la IA, el flujo puede:

- Generar una hoja interna tipo Sheets
- Crear un archivo Excel descargable

- Subir contenido formateado a Google Drive
- Enviar un correo electrónico con datos o archivos adjuntos

Cada una de estas herramientas se integra mediante bloques especializados, como generadores de hojas, convertidores a archivo binario y nodos de Gmail con autenticación OAuth.

Orquestación interna mediante lógica condicional

Uno de los aspectos más importantes del flujo es la capa lógica que se encarga de interpretar los resultados.

Nodos como Switch, If y Parsers analizan el tipo de respuesta solicitada: email, archivo, drive o json. Luego, cada camino procesa la información para devolverla de manera clara y ordenada.

Respuesta final hacia Python

Finalmente, el flujo compone un mensaje limpio y lo envía nuevamente por Webhook Response.

Esto puede incluir:

- Datos estructurados
- Un archivo descargable
- Un link de Google Drive
- Una confirmación de envío por correo

Así, el sistema responde como un asistente administrativo real del teatro.

6. Integración con Python

El script en Python solo necesita enviar un prompt.
El flujo se encarga del resto:

1. El usuario escribe una pregunta.
2. Python la envía al Webhook.
3. n8n interpreta la intención con IA.
4. Ejecuta consultas o genera archivos.
5. Retorna la respuesta a Python.

El usuario experimenta el sistema como un **chat con un asistente inteligente**.

7. Equipo Artechito

“Artechito – Soluciones Inteligentes para Teatros Modernos”

- **Loredana Bejarano** — Arquitecta de Base de Datos
- **Julián Duarte** — Ingeniero de Automatización en n8n
- **Mauricio Cuellar** — Licenciado en Análisis de Sistemas
- **César Mendoza** — Desarrollador Backend Python
- **Rocío Pagliarulo** — Desarrolladora de Base de Datos

8. Conclusión Final

El Sistema de Boletería con IA – Artechito demuestra que la combinación de bases de datos sólidas, automatización avanzada y modelos de IA puede transformar la gestión institucional de un teatro.

La solución automatiza reportes, consultas y operaciones, aportando eficiencia, velocidad y precisión.

El sistema cumple con todos los entregables académicos y propone un enfoque profesional y escalable para organizaciones culturales modernas.

9. Presentación del Proyecto

 <https://artechito.vercel.app>