

# PROYECTO INTEGRADOR

## GuardiánClima ITBA

### Documentación del Proyecto

---

**Materia:** Tecnología

**Institución:** Instituto Tecnológico de Buenos Aires (ITBA)

#### **Integrantes del Grupo:**

- José Ignacio Aldaco
- Joaquín Nicolás Dominguez Gaviola
- Santiago Ranftl
- Santiago Ramón Garriga Zimmermann
- Natan Jonas Rosenhain

**Fecha de Entrega:** 16/06/2025

---

## Índice

- 1. Introducción**
    - 1.1. Descripción del Proyecto
    - 1.2. Objetivos
  - 2. Diseño y Arquitectura**
    - 2.1. Estructura General de la Aplicación
    - 2.2. Flujo de Datos y Menús
    - 2.3. Decisiones de Diseño
  - 3. Guía de Usuario Detallada**
    - 3.1. Menú de Acceso
    - 3.2. Menú Principal
  - 4. Desafíos y Soluciones**
  - 5. Conclusiones y Aprendizajes del Equipo**
-

# 1. Introducción

## 1.1. Descripción del Proyecto

"GuardiánClima ITBA" es una aplicación de consola desarrollada íntegramente en Python, diseñada como un proyecto integrador que encapsula y aplica conocimientos de diversas áreas de la tecnología y la ingeniería. La aplicación funciona como un asistente climático personal que, tras un proceso de autenticación de usuarios, permite realizar consultas de clima en tiempo real, almacenar un historial de búsquedas, analizar datos agregados y recibir consejos de vestimenta personalizados, generados por una Inteligencia Artificial.

El núcleo del proyecto reside en la demostración práctica y la interconexión de conceptos fundamentales de Programación, Ciberseguridad, Cloud Computing y Conectividad, Análisis de Datos e Inteligencia Artificial, materializados en una herramienta funcional y cohesiva.

## 1.2. Objetivos

Los objetivos primordiales que guiaron el desarrollo de este proyecto fueron:

- **Desarrollar una aplicación funcional:** Construir desde cero una aplicación de consola robusta, modular y fácil de usar en Python.
  - **Implementar conceptos de Ciberseguridad:** Simular un proceso de registro de usuarios con un fuerte énfasis educativo en la creación de contraseñas seguras, validando su fortaleza y guiando al usuario hacia prácticas más robustas.
  - **Integrar servicios en la nube (Cloud & Conectividad):** Consumir APIs externas (OpenWeatherMap y Google Gemini) para enriquecer la aplicación con datos y funcionalidades en tiempo real, demostrando la capacidad de conectividad con servicios de terceros.
  - **Aplicar Análisis de Datos:** Gestionar la persistencia de datos a través de archivos CSV y realizar cálculos agregados para extraer estadísticas significativas del uso global de la aplicación.
  - **Utilizar Inteligencia Artificial Generativa:** Diseñar e implementar una solución práctica que aproveche un modelo de lenguaje avanzado (LLM) para ofrecer valor añadido al usuario, como son los consejos personalizados.
  - **Elaborar documentación técnica y de usuario:** Crear documentación clara y concisa que describa la arquitectura, funcionamiento y uso de la aplicación, como el presente informe.
-

## 2. Diseño y Arquitectura

### 2.1. Estructura General de la Aplicación

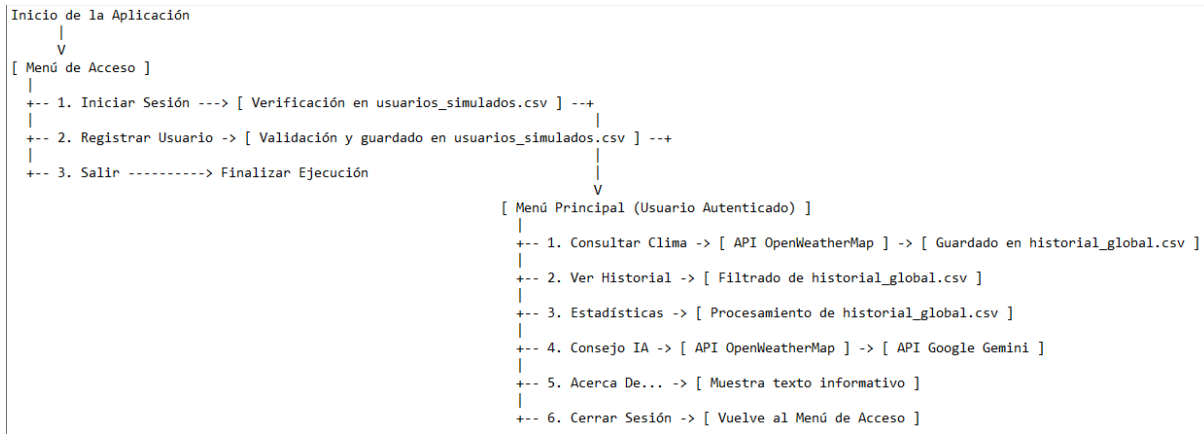
La arquitectura de "GuardiánClima ITBA" se basa en un diseño modular contenido dentro de un único script de Python. Se ha priorizado la claridad y la separación de responsabilidades a través de funciones bien definidas para cada tarea.

Los componentes principales del sistema son:

- 1. **Script Principal** ( Contiene toda la lógica de la aplicación, dividida en:
  - Funciones de gestión de usuarios (registro, login, validación).
  - Funciones de interacción con APIs (clima y IA).
  - Funciones de manipulación de archivos (lectura/escritura de CSV).
  - Funciones para la lógica de negocio (estadísticas, historiales).
  - Funciones para la interfaz de usuario (control de los menús).
- 2. **Almacenamiento de Datos (Archivos CSV):**
  - `usuarios_simulados.csv`: Actúa como una base de datos simulada para las credenciales de los usuarios. Contiene los campos `username` y `password_simulada`.
  - `historial_global.csv`: Almacena un registro de todas las consultas de clima realizadas por todos los usuarios, sirviendo como fuente para el análisis de datos y la visualización de historiales.

### 2.2. Flujo de Datos y Menús

El flujo de la aplicación está diseñado para guiar al usuario de manera intuitiva a través de sus funcionalidades. A continuación se presenta un diagrama de flujo.



### 2.3. Decisiones de Diseño

- **Lenguaje de Programación:** Se eligió **Python** por su sintaxis clara, su amplio ecosistema de librerías (especialmente `requests` y `google-generativeai`) y su idoneidad para el desarrollo rápido de prototipos y aplicaciones de consola.
  - **Persistencia de Datos:** Se optó por archivos **CSV** como método de almacenamiento para cumplir con los requisitos del proyecto y por su simplicidad. Esta elección permite una fácil inspección manual de los datos generados y es suficiente para el alcance de la aplicación.
  - **Modularidad:** El código se estructuró en funciones específicas para cada tarea. Esta decisión fue clave para facilitar el desarrollo, las pruebas y el mantenimiento del código, evitando la creación de un script monolítico y difícil de comprender.
-

## 3. Guía de Usuario Detallada

A continuación, se describe el proceso para utilizar cada una de las funcionalidades de "GuardiánClima ITBA".

### 3.1. Menú de Acceso

Al iniciar la aplicación, el usuario se encuentra con el primer menú, que controla el acceso al sistema.

- **Opción 1: Iniciar Sesión:** Al seleccionar esta opción, la aplicación solicitará el nombre de usuario y la contraseña. Si las credenciales son correctas, se concederá el acceso al Menú Principal.
- **Opción 2: Registrar Nuevo Usuario:** Se solicitará un nombre de usuario nuevo y una contraseña. La contraseña será validada según criterios de seguridad (longitud, mayúsculas, números). Si no los cumple, se informará al usuario para que la corrija.
- **Opción 3: Salir de la Aplicación:** Termina la ejecución del programa.

### 3.2. Menú Principal

Una vez autenticado, el usuario accede al núcleo funcional de la aplicación.

- **Opción 1: Consultar Clima Actual y Guardar:** Se pide una ciudad, se muestra el clima y se guarda la consulta.
  - **Opción 2: Ver Mi Historial Personal:** Muestra las consultas previas del usuario para una ciudad específica.
  - **Opción 3: Estadísticas Globales de Uso:** Muestra la ciudad más consultada, el número total de consultas y la temperatura promedio global.
  - **Opción 4: Consejo IA: ¿Cómo Me Visto Hoy?:** Genera una recomendación de vestimenta basada en el clima de una ciudad.
  - **Opción 5: Acerca De...:** Muestra información detallada sobre el proyecto.
  - **Opción 6: Cerrar Sesión:** Vuelve al Menú de Acceso.
-

## 4. Desafíos y Soluciones

Durante el ciclo de vida del proyecto, nuestro equipo se enfrentó a varios desafíos técnicos y conceptuales.

- **Desafío 1: Gestión Segura de Credenciales de API**
    - **Problema:** Riesgo de exponer las claves API en el código fuente.
    - **Solución:** Se aislaron las claves en un archivo aparte, `.env`.
  - **Desafío 2: Manejo de Errores en la Interacción con APIs Externas**
    - **Problema:** Las llamadas a servicios externos pueden fallar por múltiples razones.
    - **Solución:** Se implementó un manejo de errores robusto con bloques `try...except` para capturar excepciones específicas y proporcionar al usuario mensajes claros.
  - **Desafío 3: Diseño de un Prompt Efectivo para la IA (Prompt Engineering)**
    - **Problema:** Obtener respuestas útiles y consistentes de la API de Gemini.
    - **Solución:** Se diseñó un prompt específico que asigna un rol a la IA, proporciona datos estructurados y da instrucciones claras sobre el formato de la respuesta.
  - **Desafío 4: Mantener la Estructura y Legibilidad del Código**
    - **Problema:** Evitar que el código se volviera desorganizado a medida que crecía.
    - **Solución:** Se aplicó un diseño modular, descomponiendo la lógica en funciones pequeñas con una única responsabilidad, lo que mejoró la legibilidad y facilitó la depuración.
-

## 5. Conclusiones y Aprendizajes del Equipo

La realización del proyecto "GuardiánClima ITBA" ha sido una experiencia de aprendizaje sumamente enriquecedora que nos ha permitido materializar conceptos teóricos en una aplicación tangible.

Como equipo hemos adquirido los siguientes aprendizajes clave:

1. **Del Concepto a la Realidad:** Hemos transitado el ciclo completo de desarrollo de software, desde la conceptualización hasta la implementación y documentación.
2. **La Seguridad como Pilar Fundamental:** La simulación de validación de contraseñas nos hizo conscientes de la importancia crítica de la ciberseguridad desde el inicio del desarrollo.
3. **El Poder de la Interconectividad a través de APIs:** Hemos comprobado cómo las APIs extienden exponencialmente las capacidades de una aplicación, conectándola con servicios en la nube.
4. **Los Datos son el Nuevo Activo:** Vimos cómo un simple archivo `.csv` puede transformarse en una fuente de información valiosa para generar estadísticas y comprender patrones de uso.
5. **La Inteligencia Artificial como Herramienta Creativa:** La integración con Google Gemini nos demostró que la IA es una herramienta accesible y potente para crear funcionalidades innovadoras.

En conclusión, "GuardiánClima ITBA" no solo representa la culminación de un requisito académico, sino también un hito en nuestro desarrollo como futuros profesionales, fortaleciendo nuestras habilidades técnicas, nuestra capacidad para resolver problemas y nuestra aptitud para el trabajo en equipo.