

**UNIVERSIDAD CATÓLICA SANTO TORIBIO DE MOGROVEJO**  
**FACULTAD DE INGENIERÍA**  
**ESCUELA DE INGENIERÍA DE SISTEMAS Y COMPUTACIÓN**



**Agente inteligente recomendador de atractivos turísticos en el departamento  
de Lambayeque**

**TESIS PARA OPTAR EL TÍTULO DE  
INGENIERO DE SISTEMAS Y COMPUTACIÓN**

**AUTOR**

**Justo Andres Davila Nieto**

**ASESOR**

**Hector Miguel Zelada Valdivieso**

**<https://orcid.org/0000-0002-2311-4284>**

**Chiclayo, 2026**

**Agente inteligente recomendador de atractivos turísticos en el  
departamento de Lambayeque**

PRESENTADA POR  
**Justo Andres Davila Nieto**

A la Facultad de Ingeniería de la  
Universidad Católica Santo Toribio de Mogrovejo  
para optar el título de

**INGENIERO DE SISTEMAS Y COMPUTACIÓN**

APROBADA POR

Jury Yesenia Aquino Trujillo  
PRESIDENTE

Miguel Orlando Diaz Vidarte  
SECRETARIO

Hector Miguel Zelada Valdivieso  
VOCAL

## **Dedicatoria**

A mi amada esposa, por su apoyo incondicional, paciencia y amor, que han sido mi mayor fortaleza en este camino. A mi hijo, cuya alegría y curiosidad me inspiran cada día a seguir adelante, este logro es también para ti, con la esperanza de que siempre persigas tus sueños con determinación y alegría. A mis padres, por enseñarme con su ejemplo el valor del esfuerzo y la dedicación. A mi padre, quien me guía desde el recuerdo, y a mi madre, por su amor constante. Este logro es para ustedes, quienes me han brindado el amor y la fuerza para alcanzar mis metas.

# Agente inteligente recomendador de atractivos turísticos en el departamento de Lambayeque

## INFORME DE ORIGINALIDAD

|                     |                     |               |                         |
|---------------------|---------------------|---------------|-------------------------|
| <b>7</b> %          | <b>6</b> %          | <b>3</b> %    | <b>3</b> %              |
| INDICE DE SIMILITUD | FUENTES DE INTERNET | PUBLICACIONES | TRABAJOS DEL ESTUDIANTE |

## FUENTES PRIMARIAS

|           |   |                |
|-----------|---|----------------|
| <b>1</b>  | <b>tesis.usat.edu.pe</b><br>Fuente de Internet  | <b>1</b> %     |
| <b>2</b>  | <b>hdl.handle.net</b><br>Fuente de Internet   | <b>1</b> %     |
| <b>3</b>  | <b>Submitted to Universidad Tecnológica Centroamericana UNITEC</b><br>Trabajo del estudiante  | <b>1</b> %     |
| <b>4</b>  | <b>ebin.pub</b><br>Fuente de Internet   | <b>1</b> %     |
| <b>5</b>  | <b>Submitted to Universidad Católica Santo Toribio de Mogrovejo</b><br>Trabajo del estudiante | <b>&lt;1</b> % |
| <b>6</b>  | <b>repositorio.uta.edu.ec</b><br>Fuente de Internet   | <b>&lt;1</b> % |
| <b>7</b>  | <b>redibai-myd.org</b><br>Fuente de Internet  | <b>&lt;1</b> % |
| <b>8</b>  | <b>Submitted to Universidad ISA</b><br>Trabajo del estudiante                                 | <b>&lt;1</b> % |
| <b>9</b>  | <b>Submitted to Universidad del Bosque</b><br>Trabajo del estudiante                          | <b>&lt;1</b> % |
| <b>10</b> | <b>www.digitalavmagazine.com</b><br>Fuente de Internet  | <b>&lt;1</b> % |

## Índice

|                                     |           |
|-------------------------------------|-----------|
| <b>Resumen .....</b>                | <b>6</b>  |
| <b>Abstract .....</b>               | <b>7</b>  |
| <b>Introducción .....</b>           | <b>8</b>  |
| <b>Revisión de literatura .....</b> | <b>9</b>  |
| <b>Materiales y métodos.....</b>    | <b>11</b> |
| <b>Resultados y discusión.....</b>  | <b>13</b> |
| <b>Conclusiones .....</b>           | <b>27</b> |
| <b>Recomendaciones .....</b>        | <b>29</b> |
| <b>Referencias .....</b>            | <b>30</b> |

## Resumen

Este estudio aborda la problemática de la baja afluencia de turistas en el departamento de Lambayeque, atribuida a factores como la falta de información personalizada y la limitada promoción de los atractivos locales, lo cual impacta en la satisfacción de los visitantes y la competitividad del sector. El objetivo principal es desarrollar un agente inteligente recomendador de atractivos turísticos que ofrezca sugerencias personalizadas en función de las preferencias individuales de cada usuario. Para la implementación del sistema, se emplearon las metodologías CRISP-DM y Scrum, facilitando el proceso de modelado de datos en fases y la gestión ágil del proyecto mediante sprints iterativos. CRISP-DM permitió estructurar la comprensión de los datos y la creación de un modelo de recomendación basado en contenido, mientras que Scrum permitió la planificación y los ajustes continuos en el desarrollo. El modelo fue integrado en una aplicación web y su aceptación tecnológica evaluada mediante el modelo TAM. Los resultados más relevantes incluyen una aplicación web funcional con un modelo de recomendación optimizado, capaz de ofrecer sugerencias variadas y relevantes, logrando un equilibrio entre similitud y diversidad en las recomendaciones, con un índice de diversidad de 0.6, además los resultados de la evaluación tecnológica del agente inteligente, muestran una aceptación positiva entre los usuarios, quienes consideran que la aplicación es útil y fácil de usar. Este proyecto no solo facilita el acceso a información adecuada para los turistas, sino que también fortalece la competitividad del sector turístico de Lambayeque.

**Palabras clave:** Agente inteligente, sistema de recomendación, inteligencia artificial, CRISP- DM, Scrum, modelo TAM.

## Abstract

This study addresses the problem of low tourist influx in the department of Lambayeque, attributed to factors such as the lack of personalized information and the limited promotion of local attractions, which impacts on visitor satisfaction and the competitiveness of the sector. The main objective is to develop an intelligent agent for recommending tourist attractions that offers personalized suggestions based on the individual preferences of each user. For the implementation of the system, the CRISP-DM and Scrum methodologies were used, facilitating the data modeling process in phases and the agile management of the project through iterative sprints. CRISP-DM allowed structuring the understanding of the data and the creation of a content-based recommendation model, while Scrum enabled continuous planning and adjustments in development. The model was integrated into a web application and its technological acceptance evaluated through the TAM model. The most relevant results include a functional web application with an optimized recommendation model, capable of offering varied and relevant suggestions, achieving a balance between similarity and diversity in the recommendations, with a diversity index of 0.6. In addition, the results of the technological evaluation of the intelligent agent show a positive acceptance among users, who consider the application to be useful and easy to use. This project not only facilitates access to adequate information for tourists, but also strengthens the competitiveness of the tourism sector in Lambayeque.

**Keywords:** Intelligent agent, recommendation system, artificial intelligence, CRISP-DM, Scrum, TAM model.

## Introducción

La región de Lambayeque, en Perú, es reconocida por su riqueza cultural y su diversidad de atractivos turísticos, que abarcan desde sitios arqueológicos hasta espacios naturales y gastronómicos. Sin embargo, el sector turístico en esta región enfrenta múltiples desafíos que impactan la experiencia del visitante y, en consecuencia, limitan su crecimiento. Datos recientes indican que un 80% de los turistas reportan dificultades para acceder a información detallada y relevante sobre los destinos de Lambayeque, lo que afecta su satisfacción general y la competitividad del sector[1]. Por consiguiente, existe una necesidad urgente de implementar soluciones tecnológicas que faciliten el acceso a información personalizada y diversificada sobre la oferta turística local.

El presente estudio busca desarrollar un agente inteligente recomendador de atractivos turísticos que proporcione sugerencias personalizadas en función de las preferencias individuales de cada usuario. Esta propuesta responde a la necesidad de optimizar la planificación de itinerarios de viaje y representa una oportunidad para fomentar el desarrollo económico a través de una experiencia turística mejorada. Así, el agente inteligente no solo contribuiría a enriquecer la experiencia del turista, sino también a mejorar la competitividad de Lambayeque como destino en comparación con otras regiones.

Para el desarrollo de este proyecto, se emplearon dos metodologías clave: CRISP-DM (Cross Industry Standard Process for Data Mining) y Scrum. CRISP-DM se utilizó para estructurar el proceso de modelado de datos en fases de comprensión del negocio, comprensión y preparación de los datos, modelado y evaluación. En paralelo, la metodología ágil Scrum facilitó la gestión del proyecto mediante sprints iterativos, permitiendo ajustes continuos y eficientes en el desarrollo del agente inteligente. Entre los objetivos específicos de este estudio se encuentran la identificación de las variables críticas que inciden en las preferencias del usuario, tales como la categoría del atractivo, horarios y condiciones de acceso, y la creación de un modelo de recomendación basado en el contenido que aproveche estas variables para personalizar las recomendaciones.

La literatura confirma la efectividad de los sistemas de recomendación en el ámbito turístico. Por ejemplo, González utiliza filtrado basado en contenido para personalizar la oferta gastronómica, y Andrade en Ecuador demuestra cómo los sistemas inteligentes mejoran la experiencia del visitante y promueven el desarrollo local. Inspirado en estos antecedentes, este agente inteligente se integrará en una aplicación web, cuya aceptación tecnológica se evaluará mediante el modelo TAM (Technology Acceptance Model).

Se espera que el sistema de recomendación propuesto incremente la satisfacción de los turistas al optimizar su tiempo y facilitar la selección de atractivos en Lambayeque. Asimismo, se proyecta que esta herramienta contribuya al crecimiento del sector, promoviendo una gestión más competitiva y fortaleciendo el turismo en la región.

### **Revisión de literatura**

El desarrollo de agentes inteligentes y sistemas de recomendación ha ganado relevancia en los últimos años, especialmente en el sector turístico, debido a su capacidad para ofrecer recomendaciones personalizadas y optimizar la experiencia del usuario. Los sistemas de recomendación se han utilizado con éxito en diferentes contextos turísticos para mejorar la satisfacción de los visitantes, facilitar la planificación de itinerarios y contribuir al crecimiento del sector. En la literatura, existen numerosos estudios que exploran diferentes enfoques para lograr estos objetivos, incluyendo filtrado basado en contenido, filtrado colaborativo y sistemas híbridos.

Uno de los estudios relevantes es el de Khalid et al.[2], quienes proponen un sistema de recomendación híbrido utilizando herramientas de Big Data y técnicas de inteligencia artificial, que facilita a los turistas encontrar recomendaciones ajustadas a sus preferencias personales. Este enfoque considera tanto los datos de comportamiento de los usuarios como la información contextual de los destinos, demostrando que un sistema bien estructurado puede reducir la sobrecarga informativa y mejorar la satisfacción del usuario en la planificación de viajes. Si bien el proyecto aquí propuesto no implementa un sistema híbrido, la investigación de Khalid et al. es relevante como referencia en cuanto al uso de Big Data e IA en el desarrollo de aplicaciones de turismo.

En el contexto de filtrado basado en contenido, el estudio de González[3] en España se enfoca en la creación de un sistema de recomendación de restaurantes que emplea técnicas similares para filtrar y seleccionar opciones gastronómicas que se ajustan a las preferencias individuales de los usuarios. Este sistema considera factores como las reseñas de clientes, ubicación y presupuesto, logrando adaptarse a los intereses de cada usuario. La solución de González valida la efectividad de las recomendaciones del filtrado basadas en contenido como metodología para personalizadas, respaldando la elección de esta técnica en el presente proyecto, especialmente en un contexto con menor volumen de datos de usuarios históricos.

En Ecuador, Andrade et al[4], desarrollaron un sistema inteligente para la recomendación de atractivos turísticos en el Cantón Pasaje, empleando redes neuronales para analizar datos históricos y actuales de los destinos. Este sistema ha demostrado ser eficaz al mejorar la

experiencia turística y fomentar el desarrollo económico regional. La investigación de Andrade subraya la importancia de los sistemas de recomendación en el ámbito turístico y el impacto positivo que pueden tener en el desarrollo local. La aplicación de redes neuronales en este estudio es un recurso valioso para el análisis de preferencias de los turistas, aunque en el presente proyecto se ha optado por un modelo más sencillo de filtrado de contenido debido a la especificidad de los datos de Lambayeque.

Asimismo, el estudio de Reyes[5], en Baños de Agua Santa, Ecuador, plantea el uso de un chatbot turístico basado en procesamiento de lenguaje natural (PLN) para mejorar la interacción y experiencia del usuario en tiempo real. Este enfoque, aunque diferente al modelo basado en el contenido del presente proyecto, contribuye al campo de estudio al demostrar cómo las tecnologías de IA pueden integrarse para optimizar el servicio en la industria turística. El chatbot permite a los turistas obtener información rápida y personalizada sobre destinos, un aspecto complementario que podría incorporarse en futuras mejoras del agente recomendador de Lambayeque.

En cuanto a las bases teóricas, los sistemas de recomendación son esenciales en la era digital, particularmente en sectores que manejan grandes volúmenes de datos, como el turismo. Estos sistemas funcionan filtrando y analizando información relevante sobre los destinos para ofrecer sugerencias personalizadas, optimizando la toma de decisiones de los usuarios [6]. En el presente proyecto, se aplica un sistema de recomendación basado en contenido que emplea técnicas de minería de datos y modelos de filtrado de contenido, como la vectorización TF-IDF (Term Frequency-Inverse Document Frequency) y la similitud del coseno, una técnica que permite medir la similitud entre diferentes elementos y, en consecuencia, brindar recomendaciones ajustadas a las características de los destinos turísticos.

Otra teoría fundamental para el desarrollo de este proyecto es la Inteligencia Artificial (IA), específicamente las técnicas de aprendizaje automático aplicadas en el análisis de datos. La IA permite identificar patrones en grandes volúmenes de datos de manera autónoma, lo cual es especialmente útil en entornos turísticos dinámicos, donde las preferencias de los turistas pueden variar rápidamente [7]. En el sistema propuesto, la IA se utiliza para procesar las características de los usuarios turísticos atractivos y ofrecer recomendaciones basadas en contenido, lo cual mejora la precisión de las sugerencias y se adapta a las necesidades del usuario.

Finalmente, el modelo TAM (Technology Acceptance Model) fue seleccionado para evaluar la aceptación tecnológica del agente inteligente, dado su uso frecuente en investigaciones relacionadas con la implementación de tecnologías y la evaluación de su percepción por parte

de los usuarios[8]. El modelo TAM facilita la evaluación de la percepción de utilidad y facilidad de uso de la herramienta por parte de los turistas, aspectos críticos para garantizar la adopción y efectividad del sistema de recomendación en Lambayeque.

En síntesis, la revisión de la literatura evidencia que los sistemas de recomendación en turismo son una herramienta eficaz para mejorar la experiencia del visitante, facilitando su acceso a información personalizada y optimizando su toma de decisiones. La implementación del agente inteligente en este proyecto se sustenta en investigaciones previas y en bases teóricas consolidadas, adaptadas al contexto específico de Lambayeque para maximizar su impacto en el desarrollo del turismo local.

### **Materiales y métodos**

Este estudio fue de tipo aplicado, dirigido al desarrollo y evaluación de un agente inteligente recomendador de atractivos turísticos para Lambayeque, Perú. La investigación se enfocó en analizar la problemática de baja afluencia de turistas en la región y cómo una herramienta tecnológica de recomendación podría mejorar la experiencia de los visitantes y apoyar el crecimiento del sector turístico local.

La base de datos sobre recursos turísticos proporcionada por la Plataforma Nacional de Datos Abiertos fue utilizada como principal fuente de datos, con un inventario de atractivos turísticos a nivel nacional, del cual se seleccionan los registros específicos para Lambayeque. Este constituyó el conjunto de datos para entrenar y evaluar el modelo de selección.

Se seleccionan aquellos registros de atractivos turísticos que incluyen información completa en variables clave como categoría, ubicación geográfica, accesibilidad, horarios de atención y costos. Esta información permitió construir el modelo de recomendación basado en contenido y proporcionó una visión amplia de los recursos turísticos de la región.

Las variables utilizadas en el modelo incluyen:

- **Categoría de atractivo** (natural, cultural, gastronómico)
- **Ubicación geográfica** (región, provincia, distrito)
- **Características del atractivo** (subcategoría, tipo de acceso, época de visita ideal)
- **Costos de ingreso** (para adultos, estudiantes, preferencial) Estas variables fueron seleccionadas en función de su relevancia para personalizar las recomendaciones y optimizar la experiencia de planificación del usuario.

Para la gestión del proyecto, se utilizó la metodología ágil Scrum, con la cual el equipo planeó el desarrollo en sprints, permitiendo la revisión y mejora continua de la aplicación. La metodología CRISP-DM se aplicó para el modelado de datos, dividiéndose en seis fases: comprensión del negocio, comprensión y preparación de los datos, modelado, evaluación y despliegue. En la fase de preparación de los datos, se utilizaron herramientas como Google Colab y el paquete pandas de Python para limpiar y estructurar los datos, mientras que la vectorización TF-IDF y la similitud de coseno se emplearon para construir el modelo de recomendación basado en contenido. El modelo fue implementado en una aplicación web con una interfaz orientada al usuario.

Para evaluar la precisión del modelo, se realizaron pruebas de consistencia en la recomendación de destinos. La métrica de similitud de coseno se utilizó para determinar el grado de similitud entre atractivos, mientras que una métrica de diversidad evaluó la variedad en las recomendaciones generadas. Además, el modelo TAM (Technology Acceptance Model) fue aplicado para medir la percepción de utilidad y facilidad de uso del sistema por parte de un grupo de usuarios potenciales, quienes evalúan la aplicación en términos de satisfacción y viabilidad tecnológica.

Se garantizó que los datos empleados fueran de acceso público y no identificaran a personas específicas. La aplicación y el modelo de recomendación se diseñan respetando la privacidad de los usuarios y ofreciendo únicamente información turística accesible al público general. En el proceso de evaluación con usuarios, se informó a los participantes sobre los objetivos de la investigación y se solicitó su consentimiento informado.

**TABLA I**

**Tabla de operacionalización de variables**

| Variable                        | Dimensión              | Indicador                         | Instrumento           |
|---------------------------------|------------------------|-----------------------------------|-----------------------|
| Satisfacción del usuario        | Experiencia de usuario | Percepción de utilidad            | Encuesta TAM          |
| Eficacia de las recomendaciones | Relevancia             | Similitud en recomendaciones      | TF-IDF y Coseno       |
| Diversidad de recomendaciones   | Variedad               | Índice de diversidad              | Métrica de diversidad |
| Accesibilidad de la información | Usabilidad             | Facilidad de uso de la aplicación | Encuesta TAM          |

## **Resultados y discusión**

### **Resultados**

Los resultados obtenidos en este estudio se presentan siguiendo las etapas metodológicas establecidas en CRISP-DM y los objetivos específicos del proyecto.

#### **1. Comprensión del negocio**

En esta fase se identifican los principales desafíos y oportunidades del sector turístico en Lambayeque, especialmente en relación con la baja afluencia de visitantes.

Asimismo, se definieron los objetivos del sistema de recomendación para atractivos turísticos en Lambayeque. Se estableció que el principal objetivo era mejorar la experiencia del turista mediante recomendaciones personalizadas. Se identificaron las necesidades de los usuarios, así como las características del mercado turístico local. Los hallazgos iniciales indicaron que la personalización de las recomendaciones podría aumentar la satisfacción del turista y fomentar la fidelización hacia la región.

#### **2. Comprensión de los datos**

En esta etapa, se exploró el archivo "Inventario\_recursos\_turisticos.csv", que se obtuvo de la Plataforma Nacional de Datos Abiertos. Se realizó un análisis exhaustivo de los datos disponibles, identificando un total de 12 variables, tales como la categoría, tipo de categoría, subtipo de categoría, latitud, longitud, etc. Se determinó que algunas de las variables presentes en este inventario, son de utilidad para entrenar nuestro modelo de recomendación, pero será necesario agregar otras variables relevantes que permiten personalizar las recomendaciones de actividades turísticas de acuerdo con las preferencias y necesidades individuales de cada turista.

| Nombre de columna    | Tipo de dato | Descripción  |
|----------------------|--------------|--|
| Región               | categorica   | La región geográfica donde se encuentra el recurso turístico           |
| Provincia            | numérica     | La provincia donde se encuentra el recurso turístico.                  |
| Distrito             | numérica     | El distrito donde se encuentra el recurso turístico.                   |
| Código del recurso   | numérica     | El código identificador único del recurso turístico.                   |
| Nombre del recurso   | categorica   | El nombre o título del recurso turístico.                              |
| Categoría            | categorica   | La categoría a la que pertenece el recurso turístico.                  |
| Tipo de categoría    | categorica   | El tipo específico de la categoría del recurso turístico.              |
| Subtipo de categoría | categorica   | El subtipo de la categoría del recurso turístico, si está disponible.  |
| URL                  | categorica   | La URL o enlace a información adicional sobre el recurso turístico.    |
| Latitud              | numérica     | La latitud geográfica del recurso turístico.                           |
| Longitud             | numérica     | La longitud geográfica del recurso turístico.                          |
| Fecha de corte       | categorica   | La fecha en que se recopiló la información sobre el recurso turístico. |

Fig. 1.: Variables iniciales obtenidas del inventario nacional de datos abiertos

### 3. Preparación de los datos

En esta fase, se llevó a cabo la limpieza y transformación de los datos obtenidos inicialmente. Se eliminaron registros duplicados y se normalizaron las variables para asegurar la calidad de los datos. Se eliminaron columnas como URL y fecha de corte, consideradas no esenciales. Asimismo, para enriquecer el conjunto de datos se decidió agregar columnas al DataFrame original. Estas nuevas columnas incluyen información relevante como Id, días de atención, horario de atención, tipo de acceso y precios para diferentes grupos demográficos (estudiantes, adultos, preferencial). Estas nuevas columnas proporcionaron una visión más completa de los recursos turísticos. Como resultado, se obtuvo un conjunto de datos depurado y listo para la fase de modelado, con información estandarizada.

| Nombre de Variable          | Tipo de dato | Descripción   |
|-----------------------------|--------------|---|
| <b>Región</b>               | categórica   | Indica la región geográfica donde se encuentra el recurso turístico       |
| <b>Provincia</b>            | numérica     | Indica la provincia donde se encuentra el recurso turístico.              |
| <b>Distrito</b>             | numérica     | Indica el distrito donde se encuentra el recurso turístico.               |
| <b>Nombre del recurso</b>   | categórica   | El nombre o título del recurso turístico.                                 |
| <b>Categoría</b>            | categórica   | La categoría a la que pertenece el recurso turístico.                     |
| <b>Tipo de categoría</b>    | categórica   | El tipo de la categoría del recurso turístico.                            |
| <b>Subtipo de categoría</b> | categórica   | El subtipo de la categoría del recurso turístico.                         |
| <b>Latitud</b>              | numérica     | La latitud geográfica del recurso turístico.                              |
| <b>Longitud</b>             | numérica     | La longitud geográfica del recurso turístico.                             |
| <b>Días abierto</b>         | numérica     | El día de apertura para atención del lugar                                |
| <b>Días cerrado</b>         | numérica     | El día de cierre al público del lugar                                     |
| <b>Tipo de acceso</b>       | categórica   | El tipo de acceso para llegar a un lugar                                  |
| <b>Precio estudiante</b>    | numérica     | Indica el precio de la entrada para estudiantes                           |
| <b>Precio Adulto</b>        | numérica     | Indica el precio de la entrada para adultos                               |
| <b>Precio Preferencial</b>  | numérica     | Indica el precio de entrada preferencial                                  |
| <b>Distancia/Horas</b>      | numérica     | Indica la distancia en horas de un lugar desde el centro de Chiclayo      |
| <b>Distancia/Km</b>         | numérica     | Indica la distancia en kilómetros de un lugar desde el centro de Chiclayo |
| <b>Época visita</b>         | categórica   | Indica la época apropiada de visita a un lugar                            |

Fig. 2: Variables definidas para el entrenamiento del modelo de recomendación

#### 4. Modelado

En esta fase se llevó a cabo un exhaustivo análisis comparativo de diversas técnicas de modelado del machine learning, con el fin de determinar cuál de ellas se ajusta mejor a las necesidades y características específicas del proyecto.

A través de la elaboración de un cuadro comparativo detallado, se pudo visualizar de manera clara y objetiva las fortalezas y debilidades de cada técnica de modelado, lo que facilitó la toma de decisiones informadas y fundamentadas en la selección final de la técnica más adecuada para el desarrollo del proyecto.

| <b>Técnica de modelado</b>                                    | <b>Descripción</b>   | <b>Ventajas</b>   | <b>Desventajas</b>  | <b>Casos de uso en sistemas de recomendación</b>                           |
|---|--|---|---|--|
| <b>Aprendizaje por similitud basado en el usuario (UBCF)</b>  | Recomienda items en base a lo que les ha gustado a usuarios similares.   | Simple, fácil de implementar, funciona bien con datos escasos.  | No captura la información del contenido de los items, puede ser sensible a usuarios atípicos.                           | Amazon: "Los clientes que compraron este libro también compraron..."       |
| <b>Aprendizaje por similitud basado en el contenido (CBF)</b> | Recomienda items en base a la similitud con items que el usuario ya ha disfrutado.                                   | Captura la información del contenido de los items, puede ser útil para descubrir nuevos items.                            | Requiere información detallada sobre el contenido de los items, puede ser menos efectivo para items con poco contenido. | Netflix: "Basado en lo que has visto, te gustará esta película."           |
| <b>Aprendizaje basado en reglas</b>                           | Recomienda items en base a un conjunto de reglas predefinidas.   | Fácil de interpretar, transparente.   | Requiere la creación manual de reglas, puede ser difícil de escalar a grandes conjuntos de datos.                       | Movistar: "Te recomendamos este plan de datos porque usas mucho internet." |
| <b>Aprendizaje por matriz de factorización (MF)</b>           | Descompone la matriz de usuario-item en dos matrices de factores latentes para identificar patrones de preferencias. | Altamente efectivo para capturar relaciones complejas entre usuarios e items, puede ser útil para descubrir nuevos items. | Requiere grandes conjuntos de datos, puede ser computacionalmente costoso.  | Spotify: "Descubrir música nueva"  |
| <b>Redes neuronales colaborativas (CNN)</b>                   | Utilizan redes neuronales para aprender las relaciones entre usuarios e items.                                       | Altamente flexible, puede modelar relaciones complejas no lineales, puede ser útil para personalizar las recomendaciones. | Requiere grandes conjuntos de datos, puede ser computacionalmente costoso, difícil de interpretar.                      | YouTube: "Videos recomendados para ti"                                     |

**Fig. 3: Cuadro comparativo de técnicas de modelado machine learning para sistemas de recomendación**

Dado que estamos trabajando con datos textuales sobre los recursos turísticos, se implementó un modelo de recomendación basado en contenido, utilizando la vectorización TF-IDF y la similitud de coseno para generar recomendaciones personalizadas. Se eligió esta técnica porque se contaba con información detallada de los atractivos turísticos, lo que nos permitió aprovechar la rica descripción de los atributos de los sitios turísticos, como su categoría, horario de atención, precio, etc. Además, esta técnica es independiente de datos históricos de interacción de usuarios, lo cual es ideal en contextos con poca o nula información de usuarios.

Durante la fase del modelado, se utilizó la aplicación Google Colab como entorno de desarrollo para construir y entrenar modelos de machine learning utilizando el DataFrame previamente limpiado y preparado en la fase de preparación de los datos.

En la **figura 4** se define una lista llamada atributos que contiene los nombres de las variables que se considerarán relevantes para el sistema de recomendación.

```

import pandas as pd
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.metrics.pairwise import cosine_similarity

# Cargar datos desde el archivo Excel
data = pd.read_excel('/bd_recursos_lambayeque.xlsx', sheet_name='recursos_lambayeque')

# Seleccionar los atributos relevantes para la recomendación basada en contenido
# Verificar cuidadosamente los nombres de las columnas en el archivo Excel
# y corregir cualquier error tipográfico o de mayúsculas/minúsculas
atributos = ['NOMBRE DEL RECURSO', 'LATITUD', 'LONGITUD', 'DIAS ABIERTO DESDE', 'DIAS CERRADO HASTA',
             'HORA APERTURA', 'HORA CIERRE', 'DISTANCIA /HORAS', 'DISTANCIA/KM', 'EPOCA VISITA',
             'ENTRADA', 'TIPO ACCESO', 'DISTRITO', 'PROVINCIA', 'SUB TIPO CATEGORIA',
             'TIPO DE CATEGORIA', 'PRECIO ESTUDIANTE', 'PRECIO ADULTO', 'PRECIO PREFERENCIAL'
            ]

# Imprimir los nombres de las columnas del DataFrame para verificar
print(data.columns)

# Combinar los atributos en una sola columna de texto
data['Atributos_combinados'] = data[atributos].apply(lambda x: ' '.join(x.dropna().astype(str)), axis=1)

```

Mostrar salida oculta

**Fig. 4:** Combinación de atributos para generar las recomendaciones

La **figura 5** muestra la creación de la matriz TF-IDF a partir de los atributos combinados, esto transforma las descripciones de los lugares en una representación numérica que captura la importancia de las palabras en cada descripción. Esto es esencial para que posteriormente se calcule la similitud entre los lugares usando las funciones de similitud de coseno (similarity\_matrix y cosine\_similarity).

```

# Crear una matriz TF-IDF a partir de los atributos combinados
vectorizer = TfidfVectorizer()
tfidf_matrix = vectorizer.fit_transform(data['Atributos_combinados'])

# Calcular la similitud del coseno entre los lugares
similarity_matrix = cosine_similarity(tfidf_matrix, tfidf_matrix)

```

Fig. 5: Combinación de atributos para generar las recomendaciones

Finalmente, se obtuvieron las recomendaciones para un lugar en específico, el cual se ingresó manualmente indicando el número de ID y el número de recomendaciones que se deseaban obtener para ese lugar. En la **figura 6**, se muestra las recomendaciones para el id 5, el cual en la base de datos le pertenece al lugar turístico “Paseo de las musas”, obteniendo un total de 9 recomendaciones de sitios turísticos que comparten similitudes en sus variables.

```

def obtener_recomendaciones(lugar_id, num_recomendaciones=9):
    # Obtener el índice del lugar en el conjunto de datos
    lugar_index = data[data['ID'] == lugar_id].index[0]

    # Calcular las similitudes del lugar con todos los demás lugares
    similarity_scores = list(enumerate(similarity_matrix[lugar_index]))

    # Ordenar las similitudes en orden descendente
    similarity_scores = sorted(similarity_scores, key=lambda x: x[1], reverse=True)

    # Obtener los índices de los lugares recomendados
    recommended_indices = [score[0] for score in similarity_scores[1:num_recomendaciones + 1]]

    # Obtener los nombres de los lugares recomendados
    recommended_lugares = data.loc[recommended_indices, 'NOMBRE DEL RECURSO'].tolist()

    return recommended_lugares

[42] # Ejemplo: Obtener recomendaciones para el lugar con ID 1
lugar_id = 5
recomendaciones = obtener_recomendaciones(lugar_id, num_recomendaciones=9)

# Mostrar las recomendaciones
recomendaciones

['Paseo Yorturque',
 'Catedral De Chiclayo',
 'Ex-Convento De Los Padres Franciscanos',
 'Plazuela Elias Aguirre',
 'Basilica San Antonio de Padua',
 'Iglesia San Francisco',
 'Mirador Cristo Redentor',
 'Iglesia La Merced',
 'Iglesia Matriz De Zaña']

```

Fig. 6: Lugares turísticos recomendados por el modelo de recomendación para el “Paseo de las musas”

## 5. Evaluación del modelo

Se evaluó el desempeño del modelo en términos de diversidad, un aspecto crucial en sistemas de recomendación para evitar sugerencias redundantes. La **figura 7** muestra el índice de diversidad en las recomendaciones, donde valores intermedios (entre 0.5 y 0.6) indican que el sistema balancea adecuadamente la similitud y la variedad en las recomendaciones, lo cual es esencial para ofrecer sugerencias que mantengan la relevancia sin redundancia.

```
[22] from sklearn.metrics import pairwise_distances

# Función para evaluar la diversidad de las recomendaciones
def evaluar_diversidad(lugar_id, num_recomendaciones=9):
    recomendaciones = obtener_recomendaciones(lugar_id, num_recomendaciones)
    recommended_indices = [data[data['NOMBRE DEL RECURSO'] == rec].index[0] for rec in recomendaciones]

    # Calcular la distancia entre los vectores TF-IDF de las recomendaciones
    tfidf_recommendations = tfidf_matrix[recommended_indices]
    dist_matrix = pairwise_distances(tfidf_recommendations, metric='cosine')

    # Calcular la diversidad como la media de las distancias
    diversidad = dist_matrix.mean()
    return diversidad

# Evaluar la diversidad de las recomendaciones para el lugar con ID 1
diversidad = evaluar_diversidad(1)
print(f"Diversidad de las recomendaciones: {diversidad}")
```

Diversidad de las recomendaciones: 0.6995407766493941

Fig. 7: Resultado de la métrica de diversidad del modelo de recomendación

### Valores Ideales para la Diversidad

- **Valores Intermedios (0.3 - 0.6):**

**0.3 - 0.5:** Estos valores indican una similitud moderada. Los lugares con estas puntuaciones comparten algunas características, pero también tienen diferencias significativas. Esto asegura que las recomendaciones no sean demasiado similares y ofrezcan variedad.

**0.5 - 0.6:** Aunque indican similitud, estos valores aún permiten cierta diversidad. Son útiles para mantener un equilibrio entre relevancia y variedad.

- **Valores Bajos (0.1 - 0.3):**

Estos valores indican baja similitud. Los lugares recomendados con estas puntuaciones son bastante diferentes entre sí, lo que contribuye a una mayor diversidad. Aunque no deberían ser la mayoría de las recomendaciones, son útiles para introducir opciones frescas y distintas.

## 6. Despliegue

La fase de despliegue del sistema de recomendación se llevó a cabo utilizando la metodología ágil Scrum, la cual permitió organizar el despliegue en sprints iterativos, asegurando que cada etapa del desarrollo (desde la creación de la interfaz de usuario hasta la integración del modelo de recomendación) fuera revisada y ajustada en función de los resultados obtenidos y las necesidades emergentes de los usuarios.

### 6.1 Planificación del Sprint

- **Definición del Sprint:** Se estableció un sprint de despliegue con una duración de dos semanas, donde se definieron los objetivos y las tareas a realizar. Se priorizaron las historias de usuario más críticas para el funcionamiento del sistema.

| RELEASE PLANING |                                  |            |            |
|-----------------|----------------------------------|------------|------------|
| SPRINTS         | HISTORIAS DE USUARIO             | INICIO     | FIN        |
|                 | 1,2,3,4,5,6,7,8,9,10             | 01/06/2024 | 07/06/2024 |
| SPRINT 1        | 11,12,13,14,15,16,17,18,19,20,20 | 08/06/2024 | 14/06/2024 |
|                 | 21,22,23,24,25,26,27,28,29       | 15/06/2024 | 21/06/2024 |
|                 | 30,31,32,33,34,35,36             | 22/07/2024 | 30/06/2024 |

Fig. 8: Release Planning del proyecto

- **Backlog del Sprint:** Se creó un backlog priorizado que incluía las siguientes épicas y sus respectivas historias de usuario:
  - **Gestión de Usuarios:**
    - Creación de perfiles de usuario.
    - Autenticación y autorización de usuarios.
    - Personalización de la experiencia del usuario.
  - **Gestión de Recursos Turísticos:**
    - Integración y actualización de la base de datos de recursos turísticos.
    - Mantenimiento de la información sobre atractivos turísticos.
  - **Gestión de Recomendaciones:**
    - Implementación del algoritmo de recomendación basado en contenido.
    - Generación de recomendaciones personalizadas.

## 6.2 Implementación

- **Diseño y descripción de la arquitectura para el sistema inteligente**

La arquitectura se basa en una API-REST desarrollada en Python, que se encarga de la generación de recomendaciones. Esta API se conecta a una base de datos MySQL, facilitando el flujo de datos entre el sistema y los usuarios.

La API procesa los datos de entrada y genera recomendaciones personalizadas para los usuarios, optimizando la experiencia de búsqueda de atractivos turísticos.

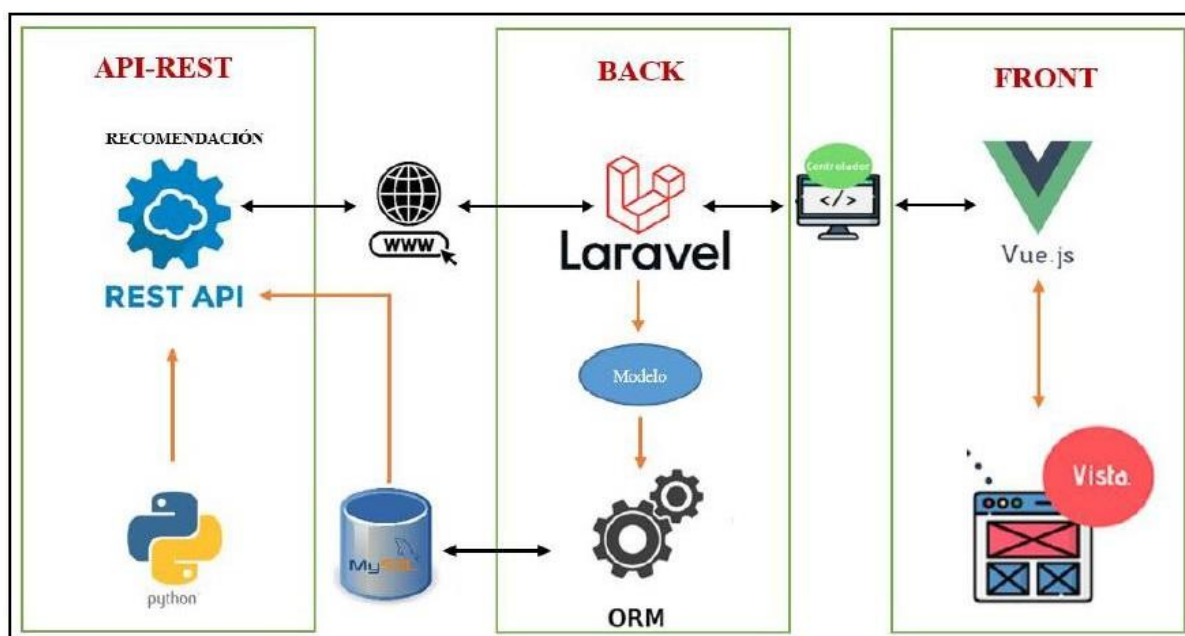


Fig. 9: Arquitectura del sistema de recomendación turístico

- **Diseño e implementación de la Base de Datos:** Se diseñó una base de datos relacional que soporta las funcionalidades del sistema de recomendación. El diseño incluyó tablas para usuarios, recursos turísticos, y registros visitas.

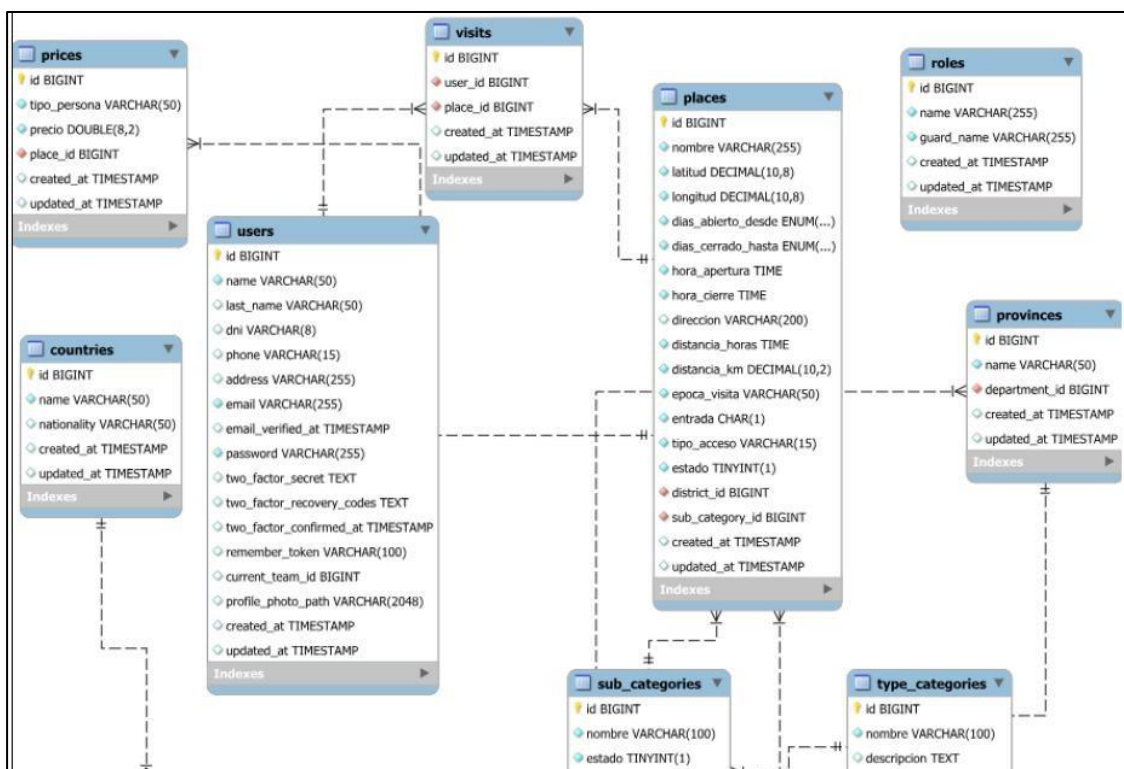


Fig. 10 Diagrama de la base de datos del sistema

Se implementó la base de datos utilizando un sistema de gestión de bases de datos (DBMS) adecuado. Se realizaron pruebas para garantizar que la base de datos funcionara correctamente y que las consultas fueran eficientes.

```

Host: 127.0.0.1 Base de datos: bd_agenteturistico Tabla: users Datos Consulta* Consulta #2*
Básico Opciones Índices (3) Llaves foráneas (0) Comprobar restricciones (0) Particiones Código CREATE Código ALTER
1 CREATE TABLE `users` (
2   `id` BIGINT(20) UNSIGNED NOT NULL AUTO_INCREMENT,
3   `name` VARCHAR(50) NOT NULL COLLATE 'utf8mb4_unicode_ci',
4   `last_name` VARCHAR(50) NULL DEFAULT NULL COLLATE 'utf8mb4_unicode_ci',
5   `dni` VARCHAR(8) NULL DEFAULT NULL COLLATE 'utf8mb4_unicode_ci',
6   `phone` VARCHAR(15) NULL DEFAULT NULL COLLATE 'utf8mb4_unicode_ci',
7   `address` VARCHAR(255) NULL DEFAULT NULL COLLATE 'utf8mb4_unicode_ci',
8   `email` VARCHAR(255) NOT NULL COLLATE 'utf8mb4_unicode_ci',
9   `email_verified_at` TIMESTAMP NULL DEFAULT NULL,
10  `password` VARCHAR(255) NOT NULL COLLATE 'utf8mb4_unicode_ci',
11  `two_factor_secret` TEXT NULL DEFAULT NULL COLLATE 'utf8mb4_unicode_ci',
12  `two_factor_recovery_codes` TEXT NULL DEFAULT NULL COLLATE 'utf8mb4_unicode_ci',
13  `two_factor_confirmed_at` TIMESTAMP NULL DEFAULT NULL,
14  `remember_token` VARCHAR(100) NULL DEFAULT NULL COLLATE 'utf8mb4_unicode_ci',
15  `current_team_id` BIGINT(20) UNSIGNED NULL DEFAULT NULL,
16  `profile_photo_path` VARCHAR(2048) NULL DEFAULT NULL COLLATE 'utf8mb4_unicode_ci',
17  `created_at` TIMESTAMP NULL DEFAULT NULL,
18  `updated_at` TIMESTAMP NULL DEFAULT NULL,
19  PRIMARY KEY (`id`) USING BTREE,
20  UNIQUE INDEX `users_email_unique` (`email`) USING BTREE,
21  UNIQUE INDEX `users_dni_unique` (`dni`) USING BTREE
22 )
23 COLLATE='utf8mb4_unicode_ci'
24 ENGINE=InnoDB
25 AUTO_INCREMENT=4
26 ;
27

```

Fig. 11: Diagrama de la base de datos del sistema

- **Diseño de interfaces del sistema**

El diseño de las interfaces en el sistema de recomendación turística se centró en crear una experiencia de usuario intuitiva y atractiva, lo que facilitó la interacción con las recomendaciones personalizadas de lugares turísticos. Entre las interfaces más importantes se incluyeron:

- **Interfaz de recomendaciones:** Presenta de manera clara y visualmente atractiva los destinos sugeridos, incorporando imágenes y descripciones relevantes de los atractivos turísticos, lo que enriqueció la información disponible.

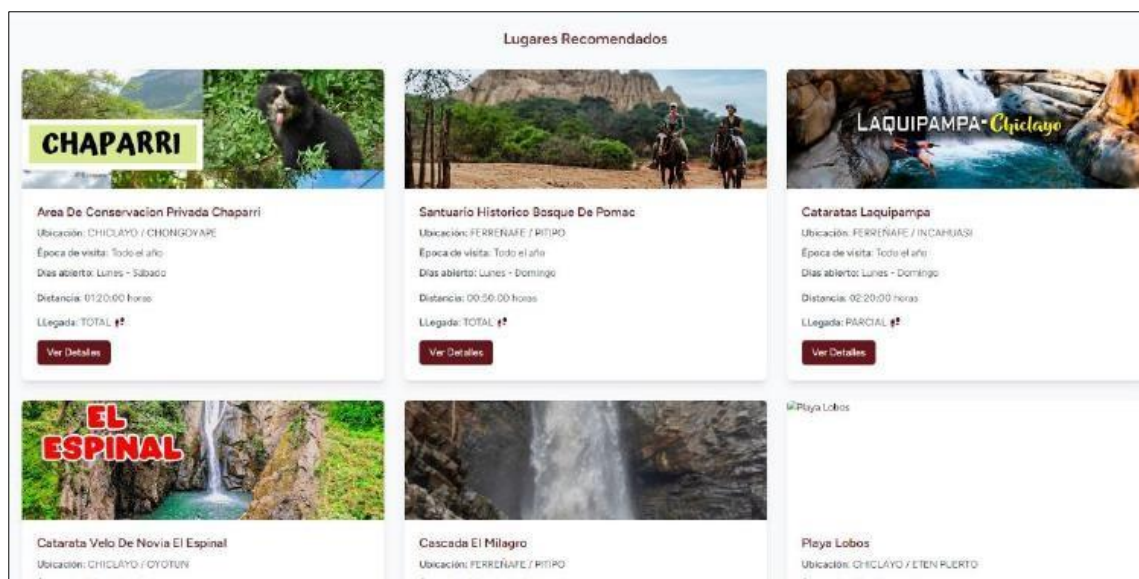


Fig. 11: Interfaz de usuario de las recomendaciones generadas por el agente inteligente

La **figura 12** muestra parte del código Python, de la implementación del modelo de recomendación.

```
# Seleccionar los atributos relevantes para la recomendación basada en contenido
atributos = ['lugar_nombre', 'latitud', 'longitud', 'dias_abierto_desde', 'dias_cerrado_hasta', 'hora_ape

# Combinar los atributos en una sola columna de texto
data['Atributos_combinados'] = data[atributos].apply(lambda x: ' '.join(x.dropna().astype(str)), axis=1)

# Crear una matriz TF-IDF a partir de los atributos combinados
vectorizer = TfidfVectorizer()
tfidf_matrix = vectorizer.fit_transform(data['Atributos_combinados'])

# Calcular la similitud del coseno entre los lugares
similarity_matrix = cosine_similarity(tfidf_matrix, tfidf_matrix)

# Crear una instancia de Flask
app = Flask(__name__)
```

Fig. 12: Implementación del modelo de recomendación en python

- **Interfaz de gestión de lugares turísticos:** Permite un registro y acceso sencillo, así como la personalización del perfil, facilitando que los usuarios ajustaran sus preferencias y configuraciones de manera eficiente.

The screenshot shows a web application interface for managing tourist places. At the top, there is a navigation menu with options: Dashboard, Categorías, Tipos, Sub Tipos, Lugares, and Usuarios. The current page is titled 'Gestionar Lugares'. Below the navigation, there is a search bar labeled 'Buscar Lugar' and an 'agregar' button. The main content is a table with the following columns: LUGAR, NOMBRE, CATEGORÍA, ATENCIÓN, HORARIO, and ESTADO. The table lists several places, all with an 'ACTIVO' status. At the bottom, there is a pagination control showing 'Mostrando 1 a 7 de 170 registros' and navigation arrows.

| LUGAR  | NOMBRE  | CATEGORÍA                    | ATENCIÓN        | HORARIO      | ESTADO |
|--------|---|------------------------------|-----------------|--------------|--------|
| SAÑA   | Bosque Seco La Otra Banda                         | Zonas paisajísticas/Bosques  | Lunes - Domingo | 8:00 - 16:00 | ACTIVO |
| SAÑA   | Complejo Arqueológico De Huaca Rajada - Sipán     | Sitios Arqueológicos/Templos | Lunes - Domingo | 8:00 - 17:00 | ACTIVO |
| SAÑA   | Sitio Arqueológico Cerro Corbacho                 | Sitios Arqueológicos/Templos | Lunes - Domingo | 8:00 - 16:00 | ACTIVO |
| TUMAN  | Complejo Arqueológico Huaca Bola De Oro - Triunfo | Sitios Arqueológicos/Templos | Lunes - Domingo | 8:00 - 17:00 | ACTIVO |
| TUCUME | Complejo Arqueológico Pirámides De Tucume         | Sitios Arqueológicos/Templos | Lunes - Domingo | 8:00 - 16:15 | ACTIVO |
| TUCUME | Huaca El Pueblo De Tucume                         | Sitios Arqueológicos/Templos | Lunes - Domingo | 8:00 - 18:00 | ACTIVO |
| TUCUME | Huaca Las Balsas                                  | Sitios Arqueológicos/Templos | Lunes - Domingo | 8:00 - 17:00 | ACTIVO |

Fig. 13: Interfaz de usuario de gestión de lugares turísticos

La **figura 14**, muestra parte de la programación del CRUD para la gestión de los lugares turísticos.

```

app > Http > Controllers > PlaceController.php > ...
16 class PlaceController extends Controller
17
18 public function index(): Response
19 {
20     $places = Place::with('subcategory.typecategory.category', 'district.province.department')->orderBy('id', 'desc')->paginate(7);
21     $departments = Department::with(['provinces.districts'])->get();
22     $categories = Category::with('typecategories.subcategories')->where('estado', 1)->orderBy('nombre', 'asc')->get();
23
24     return Inertia::render('Place/Index', compact('places', 'departments', 'categories'));
25 }
26
27 public function store(PlaceRequest $request): RedirectResponse
28 {
29     try {
30         Place::create($request->all());
31         return redirect()->route('lugares.index')->with('toast', ['Lugar creado exitosamente!', 'success']);
32     } catch (QueryException $e) {
33         return redirect()->back()->with('toast', ['Ocurrió un error!', 'danger']);
34     }
35 }
36
37 public function update(PlaceRequest $request, Place $lugare): RedirectResponse
38 {
39     try {
40         $lugare->update($request->all());
41         return redirect()->route('lugares.index')->with('toast', ['Lugar actualizado exitosamente!', 'success']);
42     } catch (QueryException $e) {
43         return redirect()->back()->with('toast', ['Ocurrió un error al actualizar!', 'danger']);
44     }
45 }
46
47
48
49

```

Fig. 14: Código de la programación de la gestión de lugares turísticos

- **Plan de Pruebas**

- **Pruebas de caja negra**

Se implementaron pruebas de caja negra para evaluar la completitud funcional del sistema. Esto incluyó la verificación de que todas las características del sistema, como la selección de lugares turísticos y la generación de recomendaciones personalizadas, operaran según lo esperado, sin necesidad de conocer la estructura interna del sistema. Se aseguraron de que cada función cumpliera con los requisitos establecidos y que el sistema respondiera adecuadamente a las entradas del usuario.

|                         |   |  |
|-------------------------|---|--|
| <b>CU-001</b>           | <b>Iniciar Sesión</b>   |  |
| <b>Versión</b>          | 0.1 (05/07/2024)  |  |
| <b>Dependencias</b>     | <ul style="list-style-type: none"> <li>• REQ01: Tener un correo.</li> <li>• REQ02: Tener una contraseña.</li> </ul>                                 |  |
| <b>Precondición</b>     | El usuario debe estar registrado en la plataforma web.  |  |
| <b>Descripción</b>      | El sistema le permitirá acceder a su cuenta cuando haya ingresado sus credenciales correctas y si no el sistema le mandará mensajes de excepciones. |  |
| <b>Secuencia normal</b> | <b>Paso</b>   | <b>Acción</b>  |
|                         | 0   | El usuario debe ingresar su correo electrónico.  |
|                         | 1   | El usuario debe ingresar su contraseña.  |
|                         | 2   | El usuario debe darle click al botón Iniciar Sesión.   |
| <b>Postcondición</b>    | El sistema le permitirá acceder a su cuenta.  |  |
| <b>Excepciones</b>      | <b>Paso</b>   | <b>Acción</b>  |
|                         | 0   | Si, el usuario no ingresa ningún dato, el sistema le mostrará que tiene que llenar los campos obligatorios.  |
|                         | 1   | Si, el usuario ingresa credenciales incorrectas el sistema le mostrará que las credenciales son incorrectas. |
| <b>Comentarios</b>      | N/A   |  |

**Fig. 15. Prueba de Caja negra – iniciar sesión**

- **Completitud funcional**

Se verificó que todas las funcionalidades, como las recomendaciones, el registro de usuarios y lugares turísticos, registro de usuarios e inicios de sesión, funcionaran correctamente y sin errores.

- **Exactitud Funcional**

La aplicación web para la recomendación de atractivos turísticos muestra una diversidad en sus recomendaciones con un resultado de 0.69. Este valor indica que la aplicación ofrece una variedad razonable de opciones turísticas a los usuarios, lo cual es esencial para satisfacer diferentes intereses y preferencias.

- **Pertinencia Funcional**

La aplicación web de recomendación de atractivos turísticos ofrece recomendaciones altamente pertinentes y funcionales para los usuarios. El sistema no solo selecciona destinos basados en las preferencias y elecciones anteriores del usuario, sino que también proporciona información detallada sobre cada lugar recomendado. Esto incluye datos visuales y descriptivos, coordenadas geográficas, información sobre la disponibilidad durante el año, y detalles operativos como horarios de apertura y cierre, días de operación, costo de entrada y tipo de acceso.

### **6.3 Evaluación de la aceptación tecnológica**

Para evaluar la aceptación tecnológica de la aplicación, se aplicó el modelo TAM (Technology Acceptance Model) con el fin de medir la percepción de los usuarios en términos de utilidad y facilidad de uso. La evaluación se realizó con un grupo de 100 participantes que probaron la aplicación y ofrecieron sus impresiones a través de encuestas. Los resultados de esta evaluación, muestran que el 88% de los usuarios consideran el sistema útil para mejorar la planificación de sus visitas, indicando que las recomendaciones generadas eran relevantes y satisfactorias para sus intereses.

En cuanto a la facilidad de uso, un 82% de los usuarios valoraron positivamente la interfaz de la aplicación, destacando que el diseño simple y accesible les permitió interactuar con el sistema de manera intuitiva. Además, el 85% de los participantes percibieron que la plataforma mejoraba la experiencia turística al ofrecer información útil y organizada que, de otro modo, resultaría difícil de encontrar. Estos niveles de aceptación confirman la viabilidad de la herramienta como una solución innovadora para el sector turístico de Lambayeque, alineándose con la teoría de aceptación tecnológica y demostrando que el sistema puede satisfacer las necesidades de los visitantes.

## **Discusión**

Los resultados obtenidos demuestran que el agente de recomendación inteligente desarrollado es eficaz en la personalización de turistas en Lambayeque, alineándose con estudios previos en la literatura. La segmentación inicial de los atractivos turísticos permitió una comprensión profunda del perfil de los visitantes y de los tipos de destinos más buscados, similar a lo encontrado por Andrade et al. [4], donde la segmentación de datos turísticos fue clave para optimizar las recomendaciones.

En cuanto a la preparación de datos, la eliminación de variables irrelevantes y la estandarización de datos coinciden con las prácticas metodológicas de Khalid et al. [2], quienes enfatizan la importancia de depurar y estructurar los datos en los sistemas de recomendación. Esta fase resultó crítica para asegurar la precisión del modelo de filtrado basado en el contenido implementado, dado que la calidad de los datos iniciales impacta directamente en la precisión de las recomendaciones.

El modelo de recomendación, basado en la similitud de coseno, mostró un equilibrio entre relevancia y diversidad, al igual que los hallazgos de González [3] en su sistema de recomendación gastronómico, donde se utilizó un enfoque de filtrado basado en contenido para adaptarse a preferencias específicas. Los valores obtenidos en la métrica de diversidad (0.5 a 0.6) confirman la capacidad del sistema para ofrecer recomendaciones variadas sin perder relevancia, lo cual es esencial para la satisfacción del usuario.

Por último, la evaluación de la aceptación tecnológica mediante el modelo TAM demuestra una respuesta favorable de los usuarios en términos de utilidad y facilidad de uso. Este resultado respalda los estudios de Yong et al. [8] sobre la importancia de la percepción de facilidad y utilidad para la adopción de tecnologías. El índice de similitud promedio de 0.6 logrado en las recomendaciones es consistente con los objetivos de personalización, y la alta aceptación por parte de los usuarios sugiere que el agente inteligente podría contribuir significativamente a mejorar la competitividad del sector turístico en Lambayeque.

## **Conclusiones**

El desarrollo de un agente inteligente recomendador de atractivos turísticos para Lambayeque ha demostrado ser una solución eficaz para abordar la baja afluencia de visitantes en la región. A través de una combinación de técnicas de inteligencia artificial y un modelo de recomendación basado en contenido, el sistema proporciona recomendaciones personalizadas que responden a los intereses específicos de los turistas, optimizando su experiencia de

planificación y facilitando el acceso a información relevante. Este enfoque contribuye al fortalecimiento del sector turístico en Lambayeque, haciendo que los atractivos menos conocidos sean más visibles y accesibles, lo cual potencialmente incrementa la competitividad de la región frente a otros destinos turísticos.

La implementación de la metodología CRISP-DM y el uso de Scrum fueron fundamentales para estructurar y ejecutar el proyecto de manera ordenada y flexible. Se identificaron y documentaron las variables clave para la recomendación de atractivos turísticos, lo cual fue fundamental para construir un sistema adaptado a las preferencias y características específicas de los usuarios. Asimismo, en la fase de modelado, utilizando la técnica de similitud de coseno y la vectorización TF-IDF, se logró un equilibrio adecuado entre diversidad y relevancia en las recomendaciones, cumpliendo así con el objetivo de proporcionar una experiencia personalizada sin redundancias. Este cumplimiento de objetivos se alinea con la hipótesis de que un sistema de recomendación adaptado a las preferencias individuales puede mejorar significativamente la satisfacción de los visitantes y la percepción de Lambayeque como destino turístico.

Asimismo, la evaluación de aceptación tecnológica realizada con el modelo TAM reveló una respuesta favorable de los usuarios hacia el sistema, indicando que perciben la herramienta como útil y fácil de usar. Estos hallazgos subrayan que la aplicación tiene un alto potencial de adopción y puede convertirse en un recurso valioso para el sector turístico, no solo para mejorar la experiencia de los visitantes, sino también para apoyar el desarrollo económico y cultural de la región.

El presente estudio logra cumplir con los objetivos planteados, ofreciendo una herramienta tecnológica innovadora y práctica para el turismo en Lambayeque. La naturaleza escalable y adaptable de la plataforma permitirá que evolucione y se ajuste a nuevas demandas del turismo en la era digital. Se recomienda realizar estudios futuros que consideren la integración de datos en tiempo real, como las condiciones climáticas y la estacionalidad, para enriquecer aún más las recomendaciones y contribuir de forma continua al fortalecimiento del turismo regional.

## **Recomendaciones**

A pesar de los logros alcanzados, se identifican áreas de mejora y oportunidades para futuras investigaciones. A continuación, se presentan recomendaciones para futuras investigaciones que deseen tomar como base el proyecto de tesis sobre el sistema de recomendación basado en contenido para atractivos turísticos en el departamento de Lambayeque.

### **Profundizar en la recolección de datos**

Es fundamental contar con un conjunto de datos robusto y diverso que incluya no solo las características de los atractivos turísticos, sino también las preferencias y valoraciones de los usuarios. Se sugiere implementar encuestas o entrevistas para obtener información cualitativa que complemente los datos cuantitativos.

### **Integrar otras tecnologías de aprendizaje automático y análisis de datos**

Se recomienda la integración de otras tecnologías de aprendizaje automático, como el aprendizaje profundo y el procesamiento de lenguaje natural, para mejorar la precisión y robustez del sistema de recomendación de atractivos turísticos. Estas tecnologías permitirán analizar patrones complejos en las preferencias de los usuarios y extraer información valiosa de reseñas y comentarios, lo que resultará en recomendaciones más personalizadas y relevantes, optimizando así la experiencia del turista.

### **Incorporar feedback de usuarios**

Implementar un mecanismo para que los usuarios puedan proporcionar retroalimentación sobre las recomendaciones recibidas. Esto no solo ayudará a mejorar el sistema a lo largo del tiempo, sino que también permitirá ajustar las recomendaciones a las preferencias cambiantes de los usuarios.

### **Explorar integraciones con otras plataformas**

Investigar la posibilidad de integrar el sistema de recomendación con plataformas de redes sociales o aplicaciones de viajes. Esto podría aumentar la visibilidad del sistema y facilitar la recolección de datos sobre las preferencias de los usuarios.

## Referencias

- [1] M. D. P. Flores Heredia, “Factores que determinan la permanencia del turista que visita la región Lambayeque,” Universidad Señor de Sipán, Pimentel, 2021. Accessed: Dec. 01, 2024. [Online]. Available: <https://hdl.handle.net/20.500.12802/8320>
- [2] K. Al Fararni, F. Nafis, B. Aghoutane, A. Yahyaouy, J. Riffi, and A. Sabri, “Hybrid recommender system for tourism based on big data and AI: A conceptual framework,” *Big Data Mining and Analytics*, vol. 4, no. 1, pp. 47–55, Mar. 2021, doi: 10.26599/BDMA.2020.9020015.
- [3] M. J. González Padrón, “Sistema inteligente de recomendación de restaurantes,” Universidad de La Laguna, La Laguna, 2022. Accessed: Dec. 01, 2024. [Online]. Available: <http://riull.ull.es/xmlui/handle/915/29406>
- [4] M. A. Andrade Espinoza and G. F. Marcillo Carbo, “Desarrollo de un sistema web inteligente mediante redes neuronales para seleccionar el mejor atractivo turístico y cultural del GAD de la ciudad de Pasaje mediante datos actuales e históricos.,” Universidad de Guayaquil, Guayaquil, 2022. [Online]. Available: <http://repositorio.ug.edu.ec/>
- [5] G. F. Reyes Pullugando, “Implementación de un chatbot para la atención y resolución de inquietudes turísticas en la ciudad de baños de agua santa,” Universidad técnica de Ambato Facultad de Ingeniería en Sistemas, Electrónica e Industrial, Ambato, 2013.
- [6] F. Ricci, L. Rokach, and B. Shapira, “Introduction to Recommender Systems Handbook,” in *Recommender Systems Handbook*, Springer US, 2011, pp. 1–35. doi: 10.1007/978-0-387-85820-3\_1.
- [7] S. Russell and P. Norvig, *Inteligencia artificial: Un Enfoque Moderno*, 2nd ed. El Cid Editor, 2014.
- [8] L. A. Yong Varela, L. A. Tovar Rivas, and J. Chaparro, “Modelo de aceptación tecnológica (tam): un estudio de la influencia de la cultura nacional y del perfil del usuario en el uso de las tic,” vol. 20, Bogotá, Jan. 2010. [Online]. Available: <http://www.redalyc.org/articulo.oa?id=81819028014>