



UNIVERSIDAD  
**NACIONAL**  
DE COLOMBIA

---

### ENTREGA 3

Integrantes:

David Camilo Cortes Salazar

Juan Manuel Espitia Pizza

Andrew Nicolay Prieto Mendoza

Sebastian Steeven Rodriguez Ortiz

Yeswah Gonzalez Tapia

Docente

Oscar Eduardo Alvarez Rodriguez

Universidad Nacional de Colombia  
Sede Bogotá

Bogotá, Colombia  
13 de octubre 2025

| Requisito              |  |   | Estimación | Argumento  |
|------------------------|--|---|------------|--|
| RF-ADM-01.1            |  | RF-ADM-01.1: Como administrador, quiero editar o desactivar los permisos de usuarios existentes en el sistema para gestionar el acceso de acuerdo a las necesidades de la organización.   | 3          | Requiere una interfaz que permita al administrador poder acceder a la funcionalidad, lo que requiere un correcto manejo de permisos y actualizaciones en la base de datos. |
| SHOULD                 |  |   |            |  |
| RF-ADM-01.2            |  | RF-ADM-01.2: Como administrador, quiero crear usuarios con sus datos personales compuestos por: nombre, apellido, correo electrónico, cédula y fecha de registro, para gestionar quién tiene acceso al sistema.   | 2          | Sencillo, ya que únicamente requiere un formulario con los campos especificados y un verificador para cada campo.  |
| MUST<br>CU_1_jmpizza   |  |   |            |  |
| RF-ADM-01.3            |  | RF-ADM-01.3: Como administrador, quiero asignar un rol a cada usuario disponible entre USER (Cajero), OP (Operador de Cierre), o ADMIN (Administrador) para establecer límites en sus funciones dentro del sistema.   | 1          | Es sencillo ya que se complementa con el requisito RF-ADM-01.2 y la lista de tipos de usuarios es fija.  |
| MUST<br>CU_2_jmpizza   |  |   |            |  |
| RF-ADM-02.1            |  | RF-ADM-02.1: Como administrador, quiero poder agregar o deshabilitar tipos de medios de pago, con la restricción de que los medios de pago "efectivo" y "tarjeta" no puedan ser eliminados, para asegurar que las opciones básicas siempre estén disponibles y mantener actualizados distintos métodos de pago que puedan surgir. | 3          | Requiere CRUD con validaciones especiales que no permitan que los métodos de pago "efectivo" o "tarjeta" sean deshabilitados.  |
| MUST<br>CU_1_aprietime |  |   |            |  |
| RF-ADM-02.2            |  | RF-ADM-02.2: Los cambios en los métodos de pago deben reflejarse en tiempo real en los formularios de registro y cierre para garantizar la consistencia en la información.  | 4          | Requiere comunicación en tiempo real y actualización dinámica para asegurar que se refleje correctamente.  |
| SHOULD                 |  |   |            |  |
| RNF-ADM-02.3           |  | RNF-ADM-02.3: El sistema no debe permitir eliminar ni modificar un método de pago (por ejemplo, "Nequi", "Daviplata") si el mismo está siendo visto desde los movimientos o cierres históricos, esto con el fin de conservar la integridad de la información de la base de datos.   | 5          | Necesita gestión de integridad referencial y bloqueo de registros históricos, lo que implica un fuerte control de permisos.  |
| COULD                  |  |   |            |  |
| RF-01.1                |  | RF-01.1: Como cajero, quiero registrar múltiples saldos iniciales (positivos) de caja al inicio de cada día o turno, según el caso, para establecer una base clara  | 4          | Requiere una interfaz que permita definir el origen de los saldos iniciales, con una validación de montos y asegurar   |
| MUST<br>CU_3_jmpizza   |  |   |            |  |

|                            |  |  |   |   |
|----------------------------|--|--|---|---|
| a                          |  | de control financiero y poder reflejar todos los medios de pago utilizados.  |   | la persistencia.  |
| RF-01.2                    |  | RF-01.2: Como cajero o administrador, quiero registrar y visualizar movimientos de caja (ingresos o egresos) con detalles como fecha, hora, tipo de saldo, monto y descripción, asegurando que cada transacción se clasifique de forma mutuamente excluyente como ingreso o egreso, para mantener un historial detallado, ordenado y trazable. | 4 | Requiere el diseño de la interfaz que permita las restricciones requeridas además de su correcta comunicación y manipulación de la base de datos.   |
| MUST<br>CU_1_dacort<br>ess |  |  |   |   |
| RF-01.3                    |  | RF-01.3: Como administrador, quiero que una vez confirmado el registro de saldos iniciales, solo usuarios con permisos de administrador puedan modificarlos para evitar alteraciones indebidas.  | 1 | Sencillo, ya que es hacer una validación con la base de datos   |
| SHOULD                     |  |  |   |   |
| RF-01.4                    |  | RF-01.4: Como cajero, quiero poder añadir una descripción opcional al hacer un movimiento y además poder agregar múltiples movimientos usando un botón "+" para incluir varios registros antes de hacer clic en "Registrar movimientos" y así registrar todos juntos en un solo paso.  | 2 | Es sencillo pues sólo necesita añadir campos en la interfaz y dependiendo la cantidad hacer el registro en la base de datos   |
| SHOULD                     |  |  |   |   |
| RF-02.1                    |  | RF-02.1: Como operador de cierre, quiero que el sistema calcule automáticamente el saldo esperado y muestre el historial de cierres anteriores por tipo de saldo, para identificar rápidamente diferencias significativas entre el saldo esperado y el real, así como patrones o variaciones inusuales en ingresos y egresos.                  | 5 | Requiere crear un panel con el histórico de cierres anteriores, poder filtrarlos por diferentes características e identificar relaciones entre registros. Esto implica hacer manipulación y optimización de consultas a la base de datos además de aplicar técnicas de análisis sobre esta información. |
| MUST<br>CU_2_dacort<br>ess |  |  |   |   |
| RF-02.2                    |  | RF-02.2: El sistema permite calcular de manera automática el saldo esperado con la fórmula: Saldo inicial + Total ingresos – Total egresos, por cada tipo de saldo, para facilitar la conciliación.  | 2 | Es sencillo hacer el cálculo del saldo esperado según el tipo.  |
| MUST<br>CU_3_dacort<br>ess |  |  |   |   |
| RF-02.3                    |  | RF-02.3: Como operador de cierre, quiero ingresar manualmente el saldo contado real para contrastarlo con el saldo esperado para detectar y corregir posibles diferencias o errores  | 1 | Sencillo, es crear un campo donde ingresar la cantidad y compararlo con el saldo esperado calculado gracias a el requisito RF-02.3  |
| SHOULD                     |  |  |   |   |

|                                |  |  |   |  |
|--------------------------------|--|--|---|--|
| RF-02.4                        |  | RF-02.4: Como operador de cierre, quiero agregar observaciones opcionales en forma de comentarios para justificar las diferencias detectadas.  | 2 | Es sencillo pues implica añadir un campo de texto que luego se guarde como una propiedad en la base de datos.  |
| COULD                          |  |  |   |  |
| RF-02.5                        |  | RF-02.5: Como operador de cierre, quiero guardar los datos del cierre de caja (fecha, hora, usuario, saldos esperados, contados, diferencias y observaciones) para auditoría.  | 1 | Es sencillo y parte fundamental de la aplicación pues guarda la operación hecha en la base de datos para poder consultarla en el futuro.                               |
| MUST<br>CU_1_yegon<br>zalezt   |  |  |   |  |
| RF-02.6                        |  | RF-02.6: Como operador de cierre, quiero realizar cierres de caja a través de un botón que lo confirme, para mantener el control de responsabilidades y garantizar que el estado final de la caja quede registrado de forma segura y verificable.  | 2 | Es sencillo, ya que solo requiere un botón que ejecute la función de cierre, registre los datos en la base de datos y actualice el estado de la caja a "cerrada".      |
| MUST<br>CU_1_serodr<br>iguezor |  |  |   |  |
| RF-02.7                        |  | RF-02.7: Como operador, quiero que el sistema utilice modelos predictivos basados en datos históricos de cierres anteriores para estimar el flujo de caja esperado en fechas de alto movimiento, como los días 15 y 30 de cada mes, permitiendo anticipar variaciones y ajustar el cierre de caja en consecuencia.   | 5 | Requiere análisis de datos históricos y la implementación de un modelo predictivo, lo que implica mayor complejidad técnica y uso de algoritmos de aprendizaje básico. |
| COULD                          |  |  |   |  |
| RF-02.8                        |  | RF-02.8: El sistema no permite que una vez confirmado el cierre, este pueda modificarse, para preservar la integridad de la información.   | 1 | Es sencillo, solo se necesita bloquear la edición de registros de cierre confirmados mediante validaciones de estado en la base de datos.                              |
| MUST<br>CU_2_yegon<br>zalezt   |  |  |   |  |
| RF-03.1                        |  | RF-03.1: El sistema debe permitir la generación de alertas automáticas basadas en reglas configurables por un usuario con permisos de administrador. Estas alertas estarán orientadas a advertir sobre situaciones anómalas o críticas, como diferencias en cierres de caja, transacciones fuera del horario habitual, montos inusualmente altos, entre otros eventos que puedan representar posibles errores operativos, para evitar fraudes o incumplimientos. | 4 | Requiere definir condiciones dinámicas y reglas que permitan el monitoreo de eventos del sistema, lo que implica trabajo de backend y validaciones periódicas.         |
| MUST<br>CU_2_serodr<br>iguezor |  |  |   |  |
| RF-03.2                        |  | RF-03.2: El sistema permite almacenar todas las alertas con ID, fecha, hora, tipo, usuario implicado y descripción técnica para facilitar la trazabilidad.   | 2 | Requiere escribir en la base de datos para guardar las alertas y enlazarla con los módulos correspondientes.   |
| SHOULD                         |  |  |   |  |
| RF-03.3                        |  | RF-03.3: Como administrador,   | 3 | Implica crear una interfaz con   |

|                              |  |  |   |   |
|------------------------------|--|--|---|---|
| MUST<br>CU_3_yegon<br>zalezt |  | quiero visualizar todas las alertas creadas en una interfaz filtrable (por usuario, tipo de alerta, rango de fechas) y a su vez brindar la opción de ordenar de forma descendente(por defecto) y ascendente según la fecha de creación, con el objetivo de facilitar el monitoreo.                 |   | filtros y orden dinámico, lo que requiere manipulación de consultas SQL.                                    |
| RF-03.4                      |  | RF-03.4: Como administrador, quiero ajustar los parámetros de configuración (umbrales, cantidades mínimas y condiciones) para definir exactamente cuándo se activan las alertas en el sistema.   | 4 | Requiere desarrollar una sección de configuración editable y vinculada con la configuración de alertas.     |
| COULD                        |  |  |   |   |
| RNF-03.5                     |  | RNF-03.5: Como administrador, quiero que los valores por defecto de las alertas sean pre cargados para cada tipo de alerta al acceder al módulo de configuración de las alertas, para facilitar la revisión global de los parámetros y permitir su ajuste sin tener que introducirlos manualmente. | 4 | Requiere crear una interfaz para configurar todo el sistema de alertas y crear configuraciones por defecto. |
| COULD                        |  |  |   |   |
| RNF-03.6                     |  | RNF-03.6: Como administrador, quiero que se me muestre una ventana de confirmación al momento de hacer cualquier modificación que requiera permisos de administrador, con el fin de evitar errores al momento de realizar modificaciones importantes al sistema.                                   | 2 | Es sencillo crear una ventana de confirmación.  |
| SHOULD                       |  |  |   |   |
| RNF-03.7                     |  | RNF-03.7: El sistema permite registrar de manera automática cada una de las modificaciones que se han realizado en la configuración de las alertas, incluyendo fecha, hora y usuario que realizaron, esto con el fin de poder revertir modificaciones en caso de error.                            | 3 | Se deriva de RNF-03.5 para crear registros en el log.   |
| SHOULD                       |  |  |   |   |
| RNF-03.8                     |  | RNF-03.8: El sistema permite, solo a usuarios con permisos de administrador, acceder al módulo de configuración de alertas con el fin de evitar que usuarios con otros permisos puedan realizar modificaciones no autorizadas.   | 2 | Es sencillo verificar el rol del usuario actual y si es el caso permitir las acciones a realizar.           |
| SHOULD                       |  |  |   |   |
| RNF-03.9                     |  | RNF-03.9: El sistema no debe permitirle al administrador editar ni eliminar alertas ya generadas desde el módulo de configuración  | 2 | Es sencillo prohibir que un usuario no pueda realizar una acción  |
| SHOULD                       |  |  |   |   |

|                     |  |  |   |   |
|---------------------|--|--|---|---|
|                     |  | e historial, para evitar que un usuario con el rol de administrador pueda modificar el historial y dificultar el proceso de auditoría al eliminar cualquier notificación.  |   |   |
| RF-04.1             |  | RF-04.1: Como administrador, quiero poder ver un listado de todos los cierres de caja realizados, ordenados del más reciente al más antiguo para análisis histórico.   | 4 | Se requiere crear el CRUD y ordenar los resultados.   |
| SHOULD              |  |  |   |   |
| RF-04.2             |  | RF-04.2: Como administrador, quiero que en mi reporte de cierres, al hacer clic sobre un cierre específico, se muestre una ventana con la siguiente información: fecha y hora del cierre, usuario que lo realizó, tipos de saldo involucrados, total de ingresos y egresos, saldo esperado (calculado automáticamente), saldo contado (ingresado por el usuario), diferencia total y observación (si aplica), para poder verificar y auditar cada cierre de forma clara y precisa.   | 3 | A través del sistema de ventanas se crea una ventana emergente que muestre información obtenida desde la base de datos.             |
| COULD               |  |  |   |   |
| RF-04.3             |  | RF-04.3: Como administrador, quiero un módulo que permita poder filtrar cierres de caja por rango de fechas, usuario, tipo de saldo, presencia de diferencias o palabras clave, permitiendo combinar estos filtros, para facilitar búsquedas precisas y eficientes.  | 2 | Es sencillo porque requiere queries con condiciones a la base de datos, está contenido en RF 0.4.1                                  |
| SHOULD              |  |  |   |   |
| RF-04.4             |  | RF-04.4: El sistema debe resaltar visualmente los cierres de caja que presenten una diferencia superior a un umbral configurable (por ejemplo, $\pm 20\%$ ) entre el total esperado, calculado según las transacciones registradas, y el monto ingresado por el usuario. Para ello, se utilizará un ícono de advertencia (como un triángulo amarillo con signo de exclamación) y un color destacado (por ejemplo, rojo). Se entiende por inconsistencias aquellas diferencias no justificadas, como faltantes, sobrantes, errores de digitación o registros incompletos. | 3 | Requiere aplicar validaciones al calcular las diferencias y aplicar resaltado visual en la interfaz según los valores configurados. |
| SHOULD              |  |  |   |   |
| RF-04.5             |  | RF-04.5: Como administrador, quiero exportar los reportes de cierre en formato PDF o Excel(.xls) para compartirlos o archivarlos.  | 3 | Requiere implementar librerías para exportación de datos en múltiples formatos, lo que añade funcionalidad útil para auditoría      |
| MUST<br>CU_3_serodr |  |  |   |   |

|                            |  |   |   |   |
|----------------------------|--|---|---|---|
| iguezor                    |  |   |   | y respaldo.   |
| RF-05.1                    |  | RF-05.1: El sistema debe proporcionar una interfaz de inicio de sesión compuesta por dos campos: uno para ingresar el número de documento del usuario registrado y otro para la contraseña. Además, debe incluir un botón para enviar la información y realizar la verificación de las credenciales. Esta funcionalidad tiene como objetivo permitir el acceso seguro de los usuarios al sistema. | 3 | Requiere un formulario de login básico y la validación de credenciales en base de datos.              |
| MUST<br>CU_2_apriet<br>ome |  |   |   |   |
| RF-05.2                    |  | RNF-05.2: El sistema debe cifrar la contraseña de cada nuevo usuario creado antes de almacenarlos en la base de datos, con el fin de evitar que usuarios malintencionados puedan obtener de manera directa esta información.  | 2 | Requiere aplicar algoritmos de cifrado antes de guardar la contraseña, lo que fortalece la seguridad. |
| MUST<br>CU_3_apriet<br>ome |  |   |   |   |
| RF-05.3                    |  | RNF-05.3: El sistema debe registrar la creación de un nuevo usuario en la base de datos con: fecha, hora, usuario que realizó la creación y tipo de usuario creado, de manera que se pueda llevar un registro de cada usuario que tiene algún tipo de acceso al sistema.  | 2 | Es sencillo, solo requiere guardar y almacenar los datos en la base de datos.                         |
| SHOULD                     |  |   |   |   |
| RF-05.4                    |  | RNF-05.4: El sistema debe mantener un historial de todos los inicios de sesión exitosos, incluyendo: usuario, fecha y hora, para facilitar las auditorías a la hora de conocer quién estaba activo en determinados tiempos.   | 3 | Requiere registrar automáticamente los datos de sesión en una tabla de logs.                          |
| COULD                      |  |   |   |   |
| RF-05.5                    |  | RNF-05.5: Si el tipo de operador de un usuario es modificado, el sistema debe registrar el cambio en el historial del sistema incluyendo: usuario afectado, tipo anterior de usuario (operador, cajero, administrador), tipo nuevo, fecha y hora del cambio, para que en la auditoría se pueda confirmar el historial de tipo de usuario.   | 2 | Requiere una función que registre los cambios de rol sobre cualquier usuario en la base de datos.     |
| SHOULD                     |  |   |   |   |