

**UNIVERSIDAD NACIONAL DE COLOMBIA**

**FACULTAD DE ADMINISTRACIÓN**

**ADMINISTRACIÓN DE SISTEMAS INFORMÁTICOS**

**SISTEMAS INTELIGENTES**

**TALLER SISTEMAS INTELIGENTES**

**Desarrollo de una aplicación con Inteligencia Artificial Generativa para el cuidado de adultos mayores**

**DOCENTE: Nestor Dario Duque Mendez**

**MANIZALES, CALDAS, COLOMBIA**

**2025**

**DAVID ANDRES DIAZ GARCIA**

***Suki — Acompañante Virtual con Inteligencia Artificial***

***Link repositorio: <https://github.com/Davdiazga/suki-bot.git>***



## **Resumen del Proyecto**

Suki es una aplicación web que funciona como un acompañante virtual capaz de contar historias, compartir datos curiosos, frases motivadoras, chistes y mucho más. Está diseñada para brindar compañía y momentos agradables a los usuarios ( Personas de la tercera edad) mediante una interfaz amigable y colorida.

La aplicación utiliza el modelo Llama2 ejecutado localmente a través de Ollama, una API que permite generar texto con IA de forma rápida y privada, sin depender de servicios en la nube. El frontend está construido con React.js, lo que permite una experiencia de usuario fluida y dinámica.

### **Objetivo General**

Desarrollar un prototipo funcional de un acompañante virtual basado en inteligencia artificial capaz de generar contenido conversacional y entretenido para el usuario.

### **Objetivos Específicos**

* Diseñar una interfaz gráfica intuitiva y atractiva.
* Implementar la comunicación con un modelo de IA local usando Ollama y Llama2.
* Incorporar preguntas sugeridas para facilitar la interacción del usuario.
* Implementar modo claro/oscuro para mejorar la experiencia visual.
* Simular el efecto de escritura progresiva para dar mayor realismo a la interacción.

## **Descripción Técnica del Proyecto**

### **Arquitectura de la Aplicación**

La aplicación está compuesta por dos partes principales:

* **Frontend (React):**
  + Interfaz de usuario que permite enviar mensajes, ver respuestas y navegar por preguntas sugeridas.
  + Se comunica con la API local de Ollama usando fetch a http://localhost:11434/api/generate.
* **Backend (Ollama):**
  + Ejecuta el modelo **Llama2** de lenguaje natural localmente.
  + Procesa las solicitudes y devuelve el texto generado como respuesta.

**Para entender mejor:**

Usuario escribe mensaje → React envía petición HTTP a Ollama →

Ollama (Llama2) genera respuesta → React muestra texto

## **Tecnologías Utilizadas**

* **React.js** = Framework para construir la interfaz de usuario.
* **Ollama** = Plataforma para ejecutar modelos de IA localmente.
* **Modelo Llama2** = Modelo de lenguaje natural para generar texto.
* **HTML, CSS (inline styles)** = Para estructurar y diseñar los componentes visuales.
* **JavaScript (ES6)** = Lógica y estado de la aplicación.

## **Diseño de la Interfaz**

* **Secciones:**  
  + **Sección izquierda:** nombre de Suki, imagen de perfil y una lista de botones con preguntas sugeridas.
  + **Sección derecha:** ventana de chat, mensajes anteriores, caja de texto para escribir nuevos mensajes y boton para enviar audio que se convierte en texto.
* Soporte para modo claro y oscuro.
* Estilo visual basado en tonos pastel, suaves y amigables.
* Efecto de escritura progresiva cuando Suki responde para simular una conversación natural.

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

El contenido generado por IA puede ser incorrecto.

## **Funcionamiento**

1. El usuario abre la aplicación web en su navegador.
2. Puede elegir una pregunta sugerida, escribir su propio mensaje o enviar un audio que se convierte en texto.
3. El mensaje se envía al servidor local de Ollama.
4. Ollama procesa la solicitud con el modelo Llama2 y devuelve una respuesta.
5. La respuesta aparece en pantalla con un efecto de escritura progresiva.
6. El usuario puede continuar la conversación o cambiar al modo oscuro.

**Código Importante**



**Guía Rápida de Instalación**

1. Instalar **Node.js** y **npm**
2. Instalar **Ollama** en el sistema:  
    [https://ollama.com/download](https://ollama.com/download?utm_source=chatgpt.com)
3. Descargar el modelo Llama2: (Bash) “ollama pull llama2”
4. Clonar el repositorio: (Bash)

git clone <https://github.com/tu-usuario/suki-chat.git>

cd suki-chat

1. Instalar dependencias: (Bash) “npm install”
2. Ejecutar Ollama local: (Bash) “ollama serve”
3. Ejecutar la aplicación React: (Bash) “npm run dev”
4. Abrir en el navegador: http://localhost:

## **Mejoras Futuras**

* Guardar el historial de conversaciones en una base de datos.
* Permitir personalizar la apariencia del bot (nombre, avatar, colores).
* Incluir síntesis de voz para que el bot lea sus respuestas.
* Adaptar el diseño a dispositivos móviles.

## **Conclusiones**

El proyecto permitió aplicar conocimientos de desarrollo web con React y el uso de modelos de lenguaje ejecutados localmente mediante Ollama. Se consiguió un prototipo funcional que cumple su propósito: brindar compañía al usuario de manera creativa y entretenida, demostrando lo que puede hacer la inteligencia artificial con la interacción humana.