

# Entrega Final PROMPT

1<sup>st</sup> Diego Alejandro Gómez

*Ingeniería Electrónica*

*Ingeniería de PROMPT*

Bogotá, Colombia

diego.gomez-mo@mail.escuelaing.edu.co

***Index Terms*—ChatGPT, Ingeniería de Prompt, Inteligencia Artificial, Optimización de Respuestas, Optimización de Recursos, Sistema Operativo de Tiempo Real (RTOS)**

## I. INTRODUCCIÓN

En el actual panorama de la interacción humano-máquina, la Ingeniería de Prompt se erige como un campo de estudio esencial en inteligencia artificial y procesamiento del lenguaje natural. Este enfoque especializado busca perfeccionar las instrucciones emitidas a sistemas de inteligencia artificial para obtener respuestas precisas y contextualmente relevantes. En este contexto, este trabajo se adentra en uno de los retos más cruciales del ámbito de los sistemas operativos: el desarrollo de un Sistema Operativo de Tiempo Real (RTOS) altamente eficiente y confiable.

Los RTOS desempeñan un papel indispensable en aplicaciones que requieren respuestas inmediatas y predecibles, como sistemas embebidos y dispositivos médicos. No obstante, la creación de un RTOS que cumpla con estas exigencias implica un proceso complejo. Este estudio investiga cómo la Ingeniería de Prompt, combinada con la capacidad de ChatGPT, puede superar las limitaciones convencionales en el desarrollo de RTOS. Se centra en mejorar la comunicación entre desarrolladores y sistemas, utilizando técnicas avanzadas de optimización de prompt para facilitar una interacción más eficaz.

El propósito fundamental de esta investigación radica en analizar a fondo cómo estas estrategias de optimización pueden potenciar la eficiencia y adaptabilidad del RTOS. Se exploran los beneficios inherentes a esta metodología, así como sus desafíos y limitaciones. En la evaluación del impacto de este enfoque innovador en la industria, se busca identificar su potencial para catalizar futuras innovaciones en el ámbito de los sistemas operativos de tiempo real. Este estudio, al hacerlo, no solo contribuye a una comprensión más profunda de la Ingeniería de Prompt, sino que también establece las bases para el desarrollo de sistemas operativos más avanzados y eficientes en el futuro.

## II. CONCEPTOS BÁSICOS DE LA INGENIERÍA DE PROMPT

La Ingeniería de Prompt se ha convertido en un campo esencial en el desarrollo de sistemas de inteligencia artificial, transformando la forma en que los humanos interactúan con las máquinas. En su esencia, la Ingeniería de Prompt se centra

en optimizar las instrucciones dadas a los sistemas, asegurando que las respuestas generadas sean precisas y contextualmente relevantes. Para comprender este enfoque, es esencial tener en cuenta algunos conceptos básicos fundamentales:

1. **Optimización de Instrucciones:** La formulación de instrucciones claras y específicas es crucial para obtener respuestas precisas. La optimización se enfoca en refinar estas instrucciones para mejorar la calidad de las respuestas generadas.
2. **Análisis Semántico:** Implica entender el significado detrás de las instrucciones. Los sistemas deben ser capaces de interpretar el contexto semántico para generar respuestas adecuadas.
3. **Adaptabilidad Contextual:** Los sistemas deben ser flexibles y capaces de ajustar sus respuestas según el contexto de la conversación o la tarea en curso.
4. **Interacción Usuario-Máquina Mejorada:** La Ingeniería de Prompt busca mejorar la comunicación, facilitando interacciones más efectivas y naturales entre usuarios y sistemas de inteligencia artificial.
5. **Feedback Iterativo:** La retroalimentación del usuario es fundamental. Utilizar el feedback para ajustar futuras interacciones es esencial para perfeccionar las técnicas de ingeniería de prompt.

## III. CHATGPT: UNA HERRAMIENTA DE INTELIGENCIA ARTIFICIAL

En el panorama de la inteligencia artificial, ChatGPT se presenta como una innovadora herramienta que ha revolucionado la interacción humano-máquina. Basado en el sofisticado modelo GPT-3.5, ChatGPT se destaca por su capacidad para comprender y generar texto coherente y contextualmente relevante. Funciona mediante una arquitectura de red neuronal profunda, lo que le permite interpretar las instrucciones del usuario y generar respuestas adaptadas al contexto de la conversación. Esta versatilidad ha encontrado aplicaciones significativas en la ingeniería de prompt, donde facilita una comunicación precisa y eficaz entre desarrolladores y sistemas, optimizando las interacciones en proyectos como los Sistemas Operativos de Tiempo Real (RTOS). ChatGPT, con su visión general detallada y su complejo funcionamiento, ha establecido un estándar excepcional en la interacción humano-máquina, redefiniendo las posibilidades en el ámbito de la inteligencia artificial.

#### IV. DESARROLLO DEL SISTEMA OPERATIVO DE TIEMPO REAL (RTOS)

Los Sistemas Operativos de Tiempo Real (RTOS) han adquirido una importancia sustancial en aplicaciones que requieren respuestas inmediatas y predecibles a eventos específicos. En este contexto, se exploran los fundamentos de los Sistemas Operativos de Tiempo Real, donde la precisión del tiempo y la confiabilidad son críticas. Estos sistemas se diseñan para manejar tareas en tiempo real, desde controladores industriales hasta dispositivos médicos, garantizando una respuesta inmediata ante eventos específicos.

##### IV-A. *Fundamentos de los Sistemas Operativos de Tiempo Real*

Los RTOS se distinguen por su capacidad para manejar tareas con restricciones temporales estrictas, asegurando que las operaciones se realicen dentro de plazos determinados. Esto se logra mediante algoritmos de planificación especializados, como el algoritmo Rate Monotonic Scheduling, que asigna prioridades a las tareas basándose en sus tasas de ejecución. Además, la gestión de interrupciones, la sincronización y la comunicación entre tareas son esenciales para el funcionamiento óptimo de un RTOS.

##### IV-B. *Requisitos y Especificaciones del RTOS a Desarrollar*

Antes de la implementación, es fundamental definir los requisitos y especificaciones del RTOS. Esto implica determinar los límites de tiempo para las tareas, los protocolos de comunicación, la gestión de memoria y otros aspectos cruciales del sistema. Los requisitos deben ser claros y precisos para garantizar que el RTOS cumpla con las expectativas y funcione de manera fiable en el entorno previsto.

##### IV-C. *Implementación del RTOS Utilizando la Ingeniería de Prompt*

En el proceso de desarrollo del RTOS, la Ingeniería de Prompt desempeña un papel fundamental. Optimizar las instrucciones proporcionadas a través de técnicas de Ingeniería de Prompt permite una comunicación más efectiva entre los desarrolladores y el sistema. Esta optimización se traduce en la generación de respuestas precisas y relevantes, mejorando así la eficiencia del desarrollo y la adaptabilidad del RTOS. La claridad en las instrucciones facilita la implementación de algoritmos complejos y garantiza una integración sin problemas de componentes clave, lo que lleva a un RTOS altamente eficiente y confiable.

#### V. INGENIERÍA DE PROMPT Y OPTIMIZACIÓN DEL RTOS

En la sinergia entre la Ingeniería de Prompt y la optimización del Sistema Operativo de Tiempo Real (RTOS) se revela una nueva dimensión en la eficiencia y confiabilidad de los sistemas críticos. La meticulosa aplicación de técnicas de Ingeniería de Prompt ha supuesto un avance sustancial en la claridad y especificidad de las comunicaciones entre los desarrolladores y el sistema. Refinar las instrucciones mediante estas metodologías ha permitido una comprensión

precisa de los requisitos del sistema, lo que ha llevado a una implementación más eficaz de algoritmos complejos y a una gestión óptima de tareas en tiempo real. Esta optimización ha generado respuestas más coherentes y ha fortalecido la capacidad del RTOS para adaptarse a diversas condiciones operativas.

Los resultados obtenidos son notables al compararlos con los sistemas tradicionales. Los tiempos de respuesta han experimentado mejoras significativas, junto con una precisión incrementada en la ejecución de tareas críticas. La mejora en la comunicación entre desarrolladores y sistema se traduce en una reducción marcada de errores y una mayor confiabilidad del RTOS en entornos desafiantes. La capacidad del sistema para adaptarse dinámicamente a situaciones imprevistas ha marcado la diferencia en su rendimiento global, estableciendo un nuevo estándar en eficiencia y flexibilidad operativa.

No obstante, este progreso no ha estado exento de desafíos. La complejidad inherente a ciertos escenarios de uso y las variaciones en las instrucciones presentaron obstáculos notables. Estos desafíos se han afrontado mediante la implementación de sistemas de retroalimentación iterativos y la continua optimización de las técnicas de Ingeniería de Prompt. Estos procesos dinámicos de retroalimentación han permitido ajustar las instrucciones de manera proactiva, superando los desafíos y mejorando constantemente la eficacia del RTOS.

#### VI. USO DE CHAT GPT EN EL DESARROLLO DEL PROYECTO

A lo largo de la realización del proyecto la influencia de Chat Gpt fue bastante relevante, pues se usó desde un inicio para obtener descripciones generales de los sub módulos del RTOS, ya que al ser un sistema que jamás se había implementado a lo largo del pregrado, se contaba con un conocimiento casi nulo sobre dicha asignatura. El Gpt en su mayoría fue usado para la corrección de errores, dado que es complejo que proporcione todo el código desde cero sin tener falencias en su creación. Adicional a ello, el diseño de un RTOS desde el inicio es un tipo de información compleja para ser encontrada en internet, por tanto la mejor opción no es dejar todo el trabajo a una máquina, es totalmente necesario desarrollar código por cuenta propia. Como solucionador de errores, principalmente de sintaxis Chat gpt posee grandes habilidades, que sorprenden en un primer instante, y este entrenamiento que le fue dado hace inferir que existe una fuente vasta de información para que el modelo ejecute su trabajo tan bien. Claramente el desarrollo de código en C, es algo que lleva años siendo trabajado y profundizado cada vez más, debido a que día a día nuevas herramientas surgen en esta área del conocimiento. En los anexos del proyecto hallados en GitHub se puede encontrar el chat completo que se tuvo con el modelo, y se puede observar detalladamente el trabajo que se realizó, dejando en claro que esta fantástica herramienta requiere un pre entrenamiento para su correcto funcionamiento. De igual manera es posible hallar el código en C en su totalidad funcionando tal como fue mostrado en clase al docente.

## VII. CONCLUSIONES

1. Optimización de la Comunicación Efectiva: La aplicación de técnicas avanzadas de Ingeniería de Prompt ha optimizado la comunicación entre desarrolladores y el Sistema Operativo de Tiempo Real (RTOS), mejorando la precisión y claridad en las instrucciones. Esto ha resultado en una ejecución más eficiente de tareas críticas y una reducción significativa de errores.
2. Mejora Sustancial en el Rendimiento del RTOS: La implementación de Ingeniería de Prompt ha conducido a una mejora notable en los tiempos de respuesta del RTOS y su capacidad para adaptarse a situaciones imprevistas. Esto ha establecido nuevos estándares en eficiencia y flexibilidad operativa, marcando una diferencia significativa en comparación con sistemas tradicionales.
3. Estándares Elevados en Interacción Humano-Máquina: La sinergia entre Ingeniería de Prompt y la optimización del RTOS ha establecido estándares sobresalientes en la interacción humano-máquina en entornos críticos. Estos avances han allanado el camino para futuras innovaciones, creando un nuevo paradigma en sistemas embebidos y el control industrial, elevando así la calidad y confiabilidad en estos entornos cruciales.