

Planteamiento del problema

En la era digital, los usuarios demandan experiencias más rápidas, precisas y personalizadas. Aunque Youtube ofrece un motor de búsqueda avanzada dentro de su plataforma, este carece de una integración sencilla que permita a los usuarios realizar consultas específicas dentro de un sitio web propio y con la posibilidad de control en alcance geográfico de los resultados.

Actualmente, muchas páginas web integran videos desde Youtube solo muestran contenido generalista sin filtros de ubicación, lo que provoca que los usuarios no encuentren información contextualizada a su región o ciudad. Asimismo, no existe en la mayoría de los casos la opción para activar o desactivar la geolocalización de acuerdo con las necesidades individuales, limitando la flexibilidad en la búsqueda de contenido.

Objetivo general

Desarrollar una página web con integración a la API de Youtube que permita realizar búsquedas de videos personalizados con geolocalización dentro del país o región en este caso solo territorio Mexicano, tomando parámetros de 1m a 1 km para ampliar la búsqueda de videos alrededor de las coordenadas proporcionadas.

Objetivos Específicos

1. Diseñar la estructura de la página web implementando una interfaz intuitiva, adaptable y fácil de usar.
2. Conectar la página con la API de YouTube para obtener resultados dinámicos de búsqueda en tiempo real, utilizando parámetros personalizados.
3. Incorporar un sistema de geolocalización que filtre los resultados de búsqueda por región, ciudad dentro del país.
4. Implementar la funcionalidad de activación y desactivación de filtros de geolocalización.
5. Optimizar la visualización de los resultados, mostrando datos relevantes como título del video, miniatura, canal, duración y enlace de reproducción.
6. Probar y validar el sistema en diferentes navegadores y dispositivos para garantizar su correcto funcionamiento y una experiencia de usuario fluida.

Levantamiento de Requerimientos:

A continuación, se detalla el análisis y la especificación de los requerimientos para el desarrollo de una API que conecte la ubicación de un usuario en Google Maps con sugerencias de videos relevantes de YouTube.

Problema a Resolver Actualmente, no existe una forma directa y automatizada para que los

usuarios descubran contenido de video en YouTube que sea geográficamente relevante para su ubicación actual o un lugar de interés específico en Google Maps. Los viajeros, exploradores urbanos o personas interesadas en un lugar particular, necesitan una herramienta que les permita encontrar videos (documentales, blogs de viajes, reseñas, eventos, etc.) directamente relacionados con los lugares que están viendo en un mapa.

El objetivo es crear un puente entre la exploración geográfica (Google Maps) y el descubrimiento de contenido audiovisual (YouTube), enriqueciendo la experiencia del usuario al ofrecerle una capa de información contextual y dinámica sobre cualquier punto geográfico.

Requerimientos Funcionales (RF)

- 1: Recepción de coordenadas geográficas. La API debe ser capaz de recibir coordenadas geográficas (latitud y longitud) como entrada principal.
- 2: Búsqueda de Videos en YouTube. La API debe utilizar las coordenadas recibidas para consultar la API de YouTube y buscar videos relevantes. La búsqueda debe considerar:
 - Palabras clave asociadas a la ubicación (nombre de la ciudad, monumento, restaurante, etc.).
 - Videos etiquetados con esa geolocalización.
- 3: Integración con Google Maps API. La API deberá poder usar servicios de Google Maps (como Geocoding API) para traducir las coordenadas en nombres de lugares legibles (ej. "Torre Eiffel, París"), y usar estos nombres para refinar la búsqueda de videos.
- 4: Filtrado y Categorización de Resultados. La API debe permitir filtrar los resultados de búsqueda por parámetros como:
 - Relevancia.
 - Fecha de publicación (videos recientes primero).
 - Popularidad (número de vistas).
 - Categoría (ej. "Viajes", "Comida", "Historia", "Música").
- 5: Devolución de Datos Estructurados. La API debe devolver una respuesta en un formato estándar y fácil de consumir (como JSON), conteniendo una lista de videos sugeridos. Cada video en la lista debe incluir:
 - Título del video.
 - URL del video.
 - URL de la miniatura (thumbnail).
 - Nombre del canal.
 - Número de vistas y fecha de publicación.
- 6: Manejo de Búsqueda por Radio. La API debe aceptar un parámetro opcional de radio (ej. 1 metro, 1 kilómetro) para ampliar el área de búsqueda de videos alrededor de las coordenadas proporcionadas.

Requerimientos No Funcionales (RNF)

- 1: Performance y Latencia. El tiempo de respuesta de la API desde que recibe la petición hasta que devuelve los resultados no debe superar los 800 milisegundos en el 95% de los casos para garantizar una experiencia de usuario fluida.
- 2: Disponibilidad. El sistema debe tener una disponibilidad del 99.8%, asegurando que esté operativo y accesible para las aplicaciones cliente en todo momento.
- 3: Escalabilidad. La arquitectura de la API debe ser capaz de manejar un aumento en el número de peticiones concurrentes sin degradar el rendimiento. Deberá soportar inicialmente hasta 1,000 peticiones por minuto, con capacidad de escalar horizontalmente.
- 4: Seguridad.
 - Toda la comunicación entre el cliente y la API debe estar cifrada mediante HTTPS (TLS 1.2 o superior).
 - El acceso a la API debe estar protegido mediante un sistema de claves de API (API Keys) para autenticar y autorizar las peticiones.
 - Se deben implementar medidas para prevenir ataques comunes como inyección de dependencias y Cross-Site Scripting (XSS).
- 5: Usabilidad (para desarrolladores). La API debe contar con una documentación clara y completa, incluyendo ejemplos de peticiones y respuestas (ej. utilizando Swagger o Postman), para facilitar su integración por parte de los desarrolladores.
- 6: Compatibilidad. La API debe seguir los principios RESTful para asegurar la compatibilidad con una amplia gama de clientes HTTP (navegadores web, aplicaciones móviles, otros servidores).
- 7: Manejo de Errores. La API debe devolver códigos de estado HTTP apropiados y mensajes de error claros y descriptivos cuando una petición no pueda ser procesada correctamente (ej. 400 para una petición mal formada, 404 si no se encuentran videos, 500 para errores internos).