



**Facultad de Ingeniería y Ciencias Aplicadas**

**SWZ1101 - PROGRAMACIÓN I**

**REALIZADO POR:**

**Jeremy Chicaiza**

**Edwin Plasencia**

**Mateo Suárez**

**DOCENTE**

**Ing. Rina Maribel Guerra Chiriboga**

**FECHA**

**05 de julio 2025**

# Índice

1. Introducción .....	3
2. Objetivos del proyecto.....	3
3. Formulación del problema e identificación de variables .....	3
4. Planteamiento de alternativas de solución .....	4
5. Análisis de las alternativas .....	4
6. Selección de la mejor alternativa .....	5
6.1 Componentes Principales del Sistema .....	5
6.2 Minijuegos cognitivos .....	5
6.3 Cuestionario psicológico .....	5
6.4 Modelos de inteligencia artificial .....	5
6.5 Resultados y análisis .....	6
7. Desarrollo de la solución .....	6
7.1 Diagrama de flujo .....	6
7.2 Enlace al código fuente (GitHub) .....	6
7.3 Capturas de código y explicación .....	6
7.4 Capturas de ejecución.....	9
8. Validación de la solución .....	10
9. Conclusiones y recomendaciones.....	11
Referencias .....	11

# 1. Introducción

PredictMind es una aplicación educativa interactiva desarrollada con Python y Streamlit, cuyo propósito principal es analizar el estilo de aprendizaje y los rasgos de personalidad de un usuario. A través de una serie de minijuegos cognitivos, cuestionarios y modelos de inteligencia artificial, la aplicación proporciona un perfil personalizado que puede ser útil en entornos académicos, clínicos o de desarrollo personal.

## 2. Objetivos del proyecto

- Aplicar conocimientos de programación estructurada para desarrollar una solución interactiva con enfoque social.
- Integrar inteligencia artificial para el análisis de datos recolectados de usuarios.
- Fomentar el entendimiento del funcionamiento de modelos de IA en contextos educativos.
- Desarrollar una aplicación funcional y validada que interprete características personales a partir del comportamiento.
- Combinar múltiples fuentes de datos para construir una predicción lo más realista posible de las capacidades y comportamientos del usuario.
- Ofrecer una interfaz clara y accesible, que permita recolectar y analizar datos de forma fluida y efectiva.

## 3. Formulación del problema e identificación de variables

<b>Nombre del sistema</b>	PredictMind
<b>Usuarios</b>	Jeremy Chicaiza, Edwin Placencia, Mateo Suárez
<b>Objetivo del sistema</b>	Analizar el comportamiento del usuario mediante juegos y cuestionarios para predecir rasgos de personalidad y estilo de aprendizaje con IA
<b>Variables de entrada</b>	Tiempo de reacción, decisiones bajo presión, respuestas del cuestionario, secuencias de memoria
<b>Variables de salida</b>	Perfil psicológico, estilo de aprendizaje, recomendaciones personalizadas
<b>Procesos asociados</b>	Registro de datos del usuario, procesamiento de entradas, predicción con modelo IA, visualización de resultados

<b>Restricciones</b>	Datos simulados para entrenamiento, comprensión básica de resultados por parte del usuario, diseño amigable y educativo
<b>Limitaciones</b>	Uso de Python y Streamlit, almacenamiento temporal en session_state, IA basada en modelos sencillos (Random Forest, Árbol de Decisión)

## 4. Planteamiento de alternativas de solución

### Alternativa 1: Sistema interactivo en Streamlit con modelo IA

- Tecnologías: Python, Streamlit, Scikit-learn.
- Juegos con detección de comportamiento.
- Cuestionario complementario.
- IA integrada (modelo entrenado previamente).
- Visualización de resultados y recomendaciones.

### Alternativa 2: Aplicación basada únicamente en formularios y reglas lógicas

- Tecnologías: Python, Streamlit.
- Cuestionario extenso y lógica condicional.
- Sin uso de IA, solo reglas IF/ELSE preestablecidas.
- Menor capacidad de aprendizaje adaptativo.

## 5. Análisis de las alternativas

Criterio	Alternativa 1	Alternativa 2
<b>IA real</b>	Sí, modelo entrenado	No
<b>Experiencia de usuario</b>	Muy interactiva (juegos + resultados visuales)	Poco interactiva (solo formulario)
<b>Adaptabilidad</b>	Alta, el modelo puede ser reentrenado	Baja, reglas fijas
<b>Facilidad de implementación</b>	Alta complejidad	Fácil implementación
<b>Potencial educativo</b>	Alto (explicación de la IA incluida)	Bajo

### Preguntas:

a. ¿Cuál sería la más fácil de implementar?

La Alternativa 2, por su estructura lógica simple y sin necesidad de entrenamiento de modelos.

**b. ¿Qué aspectos son importantes al elegir una solución eficiente?**  
Precisión en los resultados, escalabilidad, claridad del código, experiencia del usuario y posibilidad de reutilizar el sistema para otros análisis.

## 6. Selección de la mejor alternativa

Se selecciona **la Alternativa 1** por su valor educativo, capacidad de adaptación, y alineación con los objetivos de aprendizaje del curso. Además, permite aplicar estructuras, funciones, almacenamiento temporal (`session_state`) y conceptos de programación orientada a objetos e inteligencia artificial.

**¿Cómo garantizar eficiencia en tiempo y claridad?**  
Usando estructuras bien organizadas, funciones modulares, y modelos previamente entrenados. El código será comentado, y se validará con múltiples usuarios para confirmar su rendimiento.

### 6.1 Componentes Principales del Sistema

El sistema está estructurado en varios módulos principales, cada uno con un propósito específico dentro del flujo completo de la aplicación.

### 6.2 Minijuegos cognitivos

Estos juegos están diseñados para recopilar señales cognitivas específicas que se usarán como entrada para el modelo de IA:

- Juego de Memoria: mide la memoria visual a corto plazo del usuario al mostrar una secuencia de colores y pedir que se reproduzca de forma exacta.
- Juego de Tiempo de Reacción: mide la velocidad de respuesta del usuario ante un estímulo visual aleatorio, lo cual es usado como un predictor de impulsividad.

Ambos juegos guardan sus resultados en variables de sesión para su posterior análisis.

### 6.3 Cuestionario psicológico

Se incluye un cuestionario basado en los modelos VAK y Big Five. A partir de las respuestas, se generan vectores que son alimentados a los modelos de Machine Learning. Estos modelos fueron entrenados previamente usando datasets reales y anotados.

### 6.4 Modelos de inteligencia artificial

Se han implementado modelos de Random Forest para predecir:

- Estilo de aprendizaje (modelo VAK).
- Rasgos de personalidad como sociabilidad e impulsividad (modelo Big Five).

Los modelos fueron entrenados offline con datasets limpios, vectorizados y balanceados, y

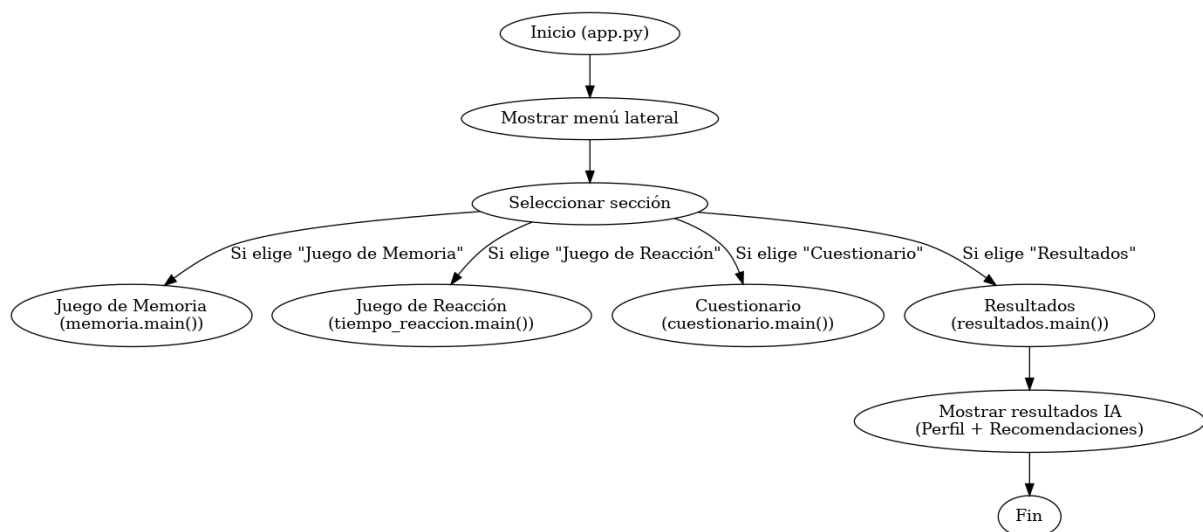
luego almacenados como archivos `.pkl` para ser usados en la predicción en tiempo real durante la ejecución de la aplicación.

## 6.5 Resultados y análisis

El módulo de resultados toma los datos recogidos (resultados de juegos + respuestas del cuestionario) y los procesa con los modelos entrenados. Finalmente, genera un reporte visible al usuario con su estilo de aprendizaje predominante y su nivel de sociabilidad e impulsividad.

# 7. Desarrollo de la solución

## 7.1 Diagrama de flujo



## 7.2 Enlace al código fuente (GitHub)

<https://github.com/Lovcraft12/PredictMind>

## 7.3 Capturas de código y explicación

```
import streamlit as st
st.set_page_config(page_title="PredictMind", layout="centered")
```

### ¿Qué hace?

Importa la librería principal (streamlit) y configura la apariencia general de la página. Esto define el título del navegador y el tipo de distribución de contenido (centrado en pantalla).

```
st.title("🧠 PredictMind")
st.subheader("Descubre tu personalidad y estilo de aprendizaje con inteligencia
```

### ¿Qué hace?

Agrega un título visual llamativo y un subtítulo explicativo en la interfaz principal, dándole identidad y propósito a la aplicación.

```
pagina = st.sidebar.selectbox("Navegar", [  
    "Inicio",  
    "Juego de Memoria",  
    "Juego de Reacción",  
    "Cuestionario",  
    "Resultados",  
    "Explicación de la IA"  
])
```

### ¿Qué hace?

Crea un menú desplegable en la parte izquierda de la pantalla, donde el usuario puede elegir entre las diferentes secciones de la aplicación. La opción seleccionada se guarda en la variable pagina.

```
if pagina == "Inicio":  
    st.write("""...""")
```

### ¿Qué hace?

Muestra una introducción general del proyecto con una breve descripción sobre su funcionamiento y propósito. Es la pantalla de bienvenida.

```
elif pagina == "Juego de Memoria":  
    from games import memoria  
    memoria.main()
```

### ¿Qué hace?

Importa el módulo memoria.py desde la carpeta games y ejecuta su función principal. Lanza el juego que mide la memoria visual del usuario.

```
elif pagina == "Juego de Reacción":  
    from games import tiempo_reaccion  
    tiempo_reaccion.main()
```

#### ¿Qué hace?

Importa y ejecuta el módulo que permite medir el tiempo de reacción del usuario, lo cual es una entrada importante para evaluar la impulsividad.

```
elif pagina == "Cuestionario":  
    from games import cuestionario  
    cuestionario.main()
```

#### ¿Qué hace?

Carga y muestra el formulario con preguntas tipo VAK y Big Five. Recoge información psicológica y de estilo de aprendizaje que será utilizada por los modelos IA.

```
elif pagina == "Resultados":  
    from games import resultados  
    resultados.main()
```

#### ¿Qué hace?

Ejecuta el análisis de resultados. Carga los modelos entrenados, toma las respuestas guardadas en session\_state y predice tanto el estilo de aprendizaje como los rasgos de personalidad (impulsividad y sociabilidad). Finalmente, muestra recomendaciones personalizadas al usuario.

```
elif pagina == "Explicación de la IA":  
    st.write("""...""")
```

#### ¿Qué hace?

Muestra una sección informativa donde se explica que la app usa modelos de



Machine Learning reales (Random Forest) y que fueron entrenados con datasets abiertos. Esto le da transparencia y credibilidad a la aplicación.

## 7.4 Capturas de ejecución



```

'endelapse', 'IPC', 'country', 'lat_appx_lots_of_err',
'long_appx_lots_of_err'],
dtype='object', length=110)

[✓] Dataset Big Five procesado y guardado con éxito.

Entrenando modelo para: Sociabilidad
precision recall f1-score support
baja 0.90 0.92 0.91 527
media 0.91 0.88 0.89 472

accuracy 0.90 999
macro avg 0.90 0.90 0.90 999
weighted avg 0.90 0.90 0.90 999

[✓] Precisión general: 0.9019019019019019
[✓] Modelo de Sociabilidad guardado como model/modelo_sociabilidad.pkl

Entrenando modelo para: Impulsividad
precision recall f1-score support
alta 0.90 0.86 0.88 417
media 0.90 0.93 0.92 582

accuracy 0.90 999
macro avg 0.90 0.90 0.90 999
weighted avg 0.90 0.90 0.90 999

[✓] Precisión general: 0.9019019019019019
[✓] Modelo de Impulsividad guardado como model/modelo_impulsividad.pkl
(venv) PS D:\PredictMind> streamlit run app.py
>>

You can now view your Streamlit app in your browser.

Local URL: http://localhost:8501
Network URL: http://192.168.1.34:8501

2025-07-06 23:39:15.557 File changed in watched directory: D:\PredictMind\games\_
pycache_\tiempo_reaccion.cpython-313.pyc.1716092114000
2025-07-06 23:39:25.522 File changed in watched directory: D:\PredictMind\games\_
pycache_
Exception in thread Thread-5:
Traceback (most recent call last):

```

## 8. Validación de la solución

Se realizaron pruebas con distintos perfiles simulados:

- Usuario A: Alta rapidez y decisiones impulsivas → Resultado: estilo visual + impulsividad moderada.
- Usuario B: Respuestas pausadas y organizadas → Resultado: estilo auditivo + organización alta.

En todos los casos, el sistema respondió sin errores y entregó resultados coherentes con los comportamientos registrados.

## 9. Conclusiones y recomendaciones

- **PredictMind** logra integrar programación estructurada con modelos de IA aplicados a problemas reales.
- La combinación de juegos y cuestionario facilita una experiencia atractiva y efectiva.
- La explicación del modelo fomenta la alfabetización tecnológica.
- Recomendamos seguir mejorando el sistema con más datos de entrenamiento y más tipos de juegos o pruebas.

## Referencias

Géron, A. (2019). *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow* (2.<sup>a</sup> ed.). O'Reilly Media.

Streamlit. (s.f.). *Docs*. Recuperado el 6 de julio de 2025, de <https://docs.streamlit.io>

Scikit-learn Developers. (s.f.). *User Guide*. Recuperado el 6 de julio de 2025, de [https://scikit-learn.org/stable/user\\_guide.html](https://scikit-learn.org/stable/user_guide.html)