

Desarrollo de un Simulador Web Interactivo Para la Reconstrucción Gráfica de Funciones Periódicas y Señales Usando Análisis de Fourier

Development of an Interactive Web Simulator for Graphical Reconstruction of Periodic Functions and Signals Using Fourier Analysis

Pérez G. Marco A.*, Pérez S. Adolfo A., Duarte G. Cristian D., J. Betancourt Ramón O., González L. Juan M., Rodríguez H. Fernando., Navarro A. Ernesto.

Universidad de Colima, Facultad de Ingeniería Electromecánica. Km 20. Carr. Manzanillo – Cihuatlán, El Naranjo, Colima, México, 28800.

Autor para la correspondencia: Marco Antonio Pérez González, marcoperez@ucol.mx

Resumen

El fundamento matemático del análisis de señales eléctricas es la serie de Fourier. Este modelo presenta una carga analítica significativa que constituye un reto para estudiantes de primeros años en carreras de ingeniería y ciencias exactas. Por esta razón, este trabajo presenta una herramienta interactiva desarrollada para apoyar el aprendizaje práctico de estos conceptos. Se describe el desarrollo de un simulador web implementado mediante una metodología ágil que incorpora elementos de los marcos de gestión de proyectos de software Scrum y Kanban. El simulador fue programado usando Python y el framework web Django. Esta herramienta permite realizar análisis gráficos en dos modalidades: la primera consiste en la creación y reconstrucción analítica de señales a partir de parámetros en el dominio del tiempo; la segunda realiza la reconstrucción partiendo del dominio de la frecuencia. Se muestran los resultados técnicos obtenidos durante la implementación y validación del software, así como las observaciones recopiladas en las pruebas con usuarios reales. Dichos resultados indican que el simulador facilita la comprensión intuitiva del análisis espectral y puede ser un apoyo efectivo en el estudio de señales y sistemas al sintetizar gráficamente la complejidad matemática del modelo de Fourier.

Palabras clave: simulador interactivo, análisis de Fourier, python, Django.

Abstract

The mathematical foundation of electric signal analysis is the Fourier Series. This approach involves significant analytical complexity, becoming a real challenge for first- and second-year students in engineering and science programs. For this reason, this work presents an interactive tool designed to support practical learning of these concepts. The development of this simulator is described, highlighting the agile methodology used, which incorporates elements from Scrum and Kanban frameworks. Python and the Django web framework were employed to implement the proposed simulator, which enables graphical analysis from two perspectives: reconstruction of signals from parameters defined in the time domain and from parameters in the frequency domain. Technical results obtained during simulation trials and validation stages are presented, evaluating both technical performance and user experience. These results suggest that the simulator facilitates intuitive comprehension of spectral analysis and has the potential to become a valuable resource for signal and systems education by graphically synthesizing the inherent mathematical complexity of Fourier analysis.

Key words: interactive simulator, Fourier analysis, python, Django.

DOI: 10.46588/invurnus.v20i1.125

Recibido 20/07/2025

Aceptado 25/07/2025

Publicado 10/08/2025

Introducción

El análisis de sistemas lineales, y en particular el análisis de funciones periódicas forma parte del currículo de la mayoría de los programas en ingeniería, especialmente en las áreas eléctrica, electrónica, mecánica y computación. Se trata, por tanto, de un fundamento crucial cuya enseñanza representa un reto constante tanto para el profesor como para el estudiante. La complejidad matemática del tema y las habilidades analíticas que, históricamente, el estudiante mexicano no logra desarrollar con éxito, constituyen el aspecto más visible del problema (OCDE, 2023).

El desarrollo matemático exhaustivo, sumado al grado de complejidad de las integrales propias del método, genera con frecuencia un alto nivel de estrés en el estudiante promedio de ingeniería. Esto se traduce en promedios de calificación en estas asignaturas que suelen estar por debajo del resto del plan de estudios (González-Quñones et al., 2022).

La búsqueda de soluciones al problema expuesto ha sido ampliamente documentada en la literatura especializada. Pepin et al. (2021) realizan una revisión exhaustiva sobre prácticas innovadoras en la enseñanza de las matemáticas en ingeniería, destacando enfoques activos, interdisciplinarios, el uso de recursos digitales y el modelado matemático. También analizan sus implicaciones en la reforma curricular y el desarrollo de competencias.

En este contexto —aunque no se pretende centrarse en aspectos pedagógicos o didácticos—, esta propuesta se inscribe dentro del enfoque de Recursos Educativos Digitales (REDs), uno de los más recurridos al momento de enfrentar los retos de la enseñanza de las matemáticas en el nivel superior.

El trabajo de Pepin et al. (2017) presenta una revisión conceptual del uso y diseño de recursos curriculares digitales. Entre sus hallazgos más relevantes destaca la noción de que el diseño de estos recursos continúa durante su uso, en un ciclo de mejora continua. Esto amplía el espacio de interacción y favorece la personalización del aprendizaje al migrar de recursos estáticos a dinámicos.

En el caso específico de los simuladores web orientados al análisis de Fourier, es posible identificar una amplia gama de propuestas con características muy variables: desde desarrollos artesanales con objetivos específicos, hasta plataformas altamente especializadas, tanto en lo técnico como en su potencial didáctico. La calculadora de Symbolab (s. f.) es un ejemplo tradicional: permite obtener la serie y transformada de Fourier de cualquier función ingresada y ofrece soluciones paso a paso. Sin embargo, su componente gráfico no es inmediato ni integrado, ya que requiere módulos adicionales, lo cual limita su interactividad.

En el otro extremo se encuentra el applet de Falstad (s. f.), centrado en la visualización gráfica sin exponer los cálculos matemáticos ni las expresiones analíticas. Aunque presenta limitaciones en cuanto a interfaz y resolución gráfica debido a su naturaleza computacional, ofrece una experiencia interactiva aceptable. El usuario puede modificar múltiples parámetros, incluyendo la incorporación de ruido eléctrico, lo cual es especialmente útil en el contexto de la ingeniería práctica. No obstante, al no incluir soluciones analíticas, el estudiante no cuenta con un puente claro entre teoría y práctica.

La Universidad de Colorado Boulder, a través del proyecto PhET Interactive Simulations (2007), ha desarrollado una serie de recursos orientados al fortalecimiento de la educación STEM. Su simulador “Fourier: Making Waves” representa un avance en términos de usabilidad y diseño, permitiendo explorar visualmente cómo se combinan diferentes armónicos. A pesar de ello, al igual que el applet de Falstad, no profundiza en el aspecto analítico ni ofrece expresiones matemáticas asociadas a las gráficas manipuladas.

Fourier Solver (s. f.) es una plataforma que integra varias de las funcionalidades previamente descritas. Sus resultados gráficos son impactantes y están respaldados por una formalización matemática sólida. No obstante, la reconstrucción de señales está limitada al dominio del tiempo. Además, al momento de esta



revisión, el proyecto parece encontrarse en desarrollo, como lo indica la presencia de encuestas de retroalimentación y algunos errores detectados en casos particulares.

Dado este panorama, la propuesta presentada en este trabajo es pertinente. Nuestra investigación previa indica que existen pocos recursos digitales que, en el contexto del análisis de Fourier, integren simultáneamente los aspectos analítico, gráfico, interactivo y la representación tanto en el dominio del tiempo como en el de la frecuencia.

Fundamento matemático

De acuerdo con su trabajo seminal, *Théorie analytique de la chaleur*, Fourier, (1822), demostró que cualquier función periódica $f(t)$ con periodo T , para la que:

$$f(t) = f(t + nT) \quad n = 1, 2, \dots \quad (1)$$

puede expresarse como una combinación de funciones trigonométricas, denominada serie de Fourier y descrita por la ecuación (2).

$$f(t) = \frac{a_0}{2} + \sum_{n=1}^{\infty} \{a_n \cos n\omega_0 t + b_n \sen n\omega_0 t\} \quad (2)$$

Donde $\omega_0 = 2\pi/T$ es la frecuencia fundamental y a_0 , a_n y b_n se denominan componentes cero, par e impar y se definen como:

$$a_0 = \frac{2}{T} \int_0^T f(t) dt \quad (3)$$

$$a_n = \frac{2}{T} \int_0^T f(t) \cos n\omega_0 t dt \quad (4)$$

$$b_n = \frac{2}{T} \int_0^T f(t) \sen n\omega_0 t dt \quad (5)$$

La figura 1 muestra esquemáticamente como una señal periódica cuadrada puede reconstruirse de manera aproximada con los primeros 11 armónicos, es decir $n = 1, 2, \dots, 11$ en la ecuación (2). Cada armónico ($n > 1$) corresponde una función senoidal desfasada respecto de la componente fundamental ($n=1$) cuya amplitud disminuye y frecuencia aumenta a medida que n se incrementa, tal como se deduce de las expresiones (2) a (5).

Esta relación se representa visualmente al avanzar en la profundidad de la gráfica: cada onda senoidal en color gris representa uno de los armónicos, comenzando por el fundamental. La gráfica en color rojo, por su parte corresponde a la reconstrucción parcial de la señal periódica —es decir, la suma de los armónicos considerados—, la cual se aproxima progresivamente a una onda cuadrada.

Evidentemente conforme el número de armónicos tiende al infinito, la señal reconstruida (en rojo) se aproxima aún más a una onda cuadrada ideal, eliminando el rizado visible y convirtiendo los “bordes” en líneas rectas casi perfectas.



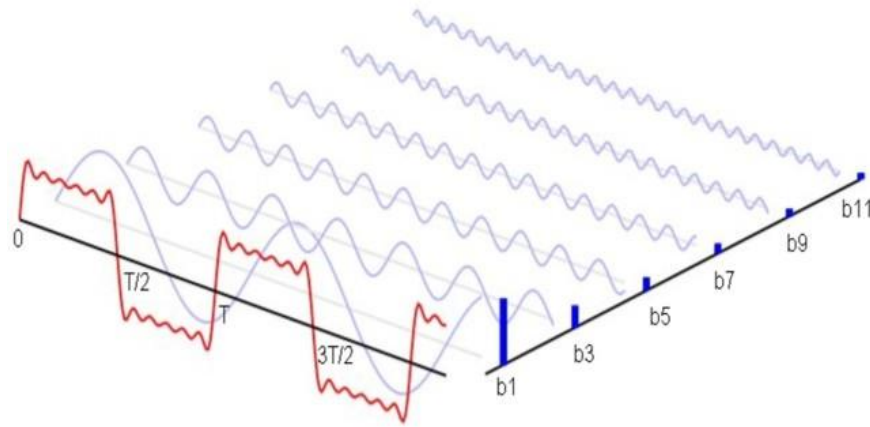


Figura 1. Aproximación de señal cuadrada con número limitado de armónicos, basado en (Lifeder, s. f.).

Cada señal eléctrica que pueda ser caracterizada por una función que cumple con la condición de la expresión (1) puede representarse mediante la ecuación (2).

A continuación, se describen dos aspectos teóricos más que son puntos importantes en el modelado matemático usado en el simulador.

Primero, la caracterización de una función periódica no sólo permite su reconstrucción, además el análisis de Fourier—también conocido como análisis espectral— posibilita la descomposición de la señal en componentes frecuenciales, cada una con una magnitud y un ángulo de fase específico.

Para ilustrar este punto, considere otra vez la ecuación (2), mediante el uso de la identidad de Euler, la serie de Fourier puede reescribirse en una forma más compacta como:

$$f(t) = \frac{a_0}{2} + \sum_{n=1}^{\infty} A_n \cos(n\omega_0 t + \phi_n) \quad (6)$$

Donde A_n y ϕ_n representan, respectivamente, la magnitud y fase del armónico n , las cuales se calculan como:

$$A_n = \sqrt{a_n^2 + b_n^2} \quad (7)$$

$$\phi_n = -\tan^{-1}\left(\frac{b_n}{a_n}\right) \quad (8)$$

La ecuación (6) incorpora el análisis espectral antes mencionado, lo que permite comprender con mayor claridad los denominados espectros de frecuencia, los cuales se dividen en espectros de magnitud y de fase. La figura 2, muestra una descomposición espectral y reconstrucción aproximada de una señal periódica, en este caso una función del tipo “sawtooth” (diente de sierra). En la figura se aprecia la naturaleza discreta de los espectros de frecuencia; cada armónico se corresponde con un múltiplo entero de la frecuencia, lo que conduce a la representación discreta usando líneas verticales en frecuencias específicas, sin puntos intermedios entre cada múltiplo.

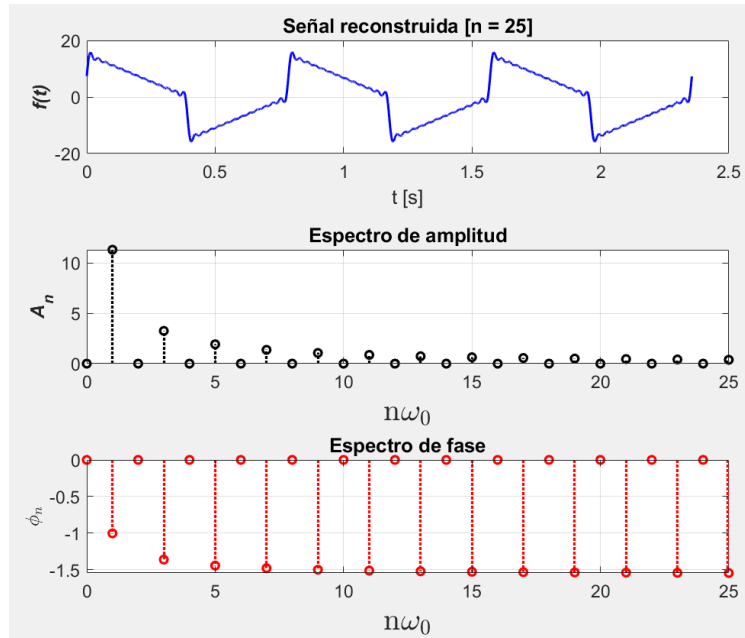


Figura 2. Reconstrucción y análisis espectral de una función periódica. *Elaboración propia.*

El abordaje tradicional consiste en resolver múltiples ejemplos hasta que el alumno logre un dominio analítico del proceso. Existe una amplia cantidad de desarrollo teórico disponible en la literatura especializada, al punto de que hoy en día pueden encontrarse formulas directas para calcular la serie de Fourier.

A manera de ilustración del tipo de señales que puede procesar el simulador presentado en este trabajo, se considera una señal de pulsos generalizada como se muestra en la figura 3. Se trata de una señal de n pulsos, en la cual el pulso k , tiene una amplitud A_k y un ancho o duración igual a $t_k - t_{k-1}$, para $k = 1, 2, \dots, m$.

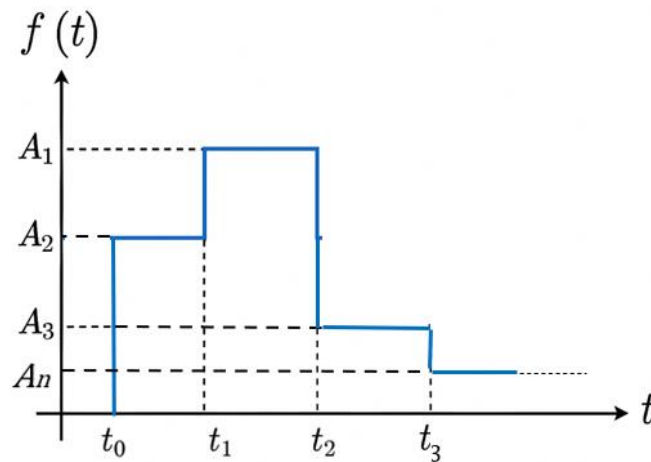


Figura 3. Señal generalizada de n pulsos. *Elaboración propia.*

Ya que el enfoque del simulador no es utilizar una solución numérica, sino que tiene por objetivo recuperar la solución analítica de los problemas, a fin de establecer una conexión entre la teoría y la programación de manera accesible para el estudiante; se desarrolla la serie de Fourier para la función generalizada anterior.



Debido a la complejidad de la expresión resultante, se presentan por separado los componentes a_0 , a_n y b_n de la ecuación (2), como se muestra a continuación:

$$a_0 = \frac{2}{T} \sum_{n=1}^k A_k (t_k - t_{k-1}) \quad (9)$$

$$a_n = \frac{1}{\pi} \sum_{n=1}^k \frac{A_k}{n} (\sin n\omega_0 t_k - \sin n\omega_0 t_{k-1}) \quad (10)$$

$$b_n = -\frac{1}{\pi} \sum_{n=1}^k \frac{A_k}{n} (\cos n\omega_0 t_k - \cos n\omega_0 t_{k-1}) \quad (11)$$

Con estos elementos se establece el marco matemático que da soporte a la herramienta desarrollada, permitiendo ahora avanzar hacia su implementación computacional en un entorno web interactivo.

Materiales y métodos

Plataforma y lenguajes de desarrollo

El simulador está programado en lenguaje Python (Van Rossum y Drake, 2009), un lenguaje de uso estándar en aplicaciones científicas y de ingeniería, que lo hace ideal para gestionar la lógica del simulador. Del conjunto de librerías de este lenguaje, se usó *NumPy* por su capacidad para de realizar cálculos con arreglos multidimensionales (Harris et al., 2020); y *Matplotlib*, ya que permite graficar diferentes tipos de datos en formatos y especificaciones de forma y color muy variadas (Hunter, 2007).

La implementación web, se hizo a través de *Django*, un framework de desarrollo web de alto nivel para *Python* para gestionar la lógica del lado del servidor, el flujo de datos y la presentación dinámica de contenidos. Django permite además una integración clara con otras tecnologías del lado cliente, lo que facilitó el diseño modular del proyecto. (The Django Software Foundation, s. f.).

Para lograr una interacción visual fluida, se incorporó *p5.js*, una biblioteca de JavaScript orientada a la creación de gráficos interactivos en el navegador (McCarthy, 2024). *Bootstrap* fue utilizado para garantizar que la interfaz fuera limpia, funcional y adaptativa, permitiendo una experiencia de usuario consistente en distintos dispositivos (Bootstrap, s. f.).

El proyecto se desarrolló en el entorno *visual studio code*, una herramienta ligera pero robusta que facilitó la escritura, organización y depuración del código (Microsoft, s.f.). Para el control de versiones y la gestión del proyecto en fases iterativas, se utilizó *GitHub*, lo cual permitió un seguimiento claro del progreso, así como la documentación de cada modificación durante el desarrollo (GitHub, s. f.).

En conjunto, este stack tecnológico fue seleccionado no solo por su potencia técnica, sino también por su adecuación al entorno educativo, buscando un balance entre precisión matemática, visualización efectiva y facilidad de uso por parte del estudiante.

La figura 4 muestra la forma en cómo se integraron las diferentes plataformas y software para el desarrollo del proyecto de software del simulador.



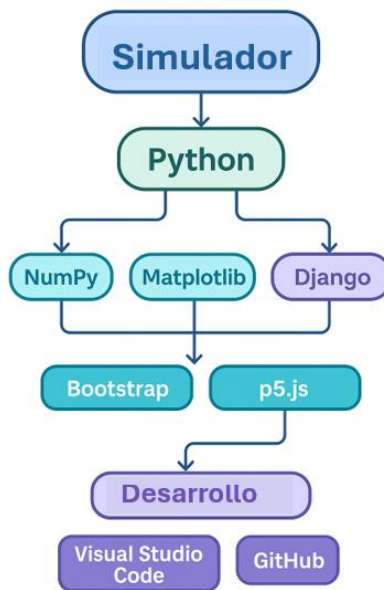


Figura 4. Integración de lenguajes y plataformas para la obtención del simulador. *Elaboración propia.*

Metodología de desarrollo de software

El desarrollo del simulador se llevó a cabo bajo un enfoque ágil, tomando elementos tanto de la metodología *Scrum* como de *Kanban*, lo cual permitió estructurar las tareas de programación y diseño de forma flexible y adaptativa.

Del enfoque *Scrum* se retomó la organización del trabajo por entregas parciales (Schwaber & Sutherland, 2020). Por su parte, el uso de tableros tipo *Kanban* permitió una visualización clara de los avances, facilitando la gestión de tareas pendientes, en proceso y finalizadas (Anderson, 2010).

Durante el proceso se priorizó la funcionalidad esencial del simulador, permitiendo que desde etapas tempranas se tuviera un prototipo utilizable. A lo largo de la implementación, el sistema fue probado de forma continua por usuarios voluntarios, lo que permitió obtener retroalimentación directa y mejorar tanto la interfaz como la lógica de procesamiento.

Arquitectura general del sistema

El simulador fue desarrollado bajo una arquitectura cliente-servidor. El cliente es el navegador del usuario, desde donde se accede a la interfaz visual e interactiva; el servidor procesa los datos, ejecutar los cálculos y generar los resultados que luego son enviados de vuelta al cliente para su visualización.

Cuando el usuario accede al simulador, puede interactuar con distintos parámetros relacionados con funciones periódicas: amplitudes, tiempos de inicio y fin, número de armónicos, tipo de función base, entre otros. Estas entradas se registran en el cliente, se envían al servidor, y tras el procesamiento correspondiente —que incluye el cálculo de los coeficientes de Fourier y la reconstrucción de la señal— se presentan los resultados en forma de gráficas dinámicas.



Funcionalidades del simulador

La aplicación está organizada en dos módulos principales, que se corresponden con los enfoques clásicos del análisis de Fourier: el dominio del tiempo y el dominio de la frecuencia.

Para acceder al simulador se requiere una cuenta de usuario, lo cual permite llevar un registro de su utilización con fines estadísticos. Una vez autenticado, el usuario es dirigido a la pantalla de inicio o *home* en donde a manera de panel informativo y en cajas de texto llamativas se presentan definiciones básicas relativas a señales periódicas y series de Fourier, la figura 5 muestra esta pantalla.



Figura 5. Módulo de inicio del simulador mostrando conceptos breves y básicos de series de Fourier.
Elaboración propia.

Desde el inicio se tiene acceso a los dos módulos principales. El primero corresponde al análisis en el dominio del tiempo. En este, se puede definir una señal periódica, especificando manualmente las amplitudes y los intervalos de duración de cada pulso. El sistema genera una reconstrucción visual de la señal compuesta a partir de estos datos (figura 6).

Esta figura muestra el menú de la simulación en el tiempo, el cual despliega los campos en los que el usuario ingresa los valores numéricos dependiendo de la señal de pulsos que pretenda reconstruir.

MathPou

Inicio

Gráficas de Ondas

Datos para gráfica

Ejemplos

DATOS PARA GRÁFICA

Amplitud:

Tiempo:

Crear

Generar gráfica

Limpiar

FÓRMULA PARA a_n :

$$a_n = \frac{1}{n\pi} \sum_{k=1}^N A[k] (\sin(n\omega_0 t_k) - \sin(n\omega_0 t_{k-1}))$$

FÓRMULA PARA b_n :

$$b_n = -\frac{1}{n\pi} \sum_{k=1}^N A[k] (\cos(n\omega_0 t_k) - \cos(n\omega_0 t_{k-1}))$$

Figura 6. Menú para la simulación en el dominio del tiempo. *Elaboración propia.*

La figura 7 muestra el menú correspondiente al análisis en el dominio de la frecuencia. Se observa que existe un menú gráfico en el que el usuario ingresa los valores de la amplitud y la fase seleccionado el número de armónico correspondiente y arrastrando hasta el valor numérico que desee. En la parte inferior hay un área para graficar en donde se reconstruye la señal a partir de estos parámetros. Del lado derecho existen datos precargados que permiten la reconstrucción de algunas señales típicas, dando así al alumno la oportunidad de verificar sus respuestas a ejercicios específicos dada las fórmulas directas.

MathPou

Inicio

Gráficas de Ondas

Datos para gráfica

Ejemplos

GRÁFICAS DE ONDAS Y SUMA

Amplitudes (A)

A1 A2 A3 A4 A5 A6 A7 A8 A9 A10 A11

Fases (F)

F1 F2 F3 F4 F5 F6 F7 F8 F9 F10 F11

Amplitud

Tiempo

Amplitud

Ejemplo num 1

Ver

Señal Triangular

Ver

Señal Cuadrada

Ver

Señal Sierra

Ver

Cerrar Sesión

Figura 7. Menú de análisis en el dominio de la frecuencia. *Elaboración propia.*



Este conjunto de funcionalidades no sólo refuerza los conceptos teóricos relacionados con la serie de Fourier, sino que también promueve un aprendizaje activo y experimental. Al combinar manipulación directa de parámetros con visualización inmediata, el simulador busca acortar la distancia entre el modelo matemático abstracto y su manifestación tangible en el análisis de señales.

Resultados y Discusión

El simulador entrega resultados que, en términos de exactitud, son técnicamente precisos y desde el punto de vista didáctico revelan su potencial para ser usado como una herramienta de apoyo tanto para profesores y estudiantes en cursos que abordan el análisis de Fourier.

Para evaluar su funcionalidad y destacar las características más relevantes de las capacidades del simulador se presentarán tres casos de usos que ilustran su comportamiento en situaciones representativas.

- Reconstrucción de una señal de tres pulsos en el dominio del tiempo
- Reconstrucción de una señal triangular utilizando datos precargados en la sección de ejemplos del módulo del dominio de la frecuencia
- Reconstrucción de una señal en el dominio de la frecuencia, introduciendo los parámetros derivados de un problema resuelto disponible en la literatura especializada.

En la primera prueba se ingresan los datos correspondientes a la función: $f(t) = \begin{cases} 2, & 0 < t < 2 \\ -4, & 2 < t < 6 \\ 5, & 6 < t < 7 \end{cases}$, la figura 8

muestra la plantilla de ingreso de los datos.

DATOS PARA GRÁFICA

Amplitud:	Tiempo:
2	2
Amplitud 2:	Tiempo 2:
-4	6
Amplitud 3:	Tiempo 3:
5	7

Figura 8. Plantilla para el ingreso de parámetros de simulación para una función de pulsos cuadrados.
Elaboración propia.

La figura 9 muestra la reconstrucción grafica de la función periódica en el intervalo $[0, 2T]$ para 100 armónicos. El resultado muestra una reconstrucción casi perfecta, con el rizado y ruido excesivo en las fronteras de los intervalos, fenómeno de Gibbs (Jerri, 1998), característicos de este enfoque cuando se trabaja con funciones cuadradas.



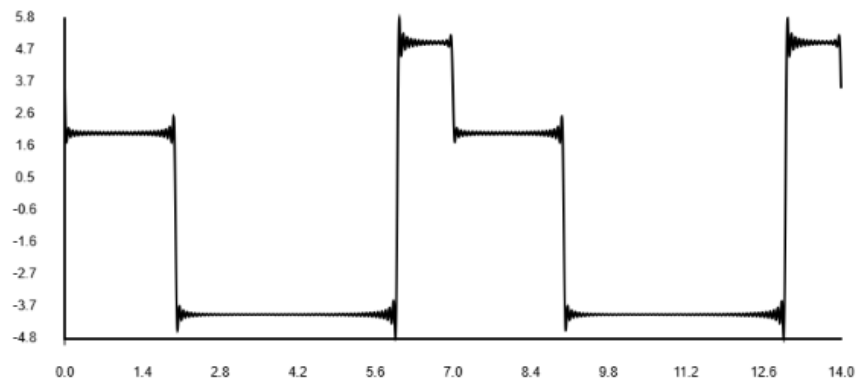


Figura 9. Reconstrucción gráfica obtenida para la simulación de una función de tres pulsos cuadrados. *Elaboración propia.*

Para la segunda simulación se accede directamente a la sección de ejemplos del módulo de análisis en el dominio de la frecuencia, seleccionando el botón “señal triangular”. La figura 10 muestra los resultados. A partir de estos resultados se observa que, aunque la forma triangular general de la onda se hace evidente, la aproximación dista de ser perfecta al considerar únicamente 11 armónicos (limitación impuesta por razones de espacio). Es importante notar cómo algunos armónicos aparecen con valor cero; esto es común en señales con ciertos tipos de simetría, y se refleja claramente en los controles deslizantes del simulador, los cuales no contienen ningún valor asignado en estos casos.

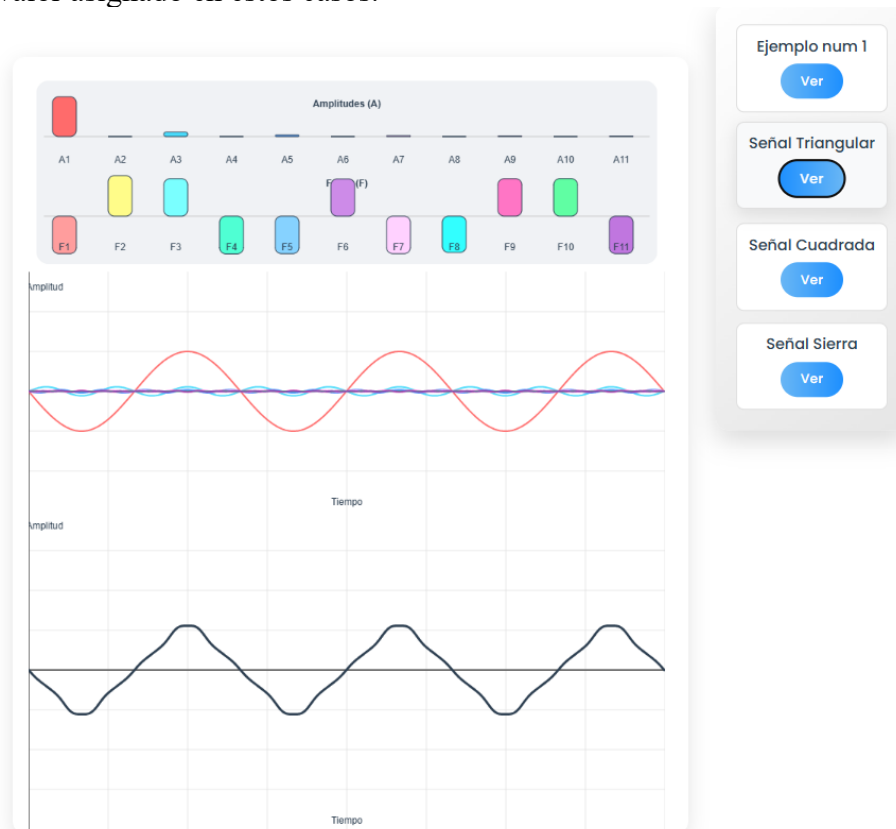


Figura 10. Señal triangular reconstruida para 11 armónicos. *Elaboración propia.*



Además, esta sección permite graficar individualmente cada uno de los armónicos no nulos que componen la señal, ofreciendo al usuario una representación visual clara de su contribución específica. Una de las características interactivas más relevantes del simulador es que permite observar en tiempo real cómo varían tanto la amplitud como la forma de la señal reconstruida al manipular los controles deslizantes. Esta funcionalidad otorga al estudiante una herramienta poderosa para visualizar el impacto que tienen los múltiplos de frecuencia de la componente fundamental sobre la señal compuesta, promoviendo así una comprensión más profunda y dinámica del análisis de Fourier.

Para la tercera simulación, se retomaron los datos de un ejercicio resuelto disponible en el ejemplo 17.2, página 767 de (Alexander, 2013). Estos valores se reproducen en la tabla 1 y su representación gráfica se muestra en la figura 11. Es importante señalar que, en el libro, los ángulos de fase están expresados en grados sexagesimales, mientras que el simulador requiere que se ingresen en radianes. La señal periódica utilizada corresponde a un caso particular de señal triangular.

Tabla 1. Datos de amplitud y fase para la reconstrucción. *Basado en (Alexander, 2013).*

n	A_n	$\phi_n[\text{rad}]$
0	0.2500	0
1	0.3773	4.1455
2	0.1670	4.442
2	0.1085	4.5033
4	0.0806	4.5546
5	0.0642	4.5857
6	0.0533	4.6067

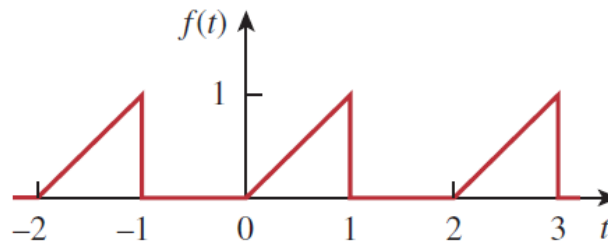


Figura 11. Señal para reconstruir en el tercer caso de simulación. *Tomado de (Alexander, 2013).*

La simulación se muestra en la figura 12. La forma de la señal es reconocible y un resultado típico para una aproximación con tan solo 6 armónicos. La reconstrucción gráfica evidentemente mejora si se incluyen más componentes.

Conviene destacar que el diseño del simulador, al emplear controles deslizantes para ingresar los valores de amplitud y fase, impone una restricción no menor en la precisión de los datos. En esta versión, los valores mínimos de paso disponibles en los controles son 0.0250 y 0.0500 —para la magnitud y el ángulo de fase, respectivamente— lo cual puede parecer poco significativo. Sin embargo, en casos como este —donde tanto la amplitud como la fase tienen variaciones pequeñas pero sensibles— estas restricciones afectan la posibilidad de replicar con precisión resultados específicos. Aunque esta limitante no compromete el objetivo pedagógico general del simulador, sí puede influir en la obtención de resultados más ajustados, especialmente cuando se busca una comparación estricta con datos de referencia.





Figura 12. Simulación del caso tres en el dominio de la frecuencia. *Elaboración propia.*

Como parte del proceso de validación, se realizaron pruebas con la versión actual del simulador. Para ello, se invitó a 20 estudiantes de la carrera de Ingeniería en Tecnologías Electrónicas, pertenecientes a distintos semestres, que estuvieran inscritos en alguna asignatura relacionada con el análisis de funciones periódicas, o que ya hubiesen cursado dicha temática. La selección de la muestra fue aleatoria.

La evaluación se llevó a cabo mediante una encuesta basada en la escala de Likert, transformada posteriormente a una escala porcentual del 0 al 100 %, donde 0 % representa una experiencia completamente insatisfactoria y 100 %, completamente satisfactoria.

Se evaluaron los siguientes aspectos:

1. Usabilidad: Facilidad para navegar entre los menús, ingresar datos, interpretar resultados y manipular los controles deslizantes.
2. Precisión: Concordancia entre los cálculos obtenidos y los datos de ejercicios resueltos en el libro, así como la exactitud de las gráficas generadas.
3. Interactividad: Capacidad de respuesta del simulador ante la manipulación de los parámetros, claridad en las instrucciones de llenado y facilidad para modificar componentes en tiempo real.

Los comentarios abiertos fueron agrupados y estandarizados en frases representativas para su análisis. Los resultados se resumen en la tabla 2.



Los datos obtenidos son alentadores para esta primera versión del simulador. Reflejan una aceptación positiva por parte de los usuarios y constituyen una base útil para evaluar críticamente el diseño actual, lo que permitirá definir acciones concretas para mejorar la aplicación en el corto plazo.

Tabla 2. Resultados de prueba de usuario. *Elaboración propia.*

Criterio	Porcentaje de satisfacción	Comentarios
Usabilidad	85 %	"Fácil de usar, pero los menús podrían ser más claros."
Precisión	90 %	"Los gráficos coinciden con los cálculos teóricos."
Interactividad	85 %	"La modificación en tiempo real es útil."

Conclusiones

El uso REDs para la enseñanza de las matemáticas es una herramienta ampliamente utilizada por docentes que imparten esta asignatura. Se ha documentado que su implementación promueve el análisis cualitativo por parte del estudiante, al disponer de más tiempo para reflexionar sobre los resultados obtenidos, ya sea mediante representaciones gráficas interactivas o estáticas. Esta primera versión del simulador ha demostrado tener un potencial significativo como apoyo didáctico en el aprendizaje del análisis de funciones periódicas mediante el enfoque de Fourier.

El simulador proporciona al alumno un entorno gráfico que le permite obtener, de forma inmediata, la reconstrucción de señales periódicas a través de dos enfoques fundamentales en el análisis de Fourier: el dominio del tiempo y el dominio de la frecuencia. Representa una herramienta ágil para verificar y contrastar su trabajo, especialmente en el módulo de análisis en frecuencia, ya que permite reconstruir la señal a partir de los coeficientes espectrales. La interactividad de este módulo fue altamente valorada durante la etapa de validación, destacándose como una de las características más apreciadas por los usuarios.

Aún existen aspectos por mejorar, tales como: incluir una guía más explícita para el ingreso de datos en ambos menús, disminuir el valor del paso mínimo en los controles deslizantes para incrementar la precisión en la reconstrucción, utilizar un rotulado más formal y detallado en las gráficas de señal reconstruida, así como uniformar el estilo visual entre ambos módulos de cálculo. No obstante, el simulador constituye un recurso con el potencial de impactar positivamente en el aprendizaje de los estudiantes que lo utilicen.

Referencias

1. Alexander, C. K., & Sadiku, M. N. O. (2013). *Fundamentals of electric circuits* (5th ed.). McGraw-Hill Education.
2. Anderson, D. J. (2010). *Kanban: Successful evolutionary change for your technology business*. Blue Hole Press.
3. Bootstrap. (s. f.). Bootstrap Framework. <https://getbootstrap.com/>



4. Falstad, P. (s. f.). *Fourier series applet*. Recuperado el 15 de julio de 2025, de <https://www.falstad.com/fourier/>
5. Fourier, J. (1822). *Théorie analytique de la chaleur*. Paris: Firmin Didot.
6. Fourier Solver. (s. f.). *Calculadora de series de Fourier y DFT*. Recuperado el 15 de julio de 2025, de <https://fouriersolver.com/home>
7. GitHub, Inc. (s. f.). GitHub Docs. <https://docs.github.com/>
8. González-Quñones, F., Tarango, J., & Bojorquez, K. (2022). Ansiedad matemática en estudiantes mexicanos de ingeniería: correlación con el pensamiento matemático y el razonamiento Covariacional. *Revista de Psicología y Ciencias del Comportamiento de la Unidad Académica de Ciencias Jurídicas y Sociales*, 13(2), 68-89.
9. Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., Wieser, E., Taylor, J., Berg, S., Smith, N. J., Kern, R., Picus, M., Hoyer, S., van Kerkwijk, M. H., Brett, M., Haldane, A., Fernández del Río, J., Wiebe, M., Peterson, P., Gérard-Marchant, P., ... Oliphant, T. E. (2020). Array programming with NumPy. *Nature*, 585(7825), 357–362. <https://doi.org/10.1038/s41586-020-2649-2>
10. Hunter, J. D. (2007). Matplotlib: A 2D graphics environment. *Computing in Science & Engineering*, 9(3), 90–95. <https://doi.org/10.1109/MCSE.2007.55>
11. Jerri, A. J. (1998). *The Gibbs phenomenon in Fourier analysis, splines and wavelet approximations*. Springer.
12. Lifeder. (s. f.). *Series de Fourier*. Recuperado el 15 de julio de 2025, de <https://www.lifeder.com/series-de-fourier/>
13. McCarthy, L. (2024). p5.js: A JavaScript library for creative coding. The Processing Foundation. <https://p5js.org/>
14. Microsoft. (s. f.). Visual Studio Code. <https://code.visualstudio.com/>
15. OCDE (2023), *Resultados PISA 2022 (Volumenes I y II) – Notas por país: México*. Organización para la cooperación y desarrollo económico. <https://gpseducation.oecd.org/CountryProfile?primaryCountry=MEX&treshold=10&topic=PI>
16. Pepin, B., Biehler, R., & Gueudet, G. (2021). Mathematics in engineering education: A review of the recent literature with a view towards innovative practices. *International Journal of Research in Undergraduate Mathematics Education*, 7(2), 163-188. <https://doi.org/10.1007/s40753-021-00139-8>
17. Pepin, B., Choppin, J., Ruthven, K., & Sinclair, N. (2017). Digital curriculum resources in mathematics education: Foundations for change. *ZDM – Mathematics Education*, 49(5), 645–661. <https://doi.org/10.1007/s11858-017-0879-z>
18. PhET Interactive Simulations. (2007). *PhET simulation: Fourier: Making Waves* [Simulación HTML]. Universidad de Colorado Boulder. Recuperado el 1 de julio de 2025, de <https://phet.colorado.edu/es/simulations/fourier-making-waves/about>
19. Schwaber, K., & Sutherland, J. (2020). *The Scrum Guide: The Definitive Guide to Scrum: The Rules of the Game*. Scrum.org. <https://scrumguides.org>
20. Symbolab. (s. f.). *Fourier series calculator*. Recuperado el 1 de julio de 2025, de <https://es.symbolab.com/solver/fourier-series-calculator>
21. The Django Software Foundation. (s. f.). Django (version 4.x). <https://www.djangoproject.com/>
22. Van Rossum, G., & Drake, F. L. (2009). *Python 3 Reference Manual*. Scotts Valley, CA: CreateSpace.

