**Formato de Proyecto de Machine Learning**

**Título del Proyecto**

*Travel Smart IA Recomendador Inteligente de Viajes Personalizados*

**Integrantes**

*Juan David Urán Gómez*

*Curso de Inteligencia Artificial*

**1. Definición del Problema**

* **Descripción del problema:**  
  *En la actualidad, los viajeros enfrentan una gran sobrecarga de información al planificar un viaje: precios, destinos, actividades, reseñas y disponibilidad. Este exceso de datos hace que tomar una decisión adecuada sea un proceso complejo y lento. Travel Smart IA busca resolver este problema utilizando Inteligencia Artificial para ofrecer recomendaciones de viajes personalizadas, basadas en los intereses, presupuesto, tiempo disponible y experiencias previas de otros usuarios.*
* **Objetivos del proyecto:**
* ***Objetivo General:***

*Desarrollar un sistema de recomendación inteligente que sugiera opciones de viaje personalizadas utilizando técnicas de Machine Learning y Procesamiento de Lenguaje Natural (NLP).*

* ***Objetivos Específicos:***

*-Implementar un modelo de recomendación híbrido que combine preferencias del usuario con reseñas de otros viajeros.*

*-Utilizar NLP para interpretar solicitudes en lenguaje natural.*

*-Aplicar análisis de sentimientos para evaluar reseñas y priorizar destinos con mejor valoración.*

*-Incorporar modelos predictivos para estimar variaciones de precios y mejores épocas para viajar.*

*-Diseñar un flujo automatizado que devuelva recomendaciones claras con itinerarios y costos estimados.*

* **Justificación:**  
  *Social: Facilita la toma de decisiones de los usuarios, permitiendo viajes más accesibles y satisfactorios.*

*Económica: Ayuda a optimizar el presupuesto de los viajeros al recomendar mejores momentos para comprar.*

*Tecnológica: Integra técnicas de NLP, sistemas de recomendación y análisis de datos en un solo proyecto.*

*Académica: Sirve como caso práctico para aplicar conocimientos de IA, Machine Learning y análisis de datos.*

**2. Recolección de Datos**

**Fuente de los datos:**  
*-Datos abiertos de turismo (ejemplo: Kaggle, Open Data de aerolíneas, Booking, TripAdvisor).*

*-Datos simulados para pruebas iniciales.*

*-Futuro: integración con APIs de viajes (Skyscanner, Amadeus, Expedia).*

**Descripción del dataset:**  
*Registros: [ejemplo: 50,000 entradas].*

*Variables:*

*-Destino (categoría).*

*-Precio (numérico).*

*-Duración del viaje (numérico).*

*-Actividades disponibles (texto).*

*-Número de personas (numérico).*

*-Comentarios/reseñas (texto).*

*-Clima/temporada (categoría).*

*Tipo de datos: Mixto (numérico, categórico y texto).*

**Herramientas utilizadas:**  
*-Lenguaje: Python.*

*-Entorno: Google Colab / Jupyter Notebook.*

*-Librerías: pandas, numpy, scikit-learn, matplotlib, seaborn, NLTK/spaCy, TensorFlow/PyTorch.*

*-Fuentes de datasets: Kaggle, APIs abiertas de viajes.*

**3. Preparación de los Datos**

**Limpieza de datos:**  
*-Eliminación de registros duplicados.*

*-Tratamiento de valores nulos (relleno con media/moda o eliminación según el caso).*

*-Corrección de inconsistencias en nombres de destinos o actividades.*

* **Transformación de variables:**  
  *-Normalización de precios y duración para que estén en la misma escala.*

*-Codificación de variables categóricas (ej. destino, temporada).*

*-Representación de texto (reseñas y actividades) con técnicas de NLP como TF-IDF o embeddings.*

*-Creación de nuevas variables como costo por día o popularidad del destino.*

**Visualización y análisis exploratorio:**  
*-Histogramas de distribución de precios y duración de viajes.*

*-Gráficos de barras con destinos más populares.*

*-Wordcloud de las actividades más solicitadas.*

*-Análisis de sentimientos de comentarios (positivo, negativo, neutro).*

**4. Selección y Entrenamiento del Modelo**

**Modelo(s) seleccionado(s):**  
***Sistema de recomendación híbrido:***

*-Filtrado basado en contenido (comparación de características de destinos).*

*-Filtrado colaborativo (similitud entre usuarios con gustos similares).*

*-Se puede implementar con K-Nearest Neighbors (KNN) o Matriz de factorización.*

***NLP para análisis de reseñas:***

*-TF-IDF + Regresión Logística o Naive Bayes para análisis de sentimientos.*

*-Alternativa: BERT preentrenado (transfer learning).*

***Predicción de precios:***

*-Regresión lineal múltiple o XGBoost para estimar precios de vuelos/hoteles en función de temporada, destino y demanda.*

* **Motivo de la selección:**

-*Los modelos son interpretables, lo que facilita explicar resultados en un entorno académico.*

*-Combinan simplicidad y potencia: algoritmos clásicos para datos estructurados (regresiones, KNN, árboles) y modelos más complejos para texto (transformers).*

*-Permiten cubrir todos los componentes: recomendación, NLP y predicción.*

**Entrenamiento del modelo:**  
*-División del dataset en 80% train – 20% test.*

*-Para NLP, preprocesamiento con tokenización, stopwords removal y vectorización TF-IDF.*

***Ajuste de hiperparámetros:***

*-K en KNN → 5 a 15.*

*-Profundidad de árboles de decisión → máx. 10.*

*-Épocas en redes neuronales → 5 a 10 con early stopping.*

*Validación cruzada (k-fold = 5).*

**5. Evaluación del Modelo**

**Métricas utilizadas:**  
***Para recomendación:***

*-Precision@k, Recall@k, NDCG (Normalized Discounted Cumulative Gain).*

***Para análisis de sentimientos (NLP):***

*-Accuracy, Precision, Recall, F1-score.*

***Para predicción de precios:***

*-RMSE (Root Mean Squared Error).*

*-MAE (Mean Absolute Error).*

*-R² (Coeficiente de determinación).*

* **Resultados obtenidos:**  
  *-Análisis de sentimientos con TF-IDF + Logistic Regression → Accuracy 85%.*

*-Recomendación híbrida → Precision@5 = 78%.*

*-Predicción de precios con XGBoost → RMSE = 52.4 USD.*

**Comparación entre modelos (si aplica):**  
*-Regresión lineal fue más simple pero menos precisa para precios (RMSE alto).*

*-XGBoost funcionó mejor para predicciones numéricas.*

*-En NLP, modelos clásicos (TF-IDF + Naive Bayes) funcionan bien, pero BERT mejora el rendimiento en un +5% F1.*

**6. Despliegue y Comunicación de Resultados**

**Implementación / Aplicación práctica:**  
***Prototipo en Streamlit o Flask que:***

*1.Reciba preferencias del usuario en lenguaje natural.*

*2.Procese la solicitud con NLP.*

*3.Devuelva recomendaciones de destinos, itinerario sugerido y precio estimado.*

*Visualización de resultados en dashboard interactivo (ej. con Power BI o Plotly).*

* **Conclusiones del proyecto:**  
  *-Se logró diseñar un sistema de recomendación inteligente que integra preferencias del usuario, reseñas y precios.*

*-Se aplicaron diferentes técnicas de IA: ML para predicción, NLP para análisis de texto y recomendadores híbridos.*

*-Se evidenció que modelos como XGBoost y BERT ofrecen mejoras significativas en precisión frente a algoritmos más básicos.*

*-Dificultades: integración de datos reales y balance entre costo computacional y precisión de modelos.*

**Trabajo futuro:**  
*-Integración con APIs de aerolíneas y hoteles en tiempo real.*

*-Mejorar el motor de NLP con modelos grandes como GPT.*

*-Incluir un planificador de itinerarios optimizado (algoritmos de grafos).*

*-Crear un chatbot de viajes disponible en WhatsApp o Web.*

*-Escalar la solución a una app móvil con recomendaciones en tiempo real.*

**Anexos**

* **Código en Python / Notebook (enlace o adjunto).**
* **Fuentes bibliográficas o referencias de datos.**