

UNIVERSIDAD AUTÓNOMA GABRIEL RENÉ MORENO

**FACULTAD DE INGENIERÍA EN CIENCIAS DE LA COMPUTACIÓN Y
TELECOMUNICACIONES**



Tarea – Clase 3

Integrantes: Paúl Jherson Rodríguez Guzmán

Registro: 218046235

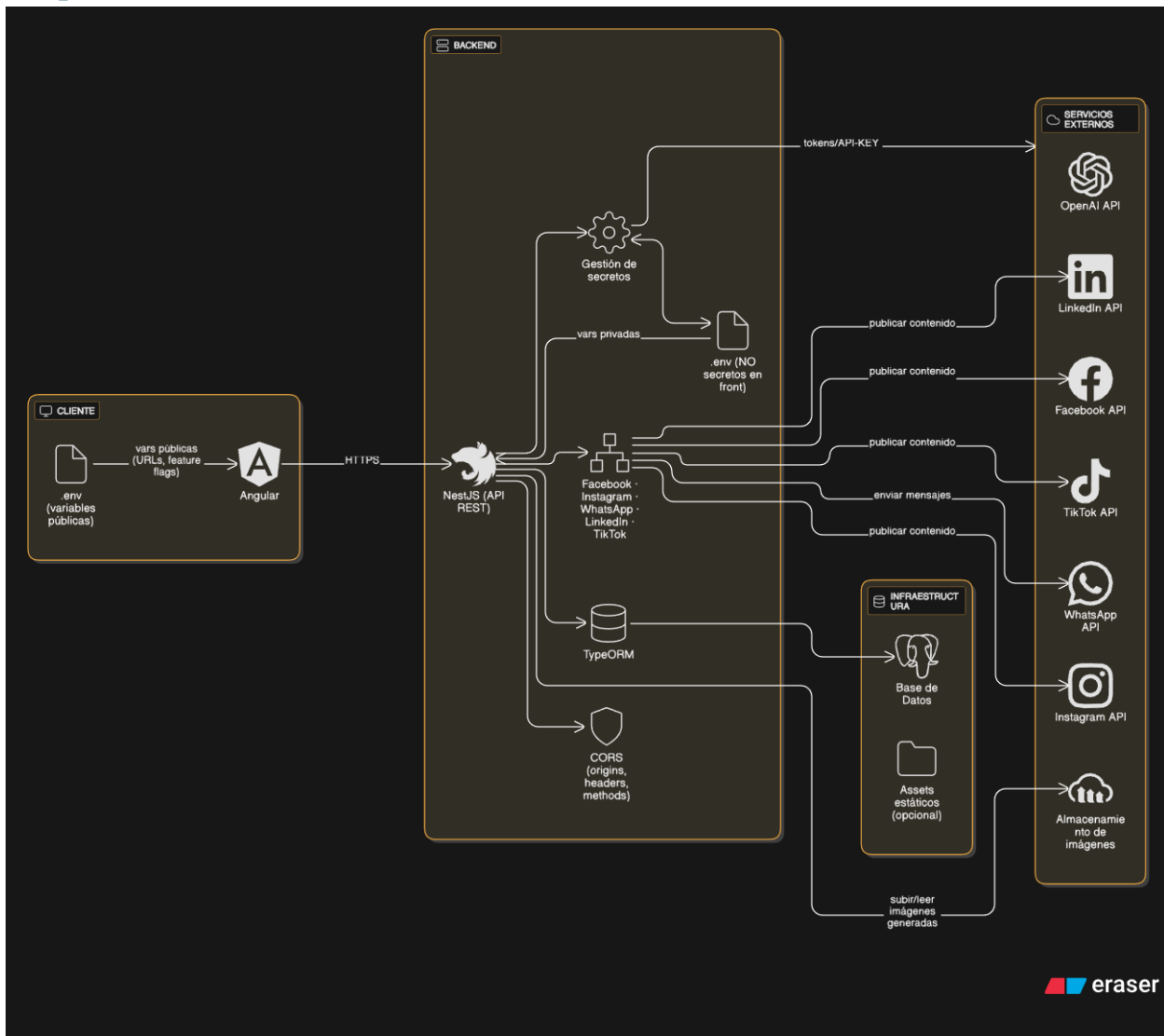
Materia: Tópicos avanzados

Santa cruz – Bolivia

LLM seleccionado y justificación

Elegí las APIs de OpenAI por su combinación de flexibilidad técnica, calidad de interacción y relación costo-beneficio. Permiten definir con precisión el formato de salida (por ejemplo, JSON con atributos específicos), ajustar el comportamiento del modelo con controles finos y cubrir casos de uso complejos con alta fidelidad en lenguaje natural. Además, ya cuento con experiencia previa integrándolas, lo que reduce la curva de aprendizaje y acelera el desarrollo

Arquitectura del sistema



Estrategia de prompts para cada red

Use el recomendado para cada red:

Red	Max Chars	Tono	Hashtags	Emojis	Formato
Facebook	~63,206 (práctico: 300-500)	Casual, conversacional	2-3	✅ Sí	Párrafos, links
Instagram	2,200	Visual, inspiracional	10-30	✅✅ Muchos	Líneas cortas, emojis
LinkedIn	3,000	Profesional, formal	3-5	⚠️ Moderado	Párrafos, bullets
TikTok	2,200	Energético, trending	3-5	✅ Sí	Corto, call-to-action
WhatsApp	4,096 (práctico: 200-300)	Personal, directo	0-1	⚠️ Moderado	Conversacional

Y esto lo engloba un prompt que tiene la siguiente estrategia:

- 1. **Role Assignment (Asignación de Rol)**
 - Define claramente el rol: "Eres un redactor de social media senior"
 - Establece autoridad y expertise en el dominio
- 2. **Constraint-Based Generation (Generación basada en restricciones)**
 - Especifica reglas específicas por red social (importadas desde socialPostsRules)
 - Define límites y requisitos claros para cada plataforma
- 3. **Locale/Language Enforcement (Forzar idioma)**
 - Escribe SIEMPRE en \${locale} asegura consistencia en el idioma de salida
- 4. **Structured Output (Salida estructurada)**
 - Solicita JSON válido que cumpla un schema específico
 - Calcula métricas como character_count
 - Instrucción explícita: "No incluyas nada fuera del JSON"
- 5. **Audience-Aware Adaptation (Adaptación por audiencia)**
 - "Adapta la redacción y el enfoque a cada audiencia"
 - Reconoce diferencias entre plataformas

Challenges encontrados y soluciones

Uno de los problemas que encontré al principio es el de como ahorrar tokens por cada llamada a la API, la solución fue dividirlo en dos, una para el chat con el humano, en esta primera llamada se espera obtener la información necesaria antes de generar los posts, una vez obtenida la información recién se hace otra llamada a otro endpoint para generar los posts. Esto evita que se esté creando posts por cada consulta.