



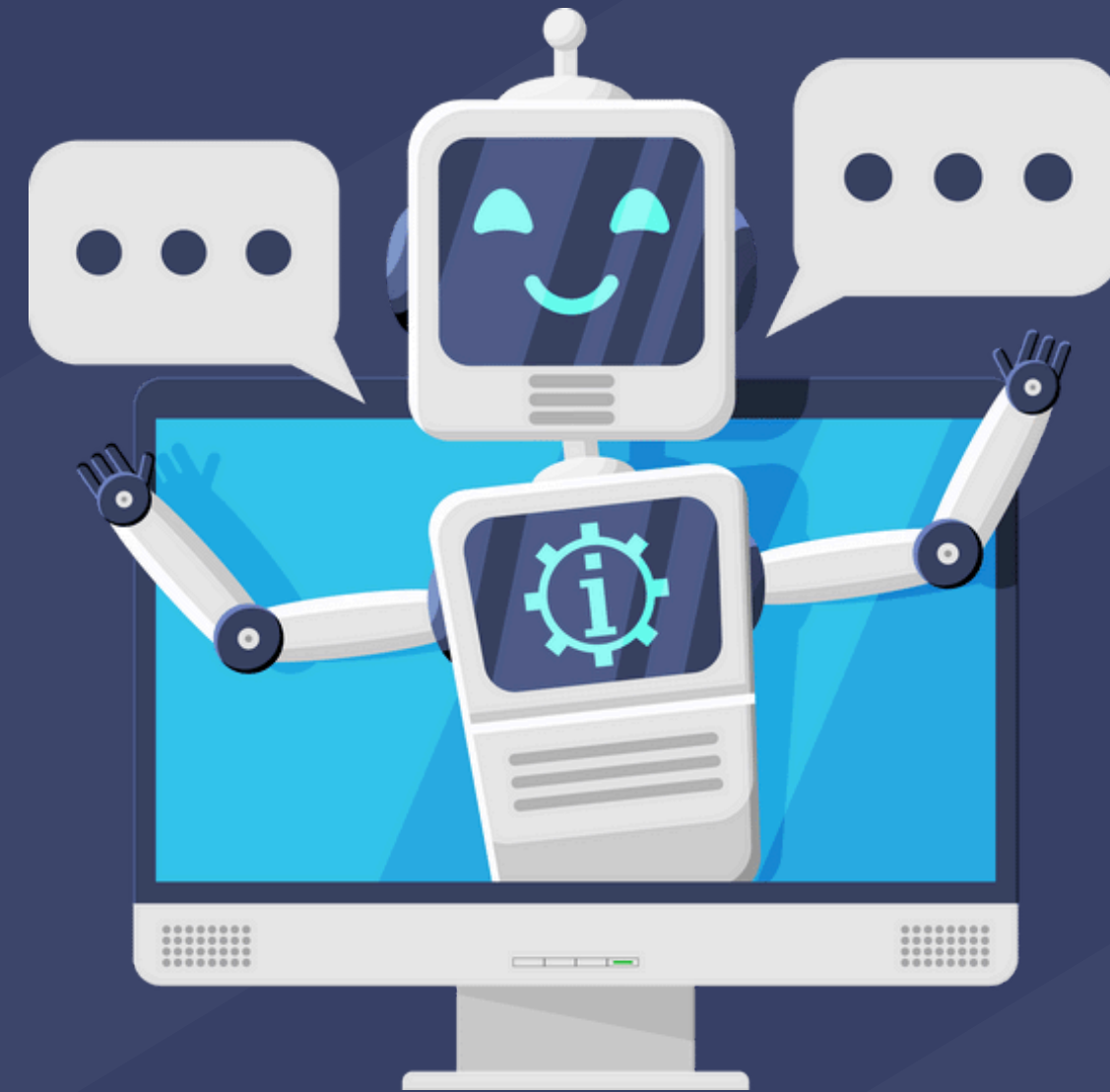
# APLICACIÓN WEB PARA DETECCIÓN DE DIABETES



**INTELIGENCIA ARTIFICIAL.**

PROYECTO REALIZADO POR:

- CALVOPINA ORELLANA DANIEL SEBASTIAN.
- CHAFLA VINUEZA ALFONSO ALEJANDRO.
- JÁCOME SARMIENTO XAVIER SEBASTIÁN
- JIMÉNEZ ELIZALDE JOSUÉ.
- MENDOZA VALDEZ EMILY ISABELA.





# Fases de desarrollo



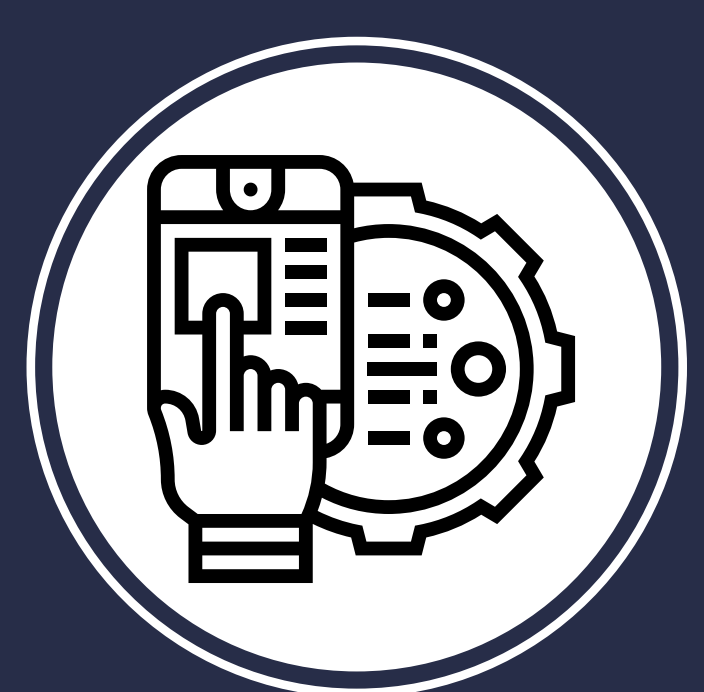
**Selección de  
variables.  
Limpieza y  
normalización de data**



**Desarrollo y  
Entrenamiento de  
modelos.**



**Selección de  
modelo.**



**Desarrollo de  
aplicación**



# Selección de variables. Limpieza y normalización de data

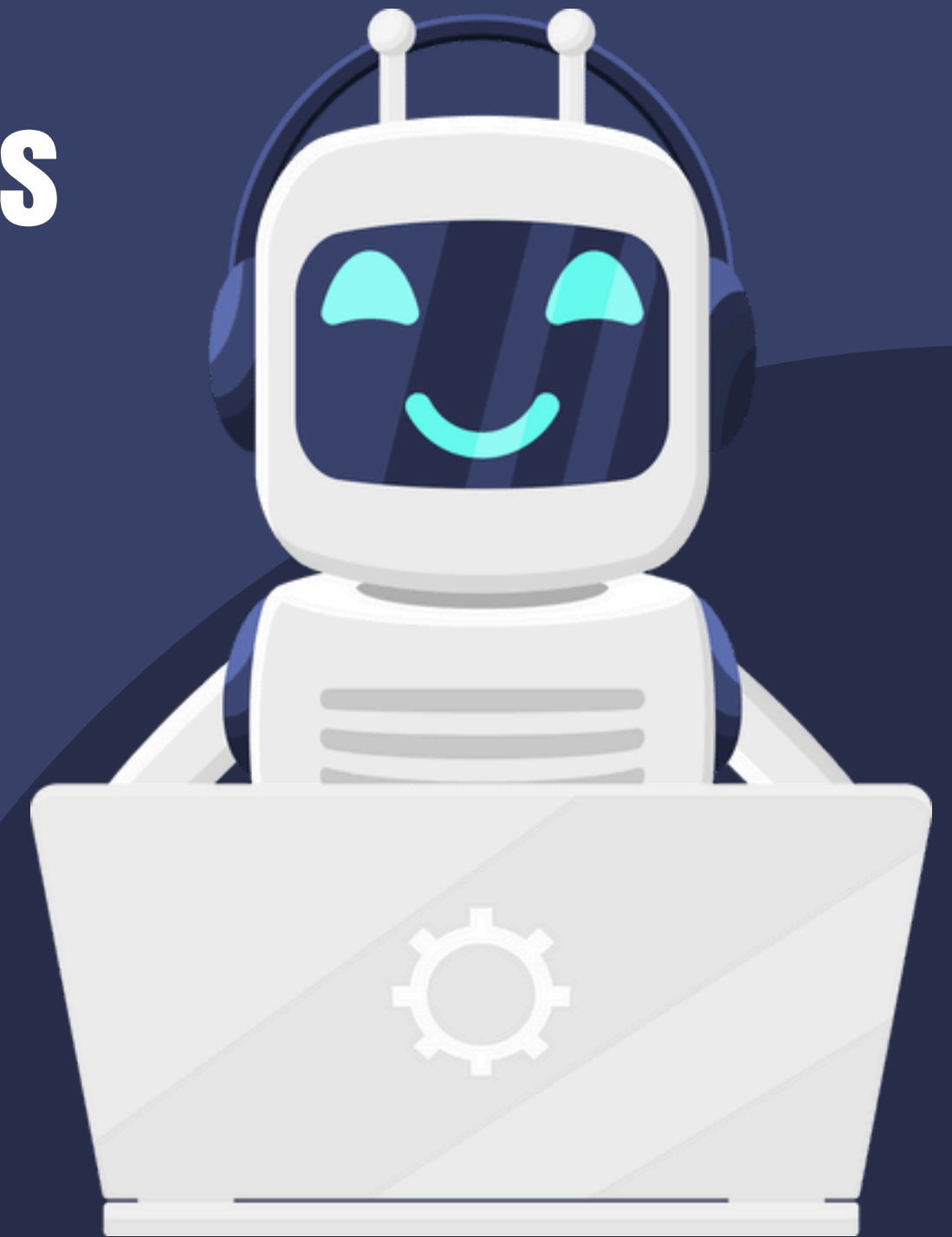
- **Carga y análisis inicial:** No se detectaron valores nulos (`df.info()` y `df.isnull().sum()`).
- **Clasificación de variables:**
  - Numéricas: solo BMI, MentHlth, PhysHlth.
  - Categóricas: resto (edad, sexo, alimentación, salud, etc.).
- **Normalización:** aplicada únicamente a BMI, por ser la única continua relevante.
- **Variables categóricas:** transformadas a dummies, eliminando la primera columna de cada grupo para evitar colinealidad.
- **Depuración final:** Se descartaron variables no relevantes una vez terminada la transformación, dejando solo las seleccionadas para el modelo.



# Desarrollo y entrenamiento de modelos

## Modelos evaluados:

- **K-Nearest Neighbors (KNN)**
  - Se probó con k de 1 a 30 (método del codo).
  - Mejor precisión con k=9.
  - Uso de KNeighborsClassifier de sklearn.neighbors.
- **Regresión Logística**
  - Implementado con LogisticRegression (solver='liblinear', class\_weight='balanced').
  - Ajustado para manejar desbalance de clases.
  - Mejor desempeño general y consistente en clasificación binaria.
- **Árboles de Decisión**
  - Se usó DecisionTreeClassifier con parámetros ajustados (max\_depth, min\_samples\_split, etc.).
  - Buena precisión, pero riesgo de sobreajuste.
  - Ventaja: alta interpretabilidad del modelo.



# SELECCIÓN DE MODELO

	precision	recall	f1-score	support
False	0.75	0.70	0.72	6172
True	0.71	0.76	0.74	6162
accuracy			0.73	12334
macro avg	0.73	0.73	0.73	12334
weighted avg	0.73	0.73	0.73	12334

## Desarrollo del modelo:

- Se entrenó un modelo de regresión logística utilizando los datos preprocesados.
- Se transformaron las variables categóricas a tipo dummy (one-hot encoding).

## Resultados:

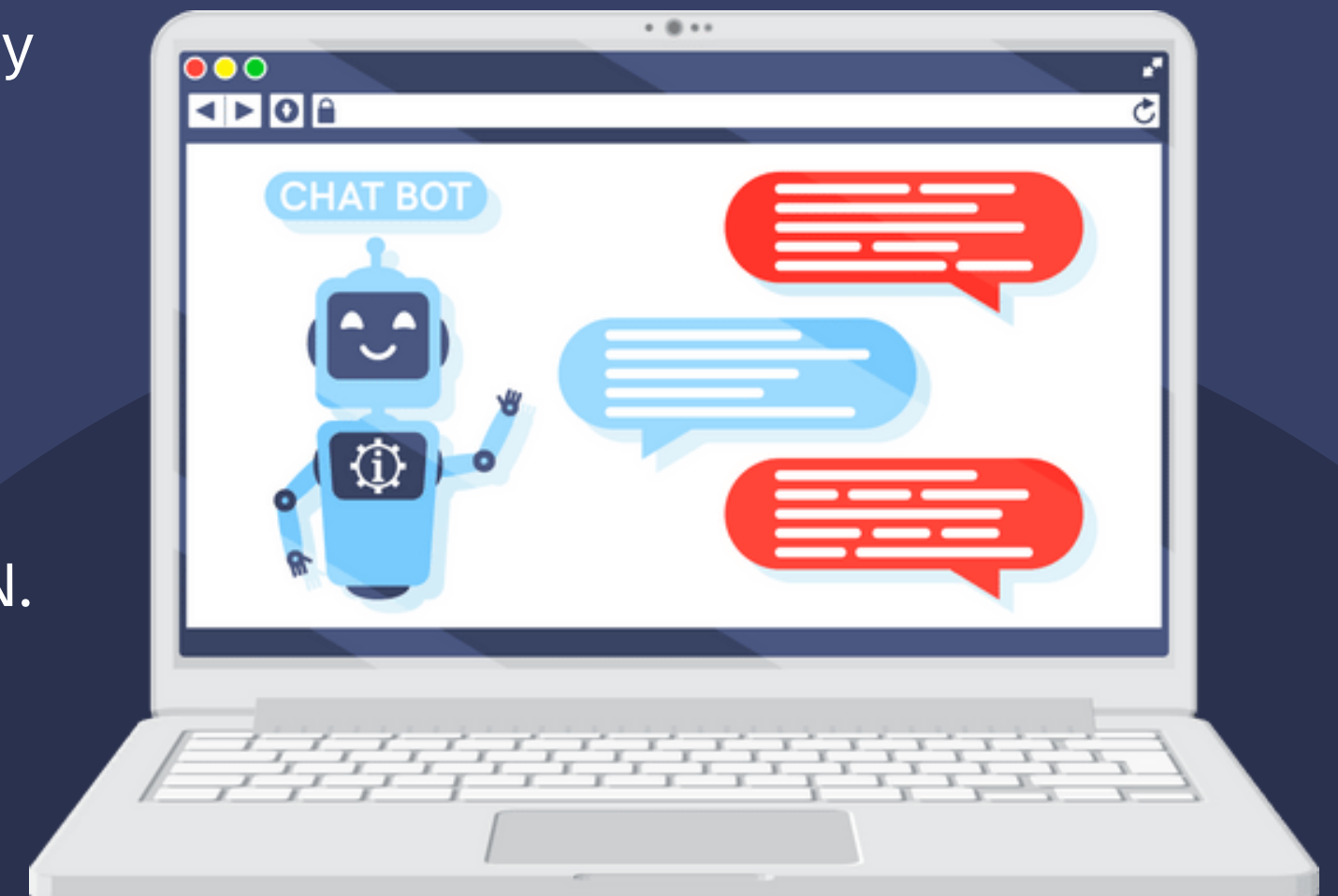
- El modelo presentó un balance sólido entre precisión y recall, superior al de otros algoritmos evaluados (como K-NN).
- Se comportó de forma estable y confiable, sin sobreajustarse a los datos.
- Su interpretación es clara y útil para aplicaciones médicas, donde se necesita entender el impacto de cada variable





# Desarrollo de aplicación

- **Frontend (HTML + CSS):**
  - Formulario web intuitivo con campos de tipo Sí/No, edad y BMI.
  - Diseño limpio y responsivo para facilitar el uso.
- **Captura y envío de datos (JavaScript):**
  - Conversión de respuestas en valores booleanos o numéricos.
  - Envío de los datos al backend vía `fetch()` en formato JSON.
- **Backend (API Flask):**
  - Recepción de datos desde el frontend.
  - Uso de un modelo de regresión logística para predecir riesgo de diabetes.
  - Respuesta en JSON con el resultado.
- **Visualización dinámica:**
  - El resultado se muestra en pantalla sin recargar la página.



**MUCHAS  
GRACIAS**

