



# INSTITUTO POLITÉCNICO NACIONAL

UNIDAD PROFESIONAL INTERDISCIPLINARIA DE  
INGENIERÍA Y CIENCIAS SOCIALES Y ADMINISTRATIVAS

## INGENIERÍA DE PRUEBAS

6NV61

### ***Plan de Pruebas***

ALUMNOS:

- FIGUEROA HERNÁNDEZ DIANA PAOLA
- MIGUEL ALARCÓN ADRIÁN MANUEL
- RUIZ SEGURA MICHELLE
- VEGA ALONSO BRENDA

FECHA: 23 DE SEPTIEMBRE DE 2025

PROFESOR: RAMON CRUZ MARTINEZ



## **Plan de Pruebas: Calculadora V3.**

### **Objetivo:**

El objetivo de este plan de pruebas es garantizar que la aplicación funcione de manera correcta, confiable y según los requerimientos establecidos. Se busca validar que las operaciones realizadas produzcan resultados precisos y consistentes, así como comprobar el adecuado manejo de entradas inválidas y situaciones de error. Además, se evaluará la respuesta de la interfaz gráfica, asegurando que los botones, pantallas de visualización y mensajes se comporten de forma clara e intuitiva. Con ello, se pretende identificar y corregir posibles fallas que afecten la experiencia del usuario o la exactitud de los cálculos.

### **Alcance:**

El alcance de las pruebas incluirá las siguientes áreas funcionales:

1. Operaciones aritméticas básicas.
2. Validación de entradas.
3. Gestión de interfaz.
4. Gestión de memoria.

### **Criterios de Aceptación:**

1. La calculadora debe dar resultados correctos en todas las operaciones básicas.
2. Ante una división entre cero, el sistema debe mostrar un mensaje de error y no permitir resultados inválidos.
3. Solo se deben aceptar valores numéricos como entradas, rechazando caracteres no válidos.
4. Los resultados deben visualizarse de manera clara en la interfaz.
5. El punto decimal no se debe permitir en usos incorrectos.
6. La función de limpiar debe restablecer los campos para iniciar un nuevo cálculo.
7. Para las funciones de memoria:

- a. M+ debe sumar correctamente el valor actual a la memoria.
  - b. M- debe restar correctamente el valor actual de la memoria.
  - c. MR debe mostrar el valor almacenado en memoria.
  - d. MC debe borrar el valor de memoria y lo establece en cero.
8. La interfaz debe ser sencilla e intuitiva, permitiendo al usuario identificar fácilmente los botones de operaciones y memoria.
  9. Los botones M+, M-, MR y MC deben estar integrados de manera coherente con el diseño ya existente, manteniendo uniformidad y facilidad de uso.
  10. El tiempo de respuesta al ejecutar cualquier operación aritmética básica no debe superar un segundo.
  11. El acceso y ejecución de las funciones de memoria (M+, M-, MR, MC) debe completarse en un tiempo igual o inferior a un segundo.
  12. El sistema debe poder ejecutarse en equipos de escritorio sin requerir configuraciones adicionales complejas.
  13. El sistema debe manejar correctamente operaciones con más de dos cantidades (ej:  $2*3*5$ ), mostrando resultados intermedios y finales de forma secuencial y correcta.
  14. El sistema debe establecer un límite máximo de dígitos para los valores ingresados, evitando desbordamientos visuales o de cálculo.
  15. El sistema debe permitir la entrada de decimales iniciando con el punto (ej: ".5" debe interpretarse como "0.5"), no solo con "0."
  16. Al mostrar un mensaje de error, el sistema debe limpiar automáticamente los campos de entrada para permitir un nuevo cálculo sin interrupciones en el flujo.
  17. El sistema debe utilizar el símbolo de división estándar ( $\div$ ) en lugar de la diagonal simple (/).

18. Los botones numéricos deben seguir el orden estándar de una calculadora (7-8-9 en fila superior, 4-5-6, etc.).
19. El sistema debe manejar adecuadamente estados nulos o vacíos en los campos de entrada, sin interrumpir la ejecución con excepciones no controladas.
20. El sistema debe permitir la entrada de números y operaciones tanto mediante clics en los botones como mediante el teclado numérico del equipo.
21. La interfaz debe tener un tamaño mínimo fijo que garantice la visualización completa de todos los botones en diferentes resoluciones de pantalla.
22. El sistema deberá permitir realizar operaciones de porcentaje mediante el botón %, interpretando correctamente el valor porcentual en el contexto de la operación en curso.

**Recursos:**

1. Personal de pruebas asignado (Tester).
2. Computadora de escritorio o portátil (No se necesitan requerimientos específicos de hardware o software más allá de los cotidianos).
3. Archivo ejecutable de la calculadora (Calculadora\_1er\_Parcial.exe).
4. Herramienta ofimática (Word, Excel, o Documentos de Google) para el registro de los resultados obtenidos.

## **Estrategia de Pruebas.**

### **Pruebas de Unidad:**

- ☐ Verificar de manera individual cada operación aritmética y cada función de memoria.
- ☐ Validar el manejo del punto decimal.
- ☐ Validar las entradas numéricas.

### **Pruebas de Integración:**

- ☐ Evaluar la interacción entre los módulos de la calculadora.
- ☐ Validar la operatividad conjunta entre las funciones de memoria y las operaciones aritméticas.

### **Pruebas de Sistema:**

- ☐ Evaluar el sistema como un todo para garantizar la funcionalidad y el rendimiento.

## **Casos de Prueba.**

### **➤ Operaciones Aritméticas**

1. Verificar la ejecución correcta de operaciones básicas: suma, resta, multiplicación y división.
2. Validar el manejo adecuado de la división entre cero, mostrando mensaje de error.
3. Confirmar que el sistema solo acepta datos numéricos válidos antes de ejecutar una operación.
4. Validar que el resultado de la operación se muestre de forma clara en el display.
5. Comprobar el uso correcto del punto decimal, evitando ingresos incorrectos (ej. "5..5").
6. Probar la aplicación correcta de la función de porcentaje (%).

7. Validar cálculos con más de dos operaciones (resultados intermedios y finales).
  8. Verificar límite máximo de dígitos en entrada, validando que el sistema rechace o trunque la entrada apropiadamente.
- Validación de entradas.
9. Verificar que el sistema permita limpiar la pantalla para realizar nuevos cálculos.
  10. Validar que al presionar el botón de borrado (Clear/AC) se reinicie correctamente el display.
  11. Validar limpieza automática tras mensaje de error
  12. Comprobar entrada decimal iniciando con punto (".5" verificando que el sistema lo interprete correctamente como "0.5")
- Gestión de Memoria
13. Probar que el sistema permite almacenar un valor en memoria mediante M +.
  14. Validar que el sistema sume un nuevo valor al existente en memoria mediante M +.
  15. Probar la resta de un valor en memoria mediante M-.
  16. Verificar la recuperación del valor almacenado en memoria mediante MR.
  17. Confirmar que la memoria pueda ser limpiada correctamente mediante MC, mostrando 0 al recuperar.
- Gestión de Interfaz.
18. Probar que el tiempo de respuesta de las operaciones aritméticas no supere 1 segundo.
  19. Validar que el acceso y ejecución de las funciones de memoria se complete en un tiempo igual o inferior a un segundo.
  20. Verificar que la interfaz sea sencilla e intuitiva.
  21. Validar que los botones M+, M-, MR, MC, ., ÷, +, - estén integrados de manera coherente con el diseño.
  22. Probar que el sistema sea compatible en equipos de escritorio.

- 23. Probar cualquier comportamiento anómalo cuando se alterna entre teclado físico e interfaz gráfica.
- 24. Validar la adaptabilidad y responsividad de la interfaz al maximizar y minimizar la pantalla.
- 25. Validar orden estándar de botones numéricos
- 26. Probar manejo de estados nulos/vacíos

### Matrices de Pruebas.

**1. Pruebas Unitarias (Unit Testing).** Estas pruebas se centran en verificar las **unidades más pequeñas** de código, como funciones o métodos individuales (ej., la lógica detrás de una suma, o la validación de un solo *input*).

ID Req.	ID Caso de Prueba	Descripción del Caso de Prueba (Resumida)	Tipo de Prueba
RQ-01	CP-01	<b>Verificar operaciones básicas</b> (suma, resta, multiplicación, división).	Unitaria: Lógica de Cálculo
RQ-02	CP-06	Probar aplicación de porcentaje (%).	Unitaria: Lógica de Cálculo
RQ-02	CP-02	Validar manejo de división entre cero.	Unitaria: Manejo de Excepciones
RQ-03	CP-03	Confirmar ingreso solo de datos numéricos.	Unitaria: Validación de <i>Input</i>
RQ-05	CP-05	Comprobar uso correcto del punto decimal.	Unitaria: Validación de <i>Input</i>
RQ-07a	CP-13	Probar almacenamiento inicial en memoria (M+).	Unitaria: Función de Memoria

RQ-0 7a	CP-14	Validar suma acumulativa en memoria (M+).	Unitaria: Función de Memoria
RQ-0 7b	CP-15	Probar resta de un valor en memoria (M-).	Unitaria: Función de Memoria
RQ-0 7d	CP-17	Confirmar que la memoria pueda ser limpiada (MC).	Unitaria: Función de Memoria
RQ-1 3	CP-07	Validar operaciones con más de dos cantidades	Unitaria: Lógica de Cálculo
RQ-1 4	CP-08	Verificar límite máximo de dígitos	Unitaria: Validación de Input
RQ-1 5	CP-12	Comprobar entrada decimal iniciando con punto	Unitaria: Validación de Input
RQ-1 9	CP-26	Probar manejo de estados nulos/vacíos	Unitaria: Manejo de Excepciones

**2. Pruebas de Integración (Integration Testing).** Estas pruebas se centran en verificar cómo las diferentes unidades de código, módulos o servicios **interactúan entre sí** (ej., el *input* pasa la validación y luego alimenta la función de cálculo).

ID Req.	ID Caso de Prueba	Descripción del Caso de Prueba (Resumida)	Tipo de Prueba
RQ-0 4	CP-04	Validar claridad del resultado en el display.	Integración: Lógica ↔ UI
RQ-0 6	CP-09, CP-10	Verificar limpieza de pantalla y botón Clear/AC.	Integración: <i>Clear</i> Función ↔ UI



RQ-07c	CP-16	Verificar la recuperación del valor con MR.	Integración: Memoria ↔ UI
RQ-20	CP-23	Probar comportamiento anómalo al alternar entre teclado físico e interfaz gráfica.	Integración: <i>Input</i> Teclado ↔ <i>Input</i> UI
RQ-16	CP-11	Validar limpieza automática tras error	Integración: Manejo de errores ↔ UI

**3. Pruebas de Sistema (System Testing).** Estas pruebas evalúan el **sistema completo** contra los requisitos funcionales y no funcionales, a menudo simulando escenarios de usuario final.

ID Req.	ID Caso de Prueba	Descripción del Caso de Prueba (Resumida)	Tipo de Prueba
RQ-21	CP-24	Validar la adaptabilidad y responsividad de la interfaz.	Sistema: Usabilidad/Responsividad
RQ-10	CP-18	Probar que el tiempo de respuesta de las operaciones aritméticas no supere 1 segundo.	Sistema: Rendimiento
RQ-11	CP-19	Validar el tiempo de respuesta de las funciones de memoria.	Sistema: Rendimiento
RQ-08	CP-20	Verificar que la interfaz sea sencilla e intuitiva.	Sistema: Usabilidad
RQ-09	CP-21	Validar coherencia del diseño de botones de memoria.	Sistema: Usabilidad/Diseño
RQ-12	CP-22	Probar compatibilidad en equipos de escritorio.	Sistema: Compatibilidad

RQ-17	CP-25	Verificar símbolo de división estándar ( $\div$ )	Sistema: Usabilidad/Diseño
RQ-18	CP-25	Validar orden estándar de botones numéricos	Sistema: Usabilidad/Diseño

Para un mayor entendimiento se realizan las especificaciones que debe cumplir cada caso de prueba.

### 1. Operaciones Aritméticas Básicas (Suma, Resta, Multiplicación y División)

- **Válido (Pasa):** El sistema debe calcular y mostrar el **resultado exacto y correcto** para las cuatro operaciones aritméticas básicas.
- **Inválido (Falla):** El resultado de cualquiera de las operaciones es incorrecto o el sistema se bloquea.

### 2. Manejo de División entre Cero

- **Válido (Pasa):** Al intentar una división por cero, el sistema **no debe bloquearse** y debe mostrar un **mensaje de error claro** (ej., "Error: División entre Cero" o "Indefinido").
- **Inválido (Falla):** El sistema se congela, se reinicia, o arroja un resultado numérico incorrecto sin una advertencia explícita.

### 3. Aceptación Solo de Datos Numéricos

- **Válido (Pasa):** El sistema debe permitir la ejecución de operaciones solo con valores numéricos. Si se intenta ingresar un carácter no numérico antes de operar, este debe ser **ignorado o generar una alerta de entrada**.
- **Inválido (Falla):** El sistema permite ejecutar una operación que incluye caracteres no numéricos, resultando en un error o cálculo incorrecto.

### 4. Claridad en el Display del Resultado

- **Válido (Pasa):** El resultado final de la operación debe mostrarse de forma **completa, legible y en un formato estándar** en la pantalla.

- **Inválido (Falla):** El resultado se trunca, el formato es confuso, o el texto es ilegible.

#### 5. Uso Correcto del Punto Decimal (Evitando por ejemplo “5..5”)

- **Válido (Pasa):** El sistema debe permitir un único punto decimal por número e **ignorar la entrada o mostrar un mensaje de error explícito** (ej., "Entrada no válida: Solo un punto decimal permitido") si se intenta ingresar un segundo punto decimal.
- **Inválido (Falla):** El sistema acepta múltiples puntos decimales en un solo número o se bloquea al intentarlo.

#### 6. Prueba de Aplicación de la Función de Porcentaje (%)

- **Válido (Pasa):** El sistema debe calcular correctamente el porcentaje. Por ejemplo, al ingresar  $50+10\%$ , el resultado debe ser 55.
- **Inválido (Falla):** El resultado del cálculo de porcentaje es incorrecto o la función (%) genera un error o se bloquea.

#### 7. Limpieza de Pantalla para Nuevos Cálculos

- **Válido (Pasa):** El sistema debe permitir limpiar la pantalla para realizar nuevos cálculos; los campos de entrada/display se **vacían completamente o se restablecen a 0**.
- **Inválido (Falla):** Algún valor anterior o resultado parcial permanece en pantalla.

#### 8. Validación del Botón de Borrado (Clear/AC)

- **Válido (Pasa):** El botón **Clear/AC** debe funcionar de manera **consistente**, reiniciando correctamente el *display* en cualquier estado (inicio, a mitad de operación, después de un error).
- **Inválido (Falla):** El botón no borra completamente la operación o tiene un comportamiento inconsistente.

#### 9. Ingreso Mixto (Teclado Físico e Interfaz Gráfica)

- **Válido (Pasa):** El usuario debe poder ingresar números y operadores alternando indistintamente entre el teclado físico y los botones de la interfaz gráfica **sin generar errores ni resultados anómalos**.

- **Inválido (Falla):** La mezcla de métodos de entrada resulta en un error, omisión de entrada o comportamiento anómalo.

#### **10. Almacenamiento Inicial en Memoria con M+**

- **Válido (Pasa):** Se ingresa un valor y, al presionar **M+**, dicho valor queda almacenado y puede ser recuperado con **MR**.
- **Inválido (Falla):** El valor recuperado no coincide con el valor ingresado.

#### **11. Suma de Nuevo Valor al Existente en Memoria con M+**

- **Válido (Pasa):** Tras almacenar un valor inicial, al ingresar un segundo valor y presionar **M+**, la memoria debe guardar la **suma** de ambos valores.
- **Inválido (Falla):** M+ sobrescribe el valor anterior o no realiza la suma correctamente.

#### **12. Resta de un Valor de la Memoria con M-**

- **Válido (Pasa):** Al ingresar un valor y presionar **M-**, la memoria debe guardar la **resta** del valor actual de la memoria.
- **Inválido (Falla):** M- no resta el valor actual de la memoria.

#### **13. Recuperación del Valor Almacenado con MR**

- **Válido (Pasa):** El botón **MR** (Memory Recall) debe recuperar el valor actual en memoria y **mostrarlo en el display**.
- **Inválido (Falla):** MR no muestra el valor correcto o no interactúa con el *display*.

#### **14. Limpieza de Memoria con MC**

- **Válido (Pasa):** Al presionar **MC** (Memory Clear), la memoria se restablece a cero. Al presionar **MR** inmediatamente después, el *display* debe mostrar 0.
- **Inválido (Falla):** Al recuperar con MR después de MC, se sigue mostrando el valor antiguo.

#### **15. Tiempo de Respuesta de Operaciones Aritméticas**

- **Válido (Pasa):** El tiempo desde la ejecución de cualquier operación hasta la visualización del resultado debe ser **igual o inferior a un segundo ( $\leq 1s$ )**.
- **Inválido (Falla):** El tiempo de respuesta de la operación supera el límite de un segundo.

#### **16. Tiempo de Respuesta de Funciones de Memoria**

- **Válido (Pasa):** La ejecución de cualquiera de las funciones de memoria (M+, M-, MR, MC) debe completarse en un tiempo **igual o inferior a un segundo ( $\leq 1s$ )**.
- **Inválido (Falla):** El tiempo de respuesta de las funciones de memoria supera el límite de un segundo.

#### **17. Interfaz Sencilla e Intuitiva**

- **Válido (Pasa):** Los usuarios de prueba deben poder **identificar y usar correctamente** los botones y funciones principales **sin necesidad de ayuda** o manual.
- **Inválido (Falla):** Los usuarios se confunden, o el flujo de trabajo resulta engorroso.

#### **18. Integración Coherente de Botones M+, M-, MR y MC**

- **Válido (Pasa):** Los botones de memoria deben **mantener la coherencia visual** con el resto de la interfaz en términos de estilo, color, fuente y disposición.
- **Inválido (Falla):** El diseño o ubicación de los botones de memoria rompe la estética general.

#### **19. Compatibilidad en Equipos de Escritorio**

- **Válido (Pasa):** El sistema debe **ejecutarse** en equipos de escritorio estándar **sin requerir configuraciones complejas adicionales** (como instalación de *drivers* o *frameworks* de terceros).

- **Inválido (Falla):** Se requiere configuración especializada o el sistema no funciona correctamente en el entorno de escritorio objetivo.

## 20. Validación de Adaptabilidad y Responsividad de la Interfaz

- **Válido (Pasa):** Al maximizar o redimensionar la ventana, la calculadora debe **adaptarse dinámicamente**, manteniendo todos los elementos visibles y funcionales, sin quedar confinada a una esquina.
- **Inválido (Falla):** Al redimensionar, el diseño se rompe, los botones se superponen o el sistema queda fijo en un rincón de la ventana.

## 21. Validar operaciones con más de dos cantidades

- **Válido (Pasa):** Al ingresar " $2 * 3 * 5$ " el sistema muestra "6" como resultado intermedio y "30" como resultado final.
- **Inválido (Falla):** El sistema ignora la primera cantidad o muestra resultados incorrectos.

## 22. Verificar límite máximo de dígitos en entrada

- **Válido (Pasa):** Al alcanzar el límite de dígitos, el sistema no acepta más entrada o muestra indicación visual del límite.
- **Inválido (Falla):** El sistema permite entrada ilimitada causando desbordamiento visual o de cálculo.

## 23. Comprobar entrada decimal iniciando con punto

- **Válido (Pasa):** Al ingresar ".5" el sistema interpreta "0.5" y realiza operaciones decimales correctamente.
- **Inválido (Falla):** El sistema ignora el punto o trata ".5" como entero.

## 24. Validar limpieza automática tras mensaje de error

- **Válido (Pasa):** Tras mostrar error por división entre cero, los campos se limpian automáticamente.

- Inválido (Falla): Los valores permanecen en pantalla después del error.

25. Verificar símbolo de división estándar ( $\div$ )

- Válido (Pasa): El botón de división muestra claramente el símbolo " $\div$ ".
- Inválido (Falla): Muestra "/" u otro símbolo diferente.

26. Validar orden estándar de botones numéricos

- Válido (Pasa): Los números siguen distribución: 7-8-9 (fila 1), 4-5-6 (fila 2), 1-2-3 (fila 3).
- Inválido (Falla): El orden es diferente o inconsistente.

27. Probar manejo de estados nulos/vacíos

- Válido (Pasa): Al operar con campos vacíos, el sistema muestra "0" o mensaje apropiado sin excepciones.
- Inválido (Falla): El sistema lanza excepciones no controladas o se bloquea.

**Cronograma:**

Las pruebas se realizarán en 1 semana.

1. Pruebas de Unidad: Día 1.
2. Pruebas de Integración: Día 2.
3. Pruebas de Sistema: Día 3.

**Informes:**

Se generarán informes de estado semanalmente, destacando los casos de prueba probados, defectos encontrados y el progreso general del proceso de prueba.