



INSTITUTO POLITÉCNICO NACIONAL

UNIDAD PROFESIONAL INTERDISCIPLINARIA DE INGENIERÍA
Y CIENCIAS SOCIALES Y ADMINISTRATIVAS

6NM60 Ingeniería de Pruebas

Documento de diseño – Calculadora Segunda Versión

Alumnos:

García Méndez Juan Carlos
Conde Basilio Leonardo
Felipe Ramos Velázquez
Javier Enrique Villaseñor Trejo

Docente:

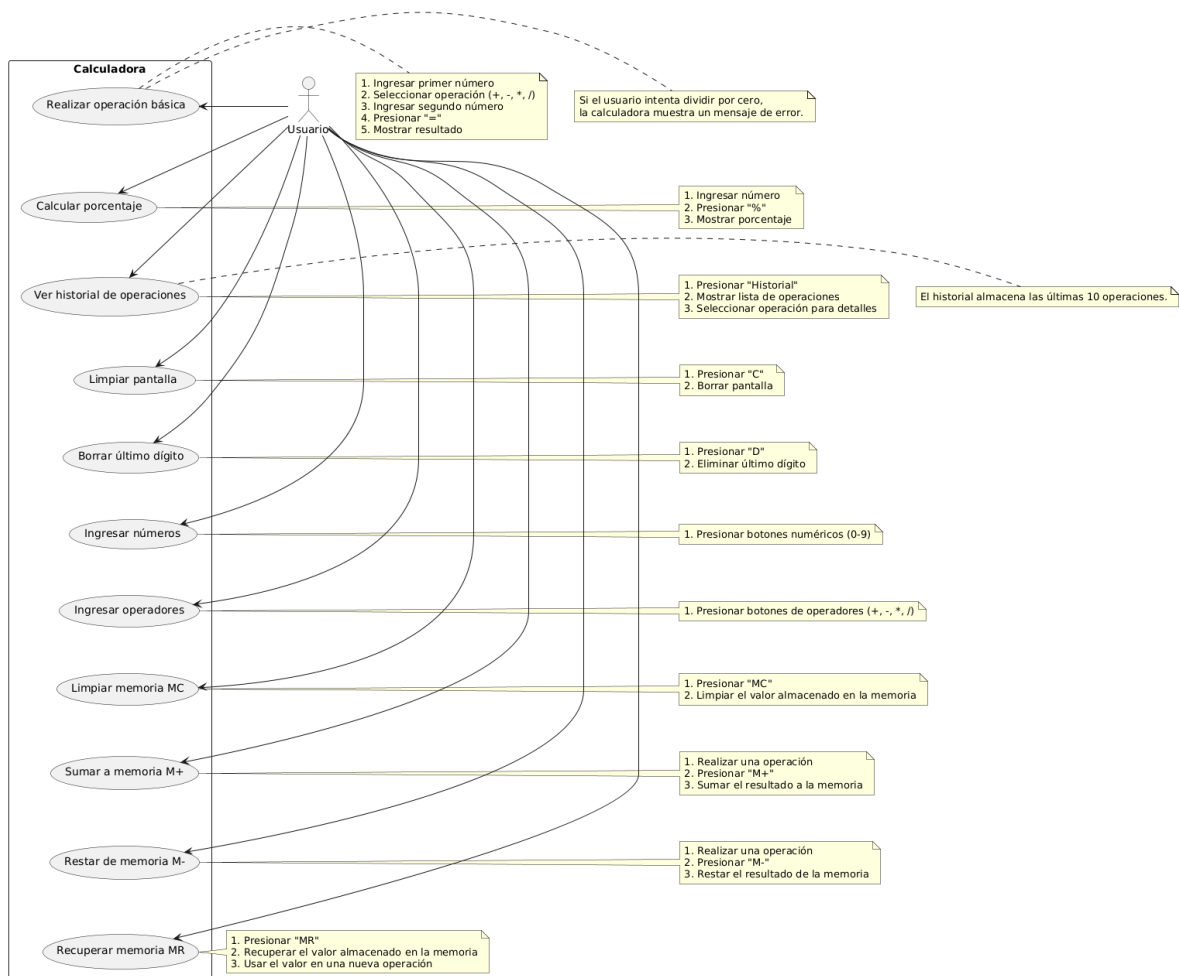
Cruz Martínez Ramón

Fecha: 17 de marzo del 2025



1. Casos de uso

Esta calculadora permitirá a los usuarios realizar operaciones básicas (suma, resta, multiplicación, división), calcular porcentajes, acceder a un historial de operaciones y gestionar una memoria con las funciones **MC**, **M+**, **M-** y **MR**. Está diseñada para ser intuitiva y fácil de usar.



Actores:

- **Usuario:** La persona que utiliza la calculadora para realizar operaciones matemáticas y gestionar la memoria.

Casos de Uso Principales:

1. **Realizar operación básica:** El usuario ingresa dos números y selecciona una operación (+, -, *, /) para obtener el resultado.
2. **Calcular porcentaje:** El usuario ingresa un número y selecciona el botón "%" para calcular su porcentaje.
3. **Ver historial de operaciones:** El usuario accede al historial para ver las operaciones realizadas anteriormente.
4. **Limpiar pantalla:** El usuario presiona el botón "C" para limpiar la pantalla y comenzar una nueva operación.
5. **Borrar último dígito:** El usuario presiona el botón "D" para borrar el último dígito ingresado.
6. **Ingresar números:** El usuario ingresa números (0-9) para realizar cálculos.
7. **Ingresar operadores:** El usuario selecciona operadores (+, -, *, /) para realizar operaciones.
8. **Limpiar memoria MC:** El usuario presiona el botón "MC" para limpiar la memoria.
9. **Sumar a memoria M+:** El usuario presiona el botón "M+" para sumar el valor actual a la memoria.
10. **Restar de memoria M-:** El usuario presiona el botón "M-" para restar el valor actual de la memoria.
11. **Recuperar memoria MR:** El usuario presiona el botón "MR" para recuperar el valor almacenado en la memoria.

Flujos de Trabajo:

1. Realizar operación básica:

- El usuario ingresa el primer número.
- El usuario selecciona una operación (+, -, *, /).
- El usuario ingresa el segundo número.
- El usuario presiona "=".
- La calculadora muestra el resultado.

2. Calcular porcentaje:

- El usuario ingresa un número.
- El usuario presiona "%".
- La calculadora muestra el porcentaje.

3. Ver historial de operaciones:

- El usuario presiona "Historial".
- La calculadora muestra una lista de las últimas operaciones realizadas.

4. Limpiar pantalla:

- El usuario presiona "C".
- La calculadora limpia la pantalla.

5. Borrar último dígito:

- El usuario presiona "D".
- La calculadora elimina el último dígito ingresado.

6. Ingresar números:

- El usuario presiona los botones numéricos (0-9).
- Los números aparecen en la pantalla.

7. Ingresar operadores:

- El usuario selecciona un operador (+, -, *, /).
- El operador se muestra en la pantalla.

8. Limpiar memoria MC:

- El usuario presiona "MC".
- La calculadora limpia el valor almacenado en la memoria.

9. Sumar a memoria M+:

- El usuario realiza una operación.
- El usuario presiona "M+".
- La calculadora suma el resultado a la memoria.

10. Restar de memoria M-:

- El usuario realiza una operación.
- El usuario presiona "M-".
- La calculadora resta el resultado de la memoria.

11. Recuperar memoria MR:

- El usuario presiona "MR".
- La calculadora recupera el valor almacenado en la memoria y lo muestra en la pantalla.

Notas Adicionales:

- Si el usuario intenta dividir por cero, la calculadora muestra un mensaje de error.
- El historial almacena las últimas 10 operaciones realizadas.

- La memoria puede almacenar un solo valor a la vez, que puede ser modificado mediante las funciones **M+**, **M-**, **MC** y **MR**.
-

2. Modelo de análisis

Entidades Principales:

1. Usuario:

- Representa al usuario que interactúa con la calculadora.
- **Atributos:**
 - id: Identificador único del usuario.
 - nombre: Nombre del usuario.
 - fechaRegistro: Fecha en la que el usuario se registró en el sistema.

2. Operacion:

- Representa una operación matemática realizada por el usuario.
- **Atributos:**
 - id: Identificador único de la operación.
 - tipo: Tipo de operación (suma, resta, multiplicación, división, porcentaje).
 - numero1: Primer número ingresado por el usuario.
 - numero2: Segundo número ingresado por el usuario.
 - resultado: Resultado de la operación.
 - fecha: Fecha en la que se realizó la operación.

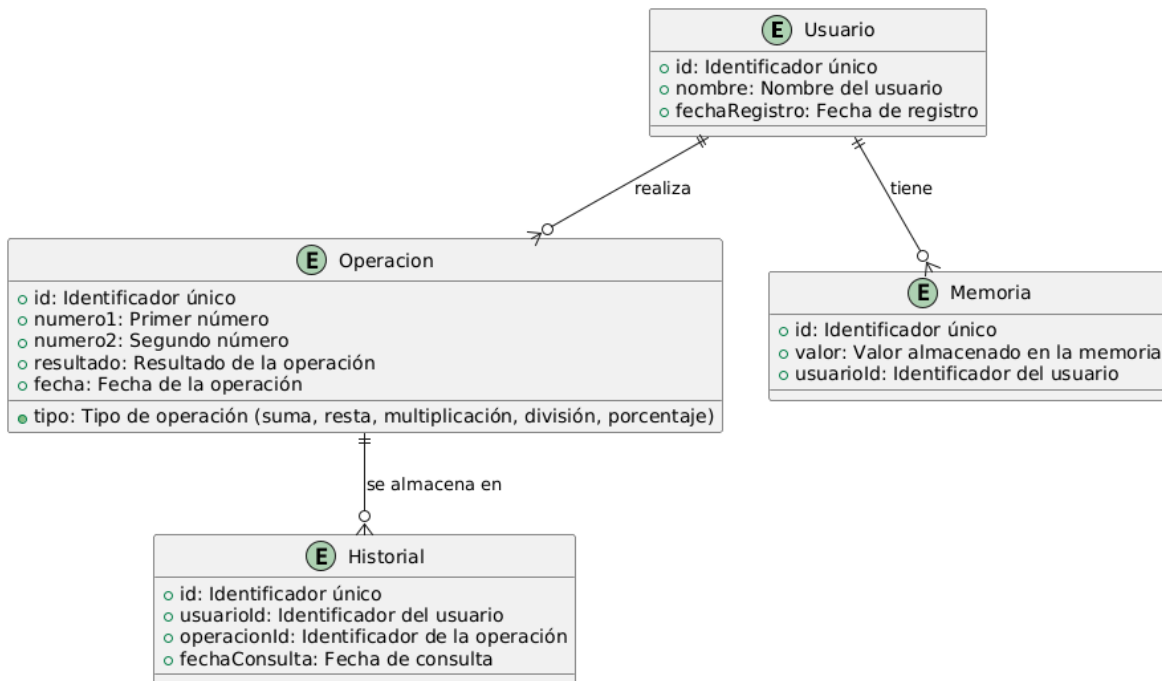
3. Historial:

- Representa el registro de las operaciones realizadas por el usuario.
- **Atributos:**
 - id: Identificador único del registro en el historial.
 - usuarioid: Identificador del usuario que realizó la operación.

- operacionId: Identificador de la operación realizada.
- fechaConsulta: Fecha en la que se consultó el historial.

4. Memoria (Nueva Entidad):

- Representa el valor almacenado en la memoria de la calculadora.
- **Atributos:**
 - id: Identificador único de la memoria.
 - valor: Valor almacenado en la memoria.
 - usuarioid: Identificador del usuario asociado a la memoria.



Relaciones:

1. Usuario realiza Operacion:

- Un **Usuario** puede realizar múltiples **Operaciones**.
- Relación: Usuario ||--o{ Operacion.

2. Operacion se almacena en Historial:

- Cada **Operacion** se almacena en el **Historial**.

- Relación: Operacion ||--o{ Historial.

3. **Usuario tiene Memoria:**

- Un **Usuario** tiene una **Memoria** asociada.
 - Relación: Usuario ||--o{ Memoria.
-

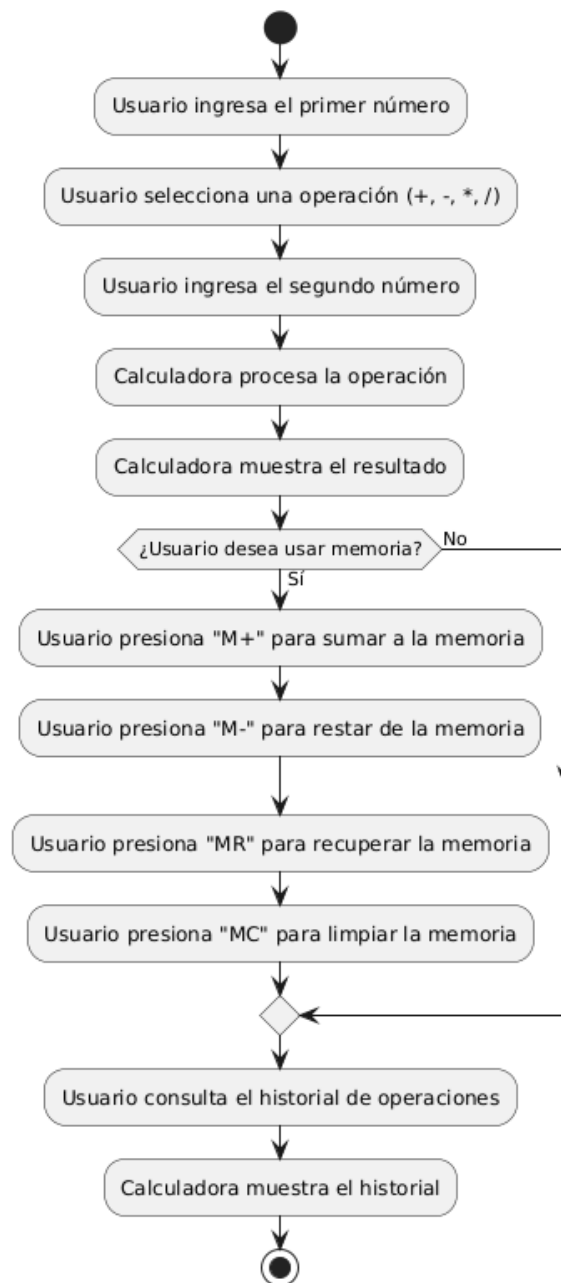
3. Diagramas basados en escenarios

Este diagrama describe el flujo de interacción entre el usuario y la calculadora para realizar una operación básica, gestionar la memoria y consultar el historial de operaciones. El flujo se divide en los siguientes pasos:

1. **Inicio:** Indica el comienzo del flujo de interacción.
2. **Usuario ingresa el primer número:** El usuario introduce el primer valor numérico.
3. **Usuario selecciona una operación:** El usuario elige entre las operaciones disponibles (suma, resta, multiplicación o división).
4. **Usuario ingresa el segundo número:** El usuario introduce el segundo valor numérico.
5. **Calculadora procesa la operación:** El sistema realiza el cálculo basado en los números y la operación seleccionada.
6. **Calculadora muestra el resultado:** El resultado de la operación se presenta al usuario.
7. **¿Usuario desea usar memoria?:** El usuario decide si desea interactuar con la memoria.
 - **Sí:**
 - **Usuario presiona "M+":** Suma el valor actual a la memoria.
 - **Usuario presiona "M-":** Resta el valor actual de la memoria.
 - **Usuario presiona "MR":** Recupera el valor almacenado en la memoria.
 - **Usuario presiona "MC":** Limpia la memoria.
 - **No:** Continúa sin interactuar con la memoria.

8. **Usuario consulta el historial de operaciones:** El usuario revisa las operaciones anteriores almacenadas en el historial.
9. **Calculadora muestra el historial:** El sistema muestra la lista de operaciones realizadas.
10. **Fin:** Indica el final del flujo de interacción.

Diagrama Basado en Escenarios - Calculadora con Memoria



Relación con el Sistema:

- Este diagrama describe un **escenario típico de uso** de la calculadora, centrándose en la realización de operaciones básicas, la gestión de la memoria y la consulta del historial.
- Refleja la **interacción entre el usuario y el sistema**, mostrando cómo el usuario proporciona entradas (números, operaciones, comandos de memoria) y cómo el sistema responde (mostrando resultados, actualizando la memoria y mostrando el historial).

Casos de Uso Relacionados:

1. **Realizar una operación básica:**
 - El usuario ingresa dos números y selecciona una operación.
 - El sistema procesa la operación y muestra el resultado.
2. **Gestionar la memoria:**
 - El usuario utiliza los botones de memoria (MC, M+, M-, MR) para interactuar con el valor almacenado.
3. **Consultar el historial de operaciones:**
 - El usuario accede al historial para ver las operaciones realizadas previamente.

4. Diagramas Orientados a Flujo

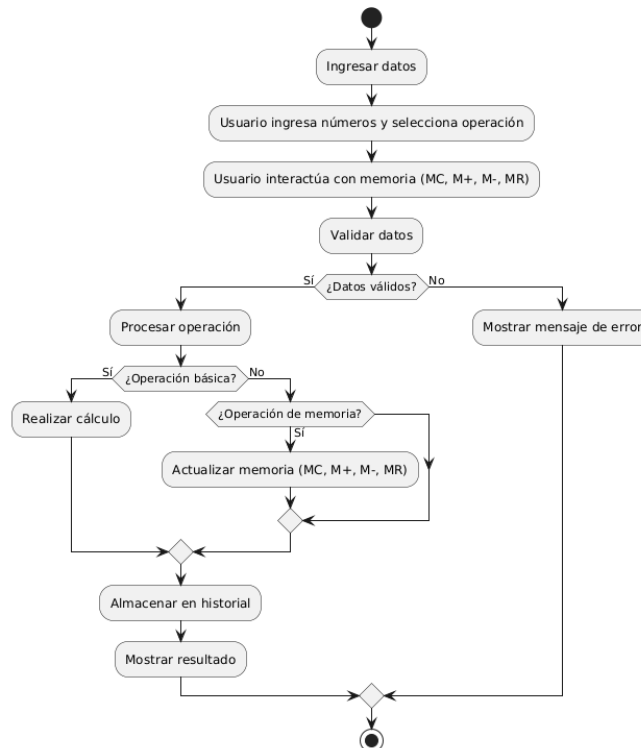
Este diagrama describe el flujo de datos dentro de la calculadora, desde la entrada del usuario hasta la generación y almacenamiento del resultado, incluyendo la gestión de la memoria. El flujo se divide en los siguientes pasos:

1. **Inicio:** Indica el comienzo del proceso.
2. **Ingresar datos:**
 - El usuario introduce números y selecciona una operación (+, -, *, /).
 - El usuario interactúa con la memoria (MC, M+, M-, MR).
3. **Validar datos:**

- El sistema verifica que los datos ingresados sean válidos (números correctos, operación soportada, comandos de memoria válidos).
- **Si los datos son válidos:**
 - **Procesar operación:**
 - Si es una **operación básica** (suma, resta, multiplicación, división), se realiza el cálculo.
 - Si es una **operación de memoria** (MC, M+, M-, MR), se actualiza el valor de la memoria.
 - **Almacenar en historial:** El resultado de la operación se guarda en el historial.
 - **Mostrar resultado:** El sistema muestra el resultado al usuario.
- **Si los datos no son válidos:**
 - **Mostrar mensaje de error:** El sistema muestra un mensaje de error al usuario.

4. **Fin:** Indica el final del proceso.

Diagrama Orientado a Flujo - Calculadora con Memoria



Relación con el Sistema:

- Este diagrama describe el **flujo de datos** dentro de la calculadora, desde la entrada del usuario hasta la generación y almacenamiento del resultado.
- Refleja cómo el sistema maneja las entradas, realiza cálculos, gestiona la memoria y almacena los resultados en el historial.

Casos de Uso Relacionados:

1. Realizar una operación básica:

- El usuario ingresa dos números y selecciona una operación.
- El sistema valida, procesa y muestra el resultado.

2. Gestionar la memoria:

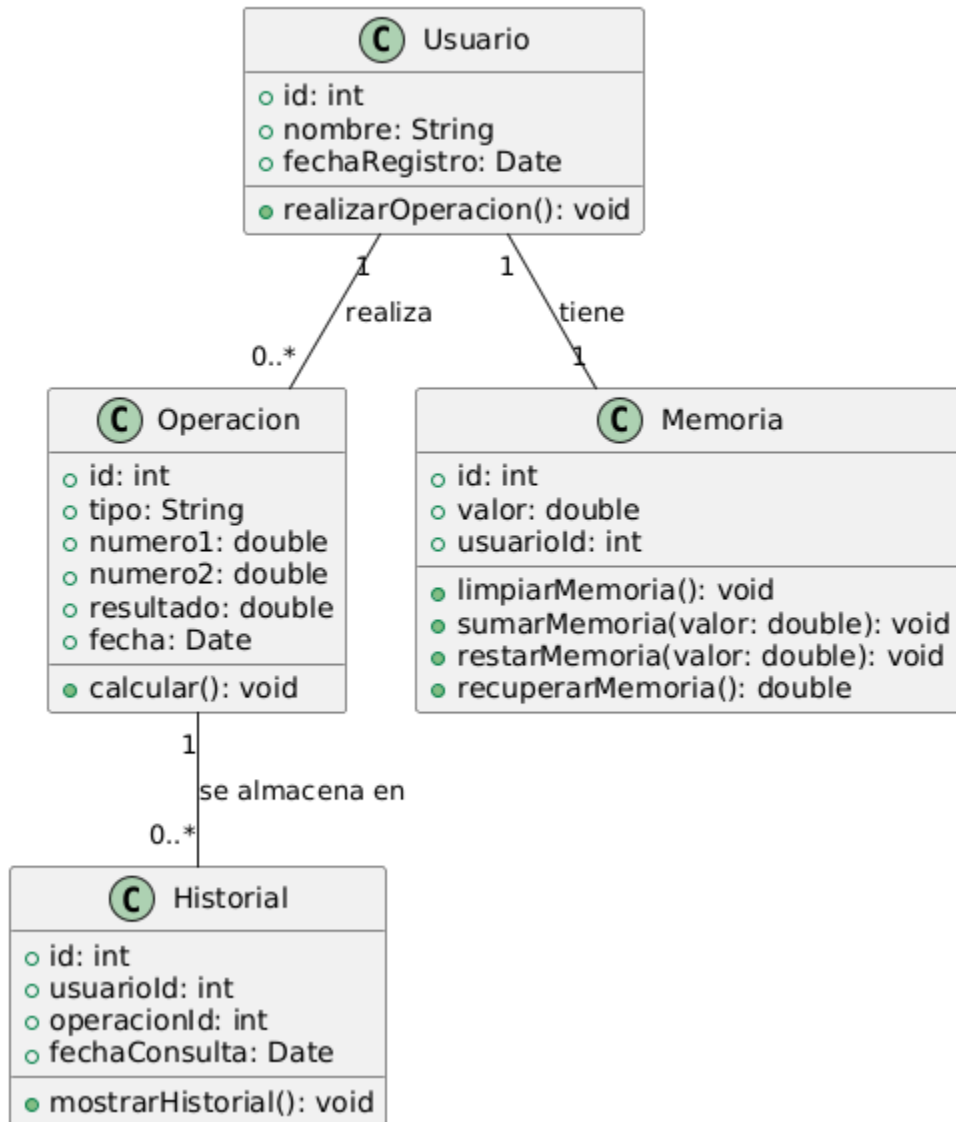
- El usuario utiliza los botones de memoria (MC, M+, M-, MR) para interactuar con el valor almacenado.

3. Almacenar en historial:

- Cada operación realizada se guarda en el historial para su posterior consulta.

5. Diagramas basados en clase

Diagrama Basado en Clases - Calculadora con Memoria



Clases Principales:

1. Usuario:

- Representa al usuario que interactúa con la calculadora.
- **Atributos:**
 - `id`: Identificador único del usuario.
 - `nombre`: Nombre del usuario.
 - `fechaRegistro`: Fecha en la que el usuario se registró en el sistema.

- **Métodos:**
 - realizarOperacion(): Permite al usuario realizar una operación.

2. Operacion:

- Representa una operación matemática realizada por el usuario.
- **Atributos:**
 - id: Identificador único de la operación.
 - tipo: Tipo de operación (suma, resta, multiplicación, división, porcentaje).
 - numero1: Primer número de la operación.
 - numero2: Segundo número de la operación.
 - resultado: Resultado de la operación.
 - fecha: Fecha en la que se realizó la operación.
- **Métodos:**
 - calcular(): Realiza el cálculo basado en los números y el tipo de operación.

3. Historial:

- Representa el registro de las operaciones realizadas por el usuario.
- **Atributos:**
 - id: Identificador único del registro en el historial.
 - usuarioid: Identificador del usuario que realizó la operación.
 - operacionId: Identificador de la operación realizada.
 - fechaConsulta: Fecha en la que se consultó el historial.
- **Métodos:**
 - mostrarHistorial(): Muestra el historial de operaciones.

4. Memoria (Nueva Clase):

- Representa el valor almacenado en la memoria de la calculadora.
- **Atributos:**
 - id: Identificador único de la memoria.
 - valor: Valor almacenado en la memoria.

- usuarioid: Identificador del usuario asociado a la memoria.
 - **Métodos:**
 - limpiarMemoria(): Limpia el valor de la memoria (MC).
 - sumarMemoria(valor: double): Suma un valor a la memoria (M+).
 - restarMemoria(valor: double): Resta un valor de la memoria (M-).
 - recuperarMemoria(): Devuelve el valor almacenado en la memoria (MR).
-

Relaciones:

1. Usuario realiza Operacion:

- Un **Usuario** puede realizar múltiples **Operaciones**.
- Relación: Usuario "1" -- "0..*" Operacion.

2. Operacion se almacena en Historial:

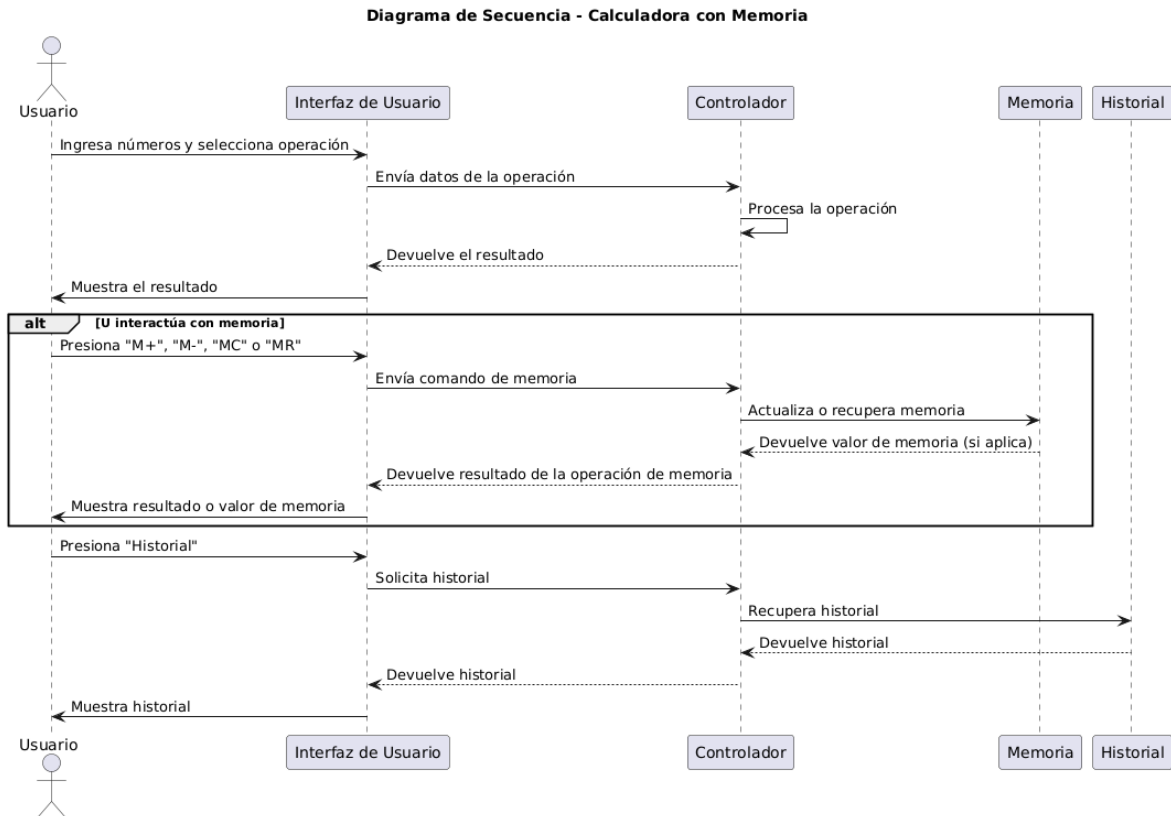
- Cada **Operacion** se almacena en el **Historial**.
- Relación: Operacion "1" -- "0..*" Historial.

3. Usuario tiene Memoria:

- Un **Usuario** tiene una **Memoria** asociada.
 - Relación: Usuario "1" -- "1" Memoria.
-

6. Diagramas de comportamiento

Este diagrama describe la interacción en tiempo real entre el usuario y la calculadora, incluyendo la realización de operaciones básicas, la gestión de la memoria y la consulta del historial.



El flujo se divide en los siguientes pasos:

1. Usuario ingresa números y selecciona operación:

- El usuario introduce números y selecciona una operación (+, -, *, /).
- La **Interfaz de Usuario (UI)** envía los datos de la operación al **Controlador (C)**.
- El **Controlador (C)** procesa la operación y devuelve el resultado a la **Interfaz de Usuario (UI)**.
- La **Interfaz de Usuario (UI)** muestra el resultado al usuario.

2. Usuario interactúa con la memoria:

- El usuario presiona uno de los botones de memoria (**M+**, **M-**, **MC**, **MR**).
- La **Interfaz de Usuario (UI)** envía el comando de memoria al **Controlador (C)**.
- El **Controlador (C)** interactúa con la **Memoria (M)** para actualizar o recuperar el valor almacenado.
- La **Memoria (M)** devuelve el valor (si aplica) al **Controlador (C)**.

- El **Controlador (C)** devuelve el resultado de la operación de memoria a la **Interfaz de Usuario (UI)**.
- La **Interfaz de Usuario (UI)** muestra el resultado o el valor de la memoria al usuario.

3. Usuario consulta el historial:

- El usuario presiona el botón "Historial".
- La **Interfaz de Usuario (UI)** solicita el historial al **Controlador (C)**.
- El **Controlador (C)** recupera el historial de la **Historial (H)**.
- La **Historial (H)** devuelve el historial al **Controlador (C)**.
- El **Controlador (C)** devuelve el historial a la **Interfaz de Usuario (UI)**.
- La **Interfaz de Usuario (UI)** muestra el historial al usuario.

Relación con el Sistema:

- Este diagrama describe la **interacción en tiempo real** entre el usuario y la calculadora.
- Refleja cómo el sistema responde a las entradas del usuario y cómo se generan los resultados, se gestiona la memoria y se consulta el historial.

Casos de Uso Relacionados:

1. Realizar una operación básica:

- El usuario ingresa dos números y selecciona una operación.
- La calculadora procesa la operación y muestra el resultado.

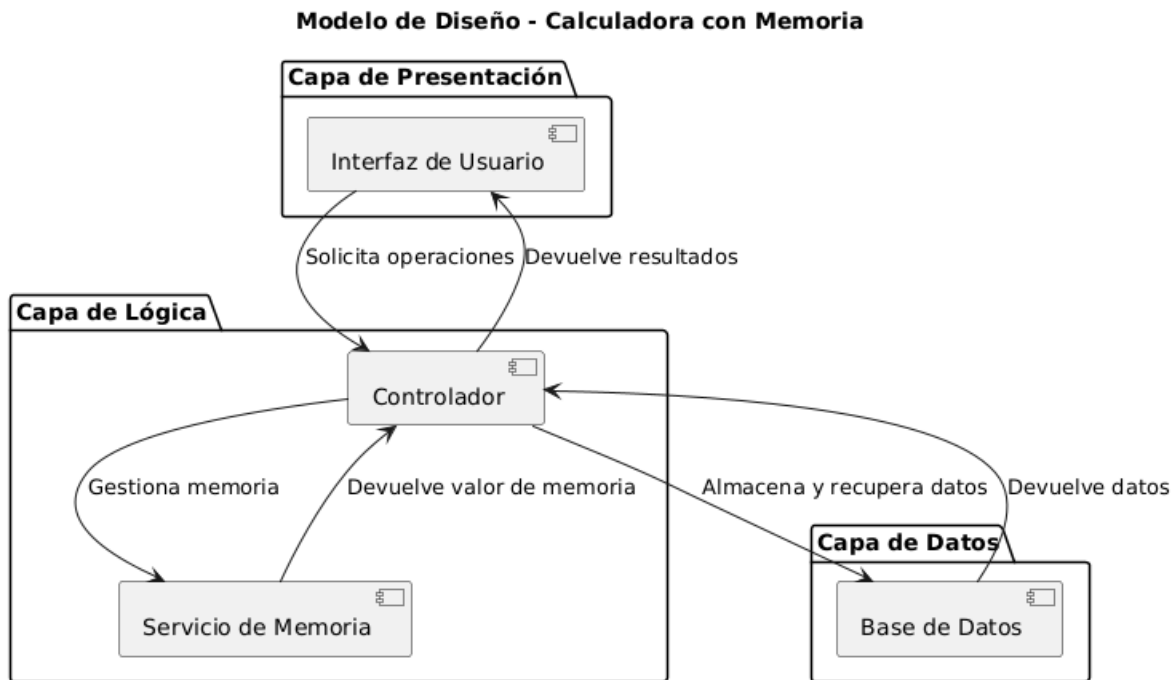
2. Gestionar la memoria:

- El usuario utiliza los botones de memoria (MC, M+, M-, MR) para interactuar con el valor almacenado.

3. Consultar el historial de operaciones:

- El usuario accede al historial para ver las operaciones realizadas previamente.
-

7. Modelo de diseño



El modelo de diseño describe la arquitectura del sistema en tres capas principales:

1. Capa de Presentación:

- **Interfaz de Usuario (UI):** Maneja la interacción directa con el usuario, como la entrada de números, la selección de operaciones y la visualización de resultados.
- **Responsabilidades:**
 - Recibir entradas del usuario (números, operaciones, comandos de memoria).
 - Mostrar resultados y mensajes al usuario.

2. Capa de Lógica:

- **Controlador (C):** Gestiona las operaciones matemáticas y la lógica de negocio. Recibe las entradas del usuario, realiza los cálculos y devuelve los resultados.
- **Servicio de Memoria (SM):** Gestiona las operaciones relacionadas con la memoria (MC, M+, M-, MR).
- **Responsabilidades:**
 - Procesar operaciones básicas (suma, resta, multiplicación, división).

- Gestionar la memoria (actualizar, recuperar, limpiar).
- Interactuar con la **Capa de Datos** para almacenar y recuperar información.

3. Capa de Datos:

- **Base de Datos (BD)**: Almacena el historial de operaciones realizadas por los usuarios y el valor de la memoria.
 - **Responsabilidades**:
 - Almacenar operaciones en el historial.
 - Almacenar y recuperar el valor de la memoria.
-

Relaciones entre las Capas:

1. Interfaz de Usuario (UI) ↔ Controlador (C):

- La **Interfaz de Usuario (UI)** envía solicitudes de operaciones al **Controlador (C)**.
- El **Controlador (C)** devuelve los resultados a la **Interfaz de Usuario (UI)**.

2. Controlador (C) ↔ Servicio de Memoria (SM):

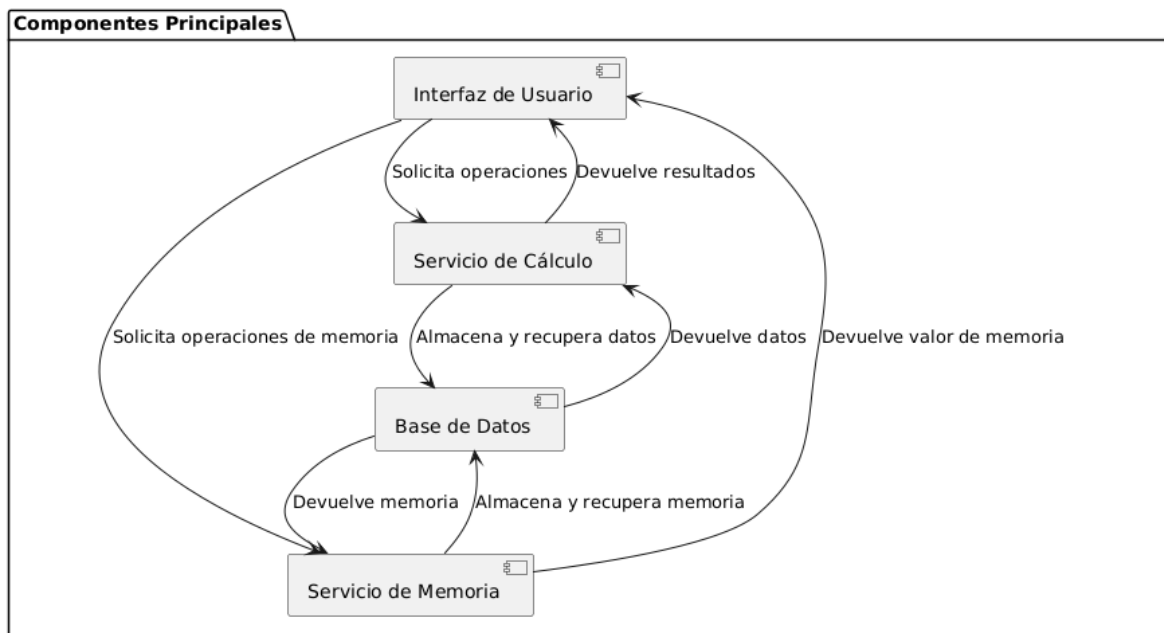
- El **Controlador (C)** interactúa con el **Servicio de Memoria (SM)** para gestionar la memoria (MC, M+, M-, MR).
- El **Servicio de Memoria (SM)** devuelve el valor de la memoria al **Controlador (C)**.

3. Controlador (C) ↔ Base de Datos (BD):

- El **Controlador (C)** almacena y recupera datos (historial de operaciones, valor de la memoria) en la **Base de Datos (BD)**.
- La **Base de Datos (BD)** devuelve los datos solicitados al **Controlador (C)**.

8. Diagramas arquitectónicos

Diagrama Arquitectónico - Calculadora con Memoria



Este diagrama describe la arquitectura del sistema a nivel de componentes, mostrando cómo los componentes principales interactúan para realizar las operaciones, gestionar la memoria y almacenar los resultados. Los componentes principales son:

1. Interfaz de Usuario (UI):

- Componente que maneja la interacción con el usuario, como la entrada de números, la selección de operaciones y la visualización de resultados.
- **Responsabilidades:**
 - Recibir entradas del usuario (números, operaciones, comandos de memoria).
 - Mostrar resultados y mensajes al usuario.

2. Servicio de Cálculo (SC):

- Componente que realiza las operaciones matemáticas (suma, resta, multiplicación, división).
- **Responsabilidades:**
 - Procesar operaciones básicas.

- Interactuar con la **Base de Datos (BD)** para almacenar y recuperar el historial de operaciones.

3. **Servicio de Memoria (SM):**

- Componente que gestiona las operaciones relacionadas con la memoria (MC, M+, M-, MR).
- **Responsabilidades:**
 - Actualizar, recuperar y limpiar la memoria.
 - Interactuar con la **Base de Datos (BD)** para almacenar y recuperar el valor de la memoria.

4. **Base de Datos (BD):**

- Componente que almacena el historial de operaciones realizadas por los usuarios y el valor de la memoria.
- **Responsabilidades:**
 - Almacenar operaciones en el historial.
 - Almacenar y recuperar el valor de la memoria.

Relaciones entre los Componentes:

1. **Interfaz de Usuario (UI) ↔ Servicio de Cálculo (SC):**

- La **Interfaz de Usuario (UI)** envía solicitudes de operaciones al **Servicio de Cálculo (SC)**.
- El **Servicio de Cálculo (SC)** devuelve los resultados a la **Interfaz de Usuario (UI)**.

2. **Interfaz de Usuario (UI) ↔ Servicio de Memoria (SM):**

- La **Interfaz de Usuario (UI)** envía solicitudes de operaciones de memoria al **Servicio de Memoria (SM)**.
- El **Servicio de Memoria (SM)** devuelve el valor de la memoria a la **Interfaz de Usuario (UI)**.

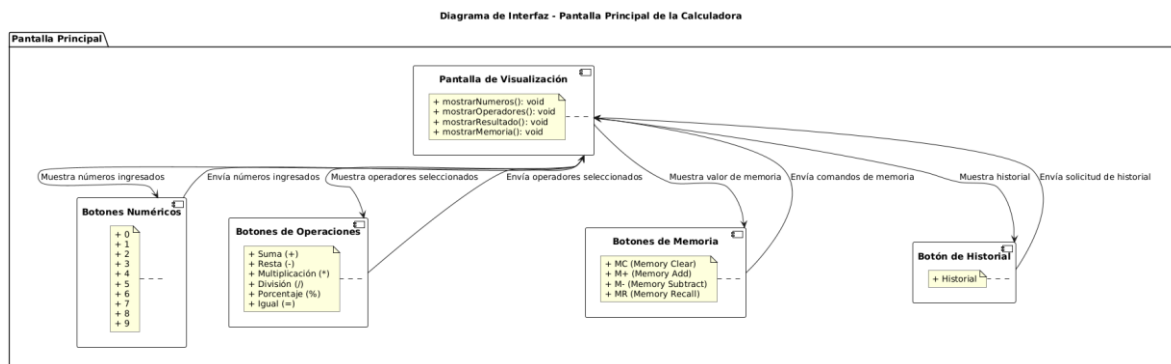
3. **Servicio de Cálculo (SC) ↔ Base de Datos (BD):**

- El **Servicio de Cálculo (SC)** almacena y recupera datos (historial de operaciones) en la **Base de Datos (BD)**.
- La **Base de Datos (BD)** devuelve los datos solicitados al **Servicio de Cálculo (SC)**.

4. Servicio de Memoria (SM) ↔ Base de Datos (BD):

- El **Servicio de Memoria (SM)** almacena y recupera el valor de la memoria en la **Base de Datos (BD)**.
- La **Base de Datos (BD)** devuelve el valor de la memoria al **Servicio de Memoria (SM)**.

9. Diagrama de interfaz



Este diagrama describe la **interfaz gráfica de usuario (GUI)** de la calculadora, específicamente la **pantalla principal**, que incluye los siguientes elementos:

1. Pantalla de Visualización (PV):

- Muestra los números ingresados, los operadores seleccionados, el resultado de las operaciones, el valor de la memoria y el historial de operaciones.
- **Métodos:**
 - `mostrarNumeros()`: Muestra los números ingresados por el usuario.
 - `mostrarOperadores()`: Muestra los operadores seleccionados.
 - `mostrarResultado()`: Muestra el resultado de la operación.
 - `mostrarMemoria()`: Muestra el valor almacenado en la memoria.

2. Botones Numéricos (BN):

- Representan los números del 0 al 9.
- **Funcionalidad:**

- El usuario presiona estos botones para ingresar números.

3. Botones de Operaciones (BO):

- Representan las operaciones básicas (suma, resta, multiplicación, división), el porcentaje (%) y el botón de igual (=).
- **Funcionalidad:**
 - El usuario selecciona estos botones para realizar operaciones.

4. Botones de Memoria (BM):

- Representan las funciones de memoria: **MC**, **M+**, **M-**, **MR**.
- **Funcionalidad:**
 - **MC**: Limpiar la memoria.
 - **M+**: Sumar el valor actual a la memoria.
 - **M-**: Restar el valor actual de la memoria.
 - **MR**: Recuperar el valor almacenado en la memoria.

5. Botón de Historial (BH):

- Permite al usuario consultar el historial de operaciones.
- **Funcionalidad:**
 - El usuario presiona este botón para ver las operaciones anteriores.

Relaciones entre los Elementos:

1. Pantalla de Visualización (PV) ↔ Botones Numéricos (BN):

- La **Pantalla de Visualización (PV)** muestra los números ingresados mediante los **Botones Numéricos (BN)**.
- Los **Botones Numéricos (BN)** envían los números ingresados a la **Pantalla de Visualización (PV)**.

2. Pantalla de Visualización (PV) ↔ Botones de Operaciones (BO):

- La **Pantalla de Visualización (PV)** muestra los operadores seleccionados mediante los **Botones de Operaciones (BO)**.
- Los **Botones de Operaciones (BO)** envían los operadores seleccionados a la **Pantalla de Visualización (PV)**.

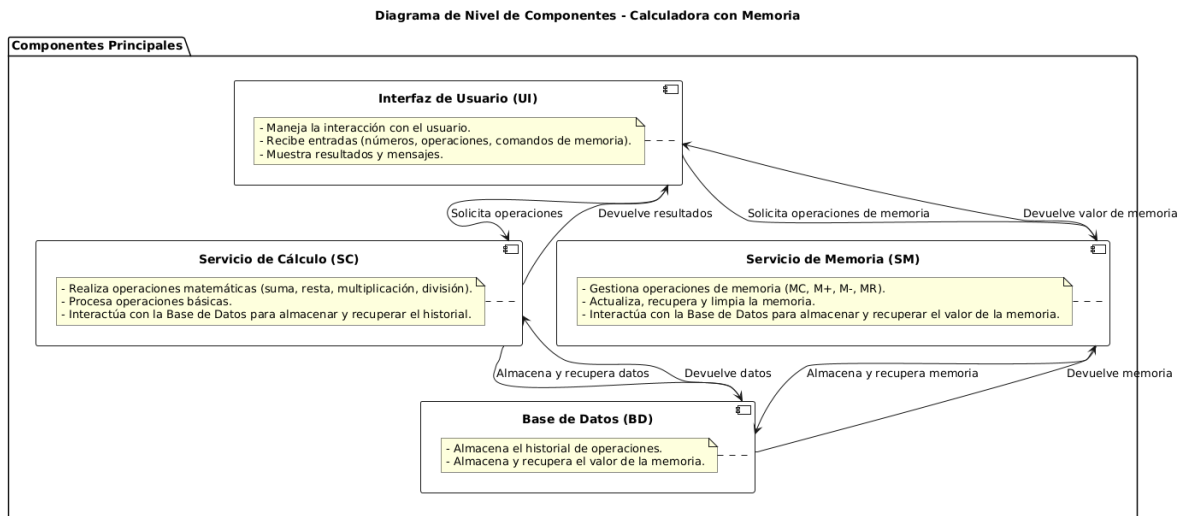
3. Pantalla de Visualización (PV) ↔ Botones de Memoria (BM):

- La **Pantalla de Visualización (PV)** muestra el valor de la memoria cuando el usuario interactúa con los **Botones de Memoria (BM)**.
- Los **Botones de Memoria (BM)** envían comandos de memoria a la **Pantalla de Visualización (PV)**.

4. Pantalla de Visualización (PV) ↔ Botón de Historial (BH):

- La **Pantalla de Visualización (PV)** muestra el historial de operaciones cuando el usuario presiona el **Botón de Historial (BH)**.
- El **Botón de Historial (BH)** envía una solicitud de historial a la **Pantalla de Visualización (PV)**.

10. Diagramas de nivel componente



Este diagrama describe la estructura del sistema a nivel de componentes, mostrando cómo los componentes principales interactúan para realizar las operaciones, gestionar la memoria y almacenar los resultados. Los componentes principales son:

1. Interfaz de Usuario (UI):

- Componente que maneja la interacción con el usuario, como la entrada de números, la selección de operaciones y la visualización de resultados.
- **Responsabilidades:**

- Recibir entradas del usuario (números, operaciones, comandos de memoria).
- Mostrar resultados y mensajes al usuario.

2. **Servicio de Cálculo (SC):**

- Componente que realiza las operaciones matemáticas (suma, resta, multiplicación, división).
- **Responsabilidades:**
 - Procesar operaciones básicas.
 - Interactuar con la **Base de Datos (BD)** para almacenar y recuperar el historial de operaciones.

3. **Servicio de Memoria (SM):**

- Componente que gestiona las operaciones relacionadas con la memoria (MC, M+, M-, MR).
- **Responsabilidades:**
 - Actualizar, recuperar y limpiar la memoria.
 - Interactuar con la **Base de Datos (BD)** para almacenar y recuperar el valor de la memoria.

4. **Base de Datos (BD):**

- Componente que almacena el historial de operaciones realizadas por los usuarios y el valor de la memoria.
- **Responsabilidades:**
 - Almacenar operaciones en el historial.
 - Almacenar y recuperar el valor de la memoria.

Relaciones entre los Componentes:

1. **Interfaz de Usuario (UI) ↔ Servicio de Cálculo (SC):**

- La **Interfaz de Usuario (UI)** envía solicitudes de operaciones al **Servicio de Cálculo (SC)**.
- El **Servicio de Cálculo (SC)** devuelve los resultados a la **Interfaz de Usuario (UI)**.

2. **Interfaz de Usuario (UI) ↔ Servicio de Memoria (SM):**

- La **Interfaz de Usuario (UI)** envía solicitudes de operaciones de memoria al **Servicio de Memoria (SM)**.
- El **Servicio de Memoria (SM)** devuelve el valor de la memoria a la **Interfaz de Usuario (UI)**.

3. **Servicio de Cálculo (SC) ↔ Base de Datos (BD):**

- El **Servicio de Cálculo (SC)** almacena y recupera datos (historial de operaciones) en la **Base de Datos (BD)**.
- La **Base de Datos (BD)** devuelve los datos solicitados al **Servicio de Cálculo (SC)**.

4. **Servicio de Memoria (SM) ↔ Base de Datos (BD):**

- El **Servicio de Memoria (SM)** almacena y recupera el valor de la memoria en la **Base de Datos (BD)**.
- La **Base de Datos (BD)** devuelve el valor de la memoria al **Servicio de Memoria (SM)**.