

## **Asignatura**

Sistemas Interactivos Inteligentes

## **Proyecto Final**

Plataforma Multimodal de Bienestar: ***ViveBien***

Hugo Iglesias Pombo

Carlota Fernández del Riego

## Índice

1. Descripción general	3
2. Descripción Funcional del Sistema	3
3. Planificación inicial y cambios realizados	4
4. Trabajo interno no visible	4
5. Requisitos y ejecución del sistema	5
6. Estado final del proyecto	5

## 1. Descripción general

La Plataforma Multimodal de Bienestar *ViveBien* es un sistema interactivo diseñado con el propósito de monitorizar y apoyar el bienestar físico y emocional del usuario mediante una interfaz web accesible, clara y altamente visual. El proyecto combina diferentes modalidades de interacción (entrada textual, análisis automático del lenguaje y selección manual de estado emocional) junto con componentes multimedia y visualizaciones dinámicas.

El sistema se estructura en módulos independientes que gestionan el análisis emocional, la simulación de métricas de bienestar, la interacción mediante chat, el acceso a rutinas guiadas y la representación visual de hábitos, manteniendo una navegación fluida, coherente y fácil de utilizar. El diseño prioriza la claridad, la usabilidad y una arquitectura técnica flexible, capaz de ampliarse o modificarse sin comprometer el funcionamiento global.

## 2. Descripción Funcional del Sistema

ViveBien es una aplicación diseñada para monitorizar y mejorar el bienestar integral del usuario mediante seis módulos principales:

- **Inicio:**
  - Panel de control con calendario semanal interactivo.
  - Visualización rápida del nivel de energía (algoritmo calculado en base a horas de sueño, pasos y estado de ánimo).
  - Gráfico de donut para el seguimiento de la meta diaria de pasos.
- **Resumen de Métricas:**
  - Visualización avanzada de datos con anillos de actividad (estilo Apple Watch).
  - Gráficas de tendencias de las últimas 2 semanas (pasos, sueño, ritmo cardíaco).
  - Detección automática de patrones (ej. "Tu mejor día es el martes").
- **Registro de Estado:**
  - Permite al usuario registrar cómo se siente mediante texto o voz.
  - Análisis de Sentimiento: Clasifica la emoción (ej. "alegría", "tristeza") usando IA local (Robertuito).
  - Recomendación Inteligente: Genera un consejo breve y accionable usando IA local (Gemma-2b).
- **Chat con Aura (Asistente IA):**
  - Chatbot empático con memoria de conversación y acceso a los datos biométricos del usuario.
  - Capaz de ver tus pasos y horas de sueño para dar consejos personalizados.
  - Modo Voz: Puede hablar (TTS) y escuchar (STT) para una experiencia manos libres y accesibilidad.

- **Rutinas de Bienestar:**

- Catálogo de vídeos integrados para respiración guiada, estiramientos y yoga.
- Clasificados por nivel de dificultad y duración.

- **Perfil y Accesibilidad:**

- Configuración de metas personales.
- Suite de Accesibilidad: Modos de alto contraste, tamaño de fuente ajustable y lectura de voz automática para personas con dificultades visuales.

### 3. Planificación inicial y cambios realizados

El proyecto partía de un mockup inicialmente planteado para una aplicación móvil con seis pantallas principales (Acceso, Resumen, Chat con Aura, Rutinas Personalizadas, Menú y Perfil). Este diseño servía como guía funcional para desarrollar una plataforma orientada al bienestar físico y emocional mediante interacción natural y rutinas guiadas.

Durante el desarrollo se realizaron varios cambios necesarios respecto al planning inicial. El más significativo fue la migración completa del diseño móvil a un entorno web en Streamlit, lo que implicó adaptar la navegación, los layouts y la distribución visual. El chat previsto en el mockup se implementó finalmente como un módulo textual integrado con un modelo de análisis emocional, manteniendo la finalidad original del apartado. Las rutinas guiadas se conservaron, aunque el componente de control audiovisual fue descartado por complejidad técnica, y en su lugar, se integraron vídeos de YouTube para garantizar compatibilidad y rendimiento.

Las métricas definidas inicialmente (pasos, sueño y estado de ánimo) se ampliaron con energía, frecuencia cardíaca, desafío semanal, un calendario de hábitos y un análisis adicional con tendencias de las dos últimas semanas, incluyendo la opción de descargar un reporte comparativo. Paralelamente, se realizaron múltiples ajustes visuales como columnas, tipografías, colores, espaciados y gráficos, para adecuar el diseño al entorno web. A nivel técnico, el código fue completamente modularizado, se empleó `st.session_state` para la gestión del estado, se eliminaron duplicados y se unificaron funciones y estilos. Estos cambios permitieron adaptar el proyecto al nuevo entorno sin perder los objetivos planteados en el planning inicial.

### 4. Trabajo interno no visible

En cuanto al trabajo interno, se realizaron numerosas pruebas y procesos que no aparecen en la versión final, pero que fueron esenciales para construir un sistema estable. En primer lugar, se probaron distintos modelos de análisis emocional, incluyendo *pysentimiento/robertuito*, variantes basadas en *transformers* y Whisper. Estos dos últimos se descartaron por latencia elevada, tamaño excesivo y mala integración con Streamlit. El módulo de chat pasó por varias iteraciones destinadas a mejorar la

coherencia de las respuestas, la gestión de errores y la integración del estado emocional detectado en cada mensaje.

Por otra parte, se llevaron a cabo pruebas específicas para la simulación biométrica, ajustando funciones que generasen valores realistas de pasos, sueño, energía, frecuencia cardíaca y progreso semanal, y verificando que las métricas se mantuvieran dentro de rangos coherentes. Las visualizaciones requirieron de muchas pruebas para optimizar la posición del gráfico donut, el espaciado entre métricas, la colocación del calendario y el comportamiento del CSS personalizado. Finalmente, se realizaron tests de navegación para asegurar estabilidad entre páginas y evitar recargas problemáticas. Todo este trabajo interno permitió que la versión final resultase fluida, coherente y libre de errores visibles.

## 5. Requisitos y ejecución del sistema

La entrega final incluye todos los elementos necesarios para la ejecución del proyecto, así como los recursos técnicos y visuales generados durante su desarrollo. El sistema requiere **Python 3.10** o una versión superior, siendo recomendable utilizar un nuevo entorno virtual. Las dependencias están recogidas en el archivo **requirements.txt**, que incluye Streamlit, pysentimiento, transformers, torch, numpy, pandas y otras librerías empleadas en la interfaz y en el análisis emocional. Para el correcto funcionamiento de las rutinas personalizadas es necesaria la conexión a internet, ya que los vídeos se cargan desde YouTube.

La ejecución del sistema es sencilla y consta simplemente de cinco pasos:

1. **Clonar el repositorio:** git clone <https://github.com/carlotadelriego/ViveBien.git>
2. **Acceder al proyecto:** cd ViveBien
3. **Crear entorno virtual y activarlo (opcional):** python -m venv venv  
source venv/bin/activate # Mac/Linux  
venv\Scripts\activate # Windows
4. **Instalar dependencias:** pip install -r requirements.txt
5. **Ejecutar la aplicación:** streamlit run vivebien\_main.py

La primera carga del modelo emocional puede requerir unos segundos adicionales debido a la inicialización del modelo y sus recursos asociados.

## 6. Estado final del proyecto

El sistema final integra todos los componentes previstos de forma estable, incluyendo métricas simuladas, análisis emocional, chat funcional, rutinas multimedia y visualizaciones personalizadas. La arquitectura modular garantiza que cada apartado opere de forma independiente aunque coherente, permitiendo una navegación fluida y sin errores. El resultado mantiene la esencia del diseño original y adapta sus funcionalidades al entorno web de manera realista y optimizada, cumpliendo plenamente con los objetivos establecidos para este proyecto.