



CORPORACION UNIVERSITARIA IBEROAMERICANA

LIBRO ELECTRÓNICO PARA GESTIÓN DE CRÉDITOS DE CLIENTES EN TIENDAS DE BARRIO

PROYECTO DE SOFTWARE

MICHELL ESTEBAN ESCARRAGA RIVERA

ID:100170716

TATIANA CABRERA 25082025_C12_202534

Octubre de 2025



BOGOTA D.C

1. Contextualización de la necesidad

En los entornos comerciales de pequeña escala como tiendas de barrio y minimercados es habitual la venta a crédito (“fiado”), donde los clientes acumulan saldos que cancelan posteriormente. Históricamente, la administración del crédito se ha llevado en libretas o cuadernos físicos. Este mecanismo, aunque sencillo, introduce problemas de trazabilidad, pérdida de datos por deterioro o extravío, errores de cálculo manual y dificultad para obtener indicadores básicos de gestión (por ejemplo, morosidad, rotación de cartera o frecuencia de pago).

La digitalización de procesos en micro y pequeñas empresas (MYPES) ofrece una oportunidad para evolucionar del registro manual al control sistemático mediante aplicaciones web. Un libro electrónico de créditos puede centralizar los datos, proveer seguridad y respaldo, y facilitar reportes que permitan al comerciante tomar decisiones rápidas basadas en evidencia (por ejemplo, pausar créditos a clientes con mora crónica o incentivar pronto pago con descuentos).

2. Planteamiento del problema

Las tiendas de barrio suelen carecer de una solución digital estandarizada para administrar las cuentas por cobrar. Esto conlleva: (1) riesgo de pérdida de información; (2) inconsistencias en el cálculo de saldos; (3) inexistencia de historial y auditoría de movimientos; y (4) dificultad para comunicar a los clientes su estado de cuenta. La problemática se agrava cuando múltiples personas atienden la tienda, dado que cada una puede registrar de forma distinta.



Problema central: ¿Cómo puede una tienda de barrio gestionar de manera eficiente, segura y accesible las cuentas de crédito de sus clientes mediante una solución web sencilla e intuitiva, que reduzca errores, mejore la trazabilidad y ofrezca reportes oportunos?

3. Alcance del proyecto

3.1 Alcance funcional

El proyecto desarrollará el sistema web “Libro Electrónico de Créditos” con los módulos:

Clientes, Créditos, Reportes y Administración (roles y auditoría).

3.2 Restricciones

Sin pasarelas de pago, ni contabilidad NIIF en el MVP. Requiere Internet. Backups diarios.

3.3 Supuestos

Tienda con hasta 10.000 movimientos/año. Zona horaria América/Bogotá. Base de datos relacional.

3.4 Criterios de aceptación del proyecto

- CRUD de clientes y movimientos;
- Reportes exportables;
- Autenticación y RBAC;

- Manuales;
- $\geq 90\%$ pruebas aprobadas;
- 0 defectos críticos al cierre.

4. Estructura de Desglose del Trabajo (EDT)

| ID | Entregable | Descripción | Criterio de aceptación |
|-----|----------------------------|--------------------------|------------------------|
| 1.0 | Proyecto Libro Electrónico | Producto final integrado | MVP aprobado |
| 1.1 | Levantamiento y | Entrevistas, | Requisitos firmados |



| | | | |
|-----|---------------|----------------------------------|-----------------------|
| | Análisis | encuestas, dominio | |
| 1.2 | Diseño | Arquitectura, ERD, prototipos UI | Aprobación PO |
| 1.3 | Desarrollo | Clientes, Créditos, Reportes | HU1–HU12 listas |
| 1.4 | Pruebas | Plan, casos, UAT | ≥90% passed, 0 P0/P1 |
| 1.5 | Despliegue | Hosting, DB, backup | URL y respaldo diario |
| 1.6 | Documentación | Manual usuario y técnico | En repositorio |

5. Objetivos

5.1 Objetivo general

Desarrollar un sistema web de libro electrónico para la gestión de créditos en tiendas de barrio que mejore la trazabilidad, reduzca errores humanos y brinde reportes oportunos para la toma de decisiones.

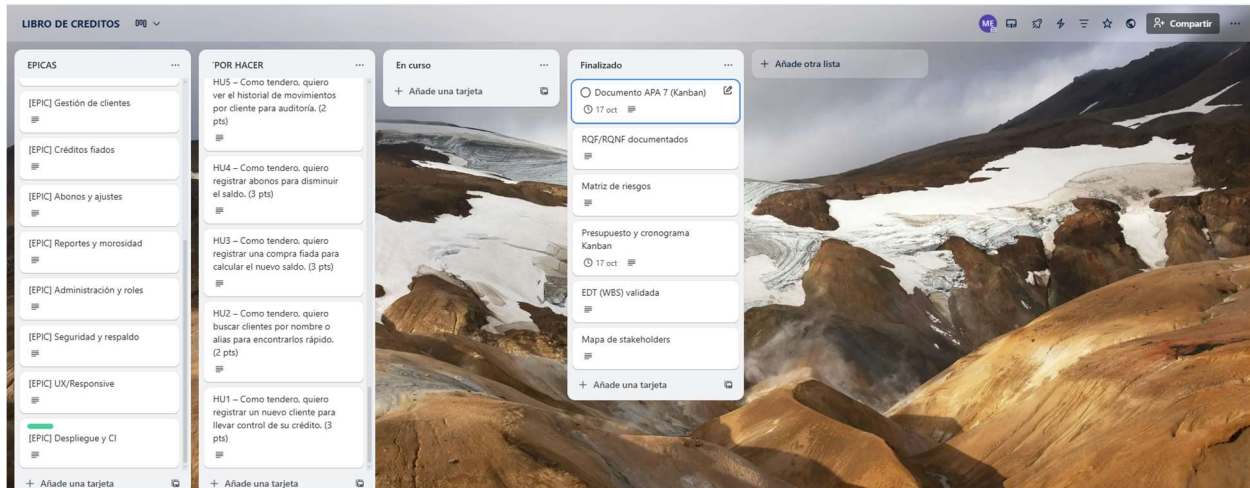
5.2 Objetivos específicos

- Analizar y documentar los requerimientos funcionales y no funcionales del sistema.
- Diseñar la arquitectura y el modelo de datos para clientes, créditos y abonos.
- Implementar los módulos de Clientes, Créditos y Reportes con control de acceso por roles.
- Construir reportes de cartera y morosidad con exportación a CSV.
- Validar el sistema con pruebas funcionales, de seguridad básica y UAT con tenderos.



6. Metodología ágil seleccionada (KANBAN)

Se adopta Kanban por su enfoque de flujo continuo, ideal para trabajo individual. No hay sprints ni lotes fijos; el trabajo fluye con límites WIP explícitos y políticas visibles.



<https://trello.com/b/JZf1gJXP/libro-de-creditos>

7. Justificación y respuesta a stakeholders

7.1 Corto plazo

Digitalizar la libreta, reducir errores y pérdidas.

7.2 Mediano plazo

Añadir reportes de comportamiento y alertas de morosidad.

7.3 Largo plazo

Mover a móvil e integrar inventario/facturación.

7.4 Respuesta a stakeholders

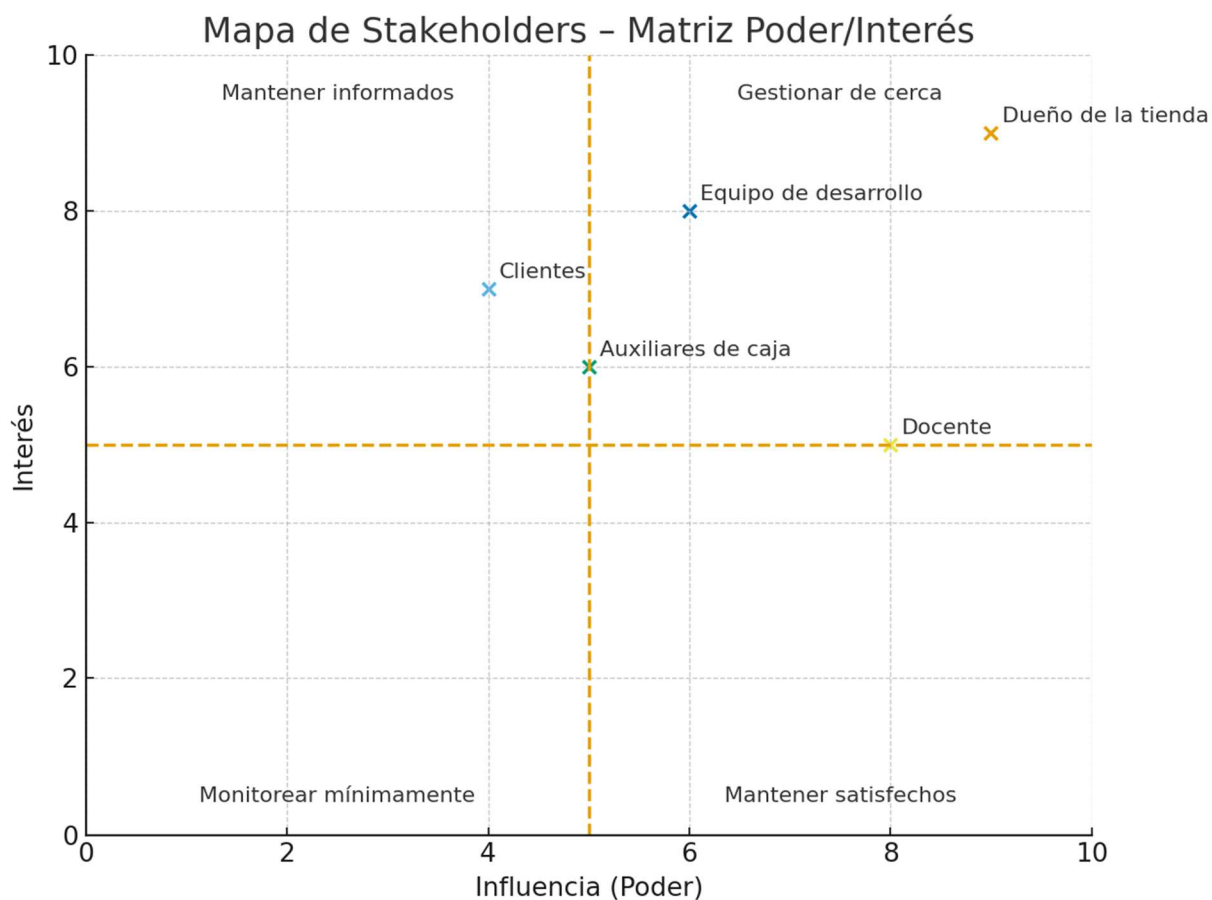
- Dueño: control de cartera y reportes.
- Clientes: transparencia de saldos y abonos.



- Docencia: evidencia de buenas prácticas de ingeniería.
- Equipo: aprendizaje con impacto real.

8. Mapa de stakeholders y clasificación

| Stakeholder | Rol | Interés | Influencia | Clasificación |
|----------------------|----------------------|---------|------------|---------------|
| Dueño de la tienda | Usuario principal | Alto | Alto | Clave |
| Clientes | Usuarios secundarios | Medio | Bajo | Apoyo |
| Auxiliares de caja | Operadores | Medio | Medio | Involucrados |
| Docente | Supervisor | Medio | Alto | Clave |
| Equipo de desarrollo | Técnico | Alto | Medio | Involucrado |



9. Matriz de riesgos

| Riesgo | Prob. | Impacto | Severidad | Estrategia | Mitigación | Responsable |
|------------------|-------|---------|-----------|------------|-------------------|---------------|
| Pérdida de datos | Media | Alta | Alta | Prevención | Backups diarios y | Líder técnico |



| | | | | | | |
|----------------------|-------|-------|-------|------------|----------------------------------|--------------|
| | | | | | versionado | |
| Brechas de seguridad | Baja | Alta | Media | Mitigar | RBAC, contraseñas fuertes, HTTPS | DevOps |
| Retraso de tareas | Media | Media | Media | Evitar | Plan semanal y dailies | Scrum Master |
| Caídas del hosting | Baja | Media | Baja | Transferir | Proveedor con SLA | PO |

Matriz de riesgos — Documento inicial

| Riesgo | Prob. | Impacto | Severidad | Estrategia | Mitigación | Responsable |
|----------------------------|-------|---------|-----------|------------|--|---------------|
| Pérdida de datos | Media | Alta | Alta | Prevención | Backups diarios y control de versiones | Líder técnico |
| Brechas de seguridad | Baja | Alta | Media | Mitigar | RBAC, contraseñas fuertes, HTTPS | DevOps |
| Retrasos del equipo | Media | Media | Medio | Evitar | Plan semanal y dailies efectivas | Scrum Master |
| Disponibilidad del hosting | Baja | Media | Baja | Transferir | Proveedor con SLA documentado | PO |

10. Presupuesto

| Concepto | Cantidad | Costo estimado (COP) |
|----------------------------|----------|----------------------|
| Dominio .com / .co (1 año) | 1 | \$60.000 |



| | | |
|--|-------|-------------|
| Hosting básico (12 meses) | 1 | \$180.000 |
| Certificado SSL (incluido) | 1 | \$0 |
| Mano de obra (3 devs x 40 h x \$7.500/h) | 120 h | \$900.000 |
| Conectividad y pruebas | — | \$50.000 |
| Contingencias 10% | — | \$119.000 |
| TOTAL | — | \$1.309.000 |

11. Repositorio del proyecto

Repositorio: <https://github.com/arleking-off/PROYECTO-SOFTWARE-IBERO>

12. Levantamiento de información (herramientas y conclusiones)

12.1 Objetivo del levantamiento

Identificar necesidades reales, dolores y expectativas de los tenderos respecto a la gestión de créditos ('fiar') para derivar requerimientos funcionales y no funcionales del sistema 'Libro Electrónico de Créditos', así como prioridades de producto y criterios de aceptación.

12.2 Alcance y sujetos de estudio

Unidad de análisis: tiendas de barrio/minimercados que registran fiado en libreta. Sujetos: propietarios y/o auxiliares de caja responsables del registro de deudas y abonos. Contexto: operación diaria en mostrador (horarios pico y valle), uso de libreta y/o Excel básico.

12.3 Metodología y técnicas aplicadas

Enfoque mixto (cualitativo—cuantitativo) con triangulación: 1) Entrevistas semiestructuradas (15–20 min); 2) Observación directa con lista de chequeo (30–45 min por punto); 3) Encuesta corta (Likert 1–5, 10 ítems); 4) Análisis de libretas físicas.

12.4 Instrumentos de recolección



Incluye: Guion de entrevista, Lista de chequeo de observación, Encuesta (Likert), y plantilla de consentimiento informado. Ver anexos adjuntos en este paquete.

12.5 Muestra y plan de trabajo

Muestreo intencional (6–10 comercios, 1–2 personas por comercio). Día 1–2: entrevistas; Día 3: observación; Día 4: encuesta y consolidación; Día 5: análisis y conclusiones.

12.6 Consideraciones éticas y de seguridad

Consentimiento informado; anonimato (T1..T10); no registrar datos sensibles de clientes; almacenamiento privado/cifrado de transcripciones; eliminación si se solicita.

12.7 Procesamiento y análisis

Cuantitativo: promedios, desviaciones, distribución Likert. Cualitativo: codificación abierta (rapidez, búsqueda, confiabilidad, reportes, móvil). Herramientas: Google Forms/Sheets o Excel, exportes CSV.

12.8 Resultados (piloto)

Rapidez: fiado 31 s, abono 26 s; errores: montos/fechas (4/8); 75% usa alias; saldo inmediato 4,7/5; morosidad solicitada 7/8; exportación 5/8; móvil 4,3/5; backups 4,6/5.

12.9 Trazabilidad hallazgo→requerimiento→módulo→HU→criterio

Se documenta la cadena completa para asegurar alineación: ver tabla en el documento base. Ejemplos: saldo inmediato→RF→Créditos→HU5→<1 s; búsqueda por alias→RF→Clientes→HU2→P90 < 500 ms; morosidad>30 días→RF→Reportes→HU6→CSV.

12.10 Conclusiones del levantamiento

Priorizar rapidez y saldo visible; búsqueda por alias; reporte de cartera con exportación; backups automáticos; UI móvil.

12.11 Implicaciones para el diseño

Flujo corto (un formulario), saldo/últimos movimientos visibles, búsqueda incremental, reporte de cartera con filtros, backups programados, UI responsiva.



12.12 Limitaciones

Muestra pequeña, horarios limitados, sesgo de auto-reporte; se recomienda validar con 3–5 tenderos adicionales.

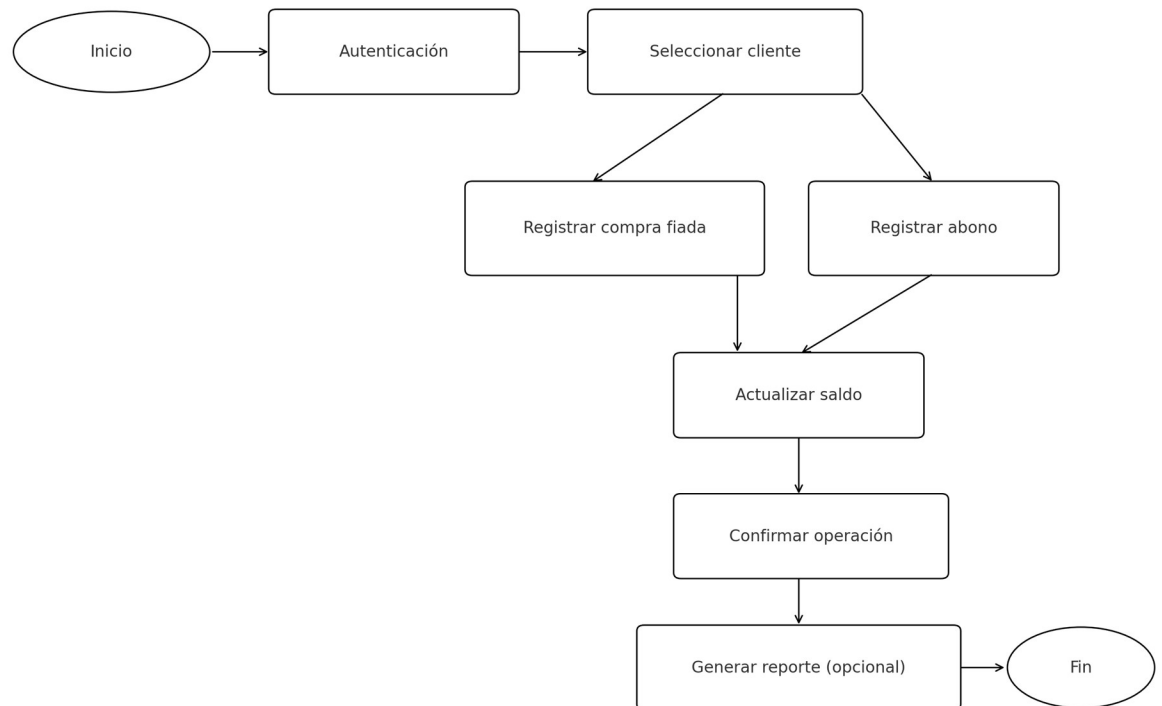
Entrevista – Tenderos (15–20 min)

Objetivo: comprender el proceso actual de fiado/abonos, dolores, expectativas y reportes críticos.

1. ¿Cómo registran hoy las ventas fiadas y los abonos? Muéstreme el proceso típico.
2. ¿Qué errores son más frecuentes (olvidos, montos, fechas, cliente equivocado)?
3. ¿Qué información necesita ver primero en su 'libreta' digital?
4. ¿Cada cuánto le piden los clientes su estado de cuenta? ¿Cómo se lo entrega?
5. ¿Qué haría que un sistema web le resulte fácil y rápido en hora pico?
6. ¿Qué reportes le ayudarían a cobrar mejor (morosidad, top de deudores, abonos del día)?
7. ¿Qué le preocupa de digitalizar (perder datos, no saber usarlo, costo)?
8. ¿Usa el celular en mostrador? ¿Qué tamaño/tipo de interfaz prefiere?
9. Si tuviera que elegir solo tres funciones para el MVP, ¿cuáles serían? ¿por qué?
10. ¿Hay alguien más que registra? ¿Cómo coordinan para evitar errores?

13. Diagrama de flujo de la solución

Inicio → Autenticación → Seleccionar cliente → (Compra fiada | Abono) → Actualizar saldo → Confirmar → Reporte → Fin.



14. Historias de Usuario (KANBAN) y criterios de aceptación

| ID | Historia de usuario | Criterios de aceptación (Gherkin) |
|-----|--|---|
| HU1 | Como tendero, quiero registrar un nuevo cliente para llevar control de su crédito. | Dado que ingreso a 'Clientes', cuando completo nombre y contacto, entonces se crea y confirma. |
| HU2 | Como tendero, quiero buscar clientes por nombre o alias para encontrarlos rápido. | Dado un texto de búsqueda, cuando presiono 'Buscar', entonces se listan coincidencias en <500 ms. |
| HU3 | Como tendero, quiero | Dado un cliente activo, |



| | | |
|------|---|--|
| | registrar una compra fiada para calcular el nuevo saldo. | cuando registro el monto fiado, entonces el sistema suma al saldo y guarda. |
| HU4 | Como tendero, quiero registrar abonos para disminuir el saldo. | Dado un saldo pendiente, cuando registro un abono válido, entonces el saldo disminuye y se audita. |
| HU5 | Como tendero, quiero ver historial de movimientos por cliente para auditoría. | Dado un cliente, cuando abro 'Historial', entonces veo lista por fecha con totales. |
| HU6 | Como dueño, quiero reportes de saldos y morosidad para priorizar cobro. | Dado 'Reportes', cuando elijo 'Cartera', entonces obtengo saldos por cliente y exportación CSV. |
| HU7 | Como administrador, quiero roles y permisos para proteger operaciones críticas. | Dado 'Administración', cuando asigno rol 'auxiliar', entonces se restringe eliminar movimientos. |
| HU8 | Como tendero, quiero editar datos de cliente para mantenerlos actualizados. | Dado un cliente, cuando edito y guardo, entonces se versiona el cambio y queda auditado. |
| HU9 | Como tendero, quiero anular un movimiento por error para corregir saldos. | Dado un movimiento reciente, cuando lo anulo con motivo, entonces se crea un reverso controlado. |
| HU10 | Como tendero, quiero ver alertas de morosidad para clientes con atraso. | Dado 'Reportes', cuando filtro por 'mora > 30 días', entonces obtengo la lista con |



| | | |
|------|--|--|
| | | contacto rápido. |
| HU11 | Como administrador, quiero backups diarios automáticos para proteger la información. | Dado la programación diaria, cuando se ejecuta el backup, entonces se almacena en ubicación segura por 7 días. |
| HU12 | Como usuario, quiero una interfaz responsiva para usarla en el celular. | Dado el acceso móvil, cuando abro la app, entonces los componentes se adaptan a pantalla. |

15. Definición de RQF y RQNF

15.1 Requerimientos funcionales (RQF)

| ID | Descripción | Prioridad (MoSCoW) | Criterio de aceptación |
|-------|----------------------------|--------------------|--|
| RF-01 | CRUD de clientes | Alta | Se gestiona ciclo de vida completo con validaciones. |
| RF-02 | Registro de compras fiadas | Alta | Monto, fecha, usuario; afecta saldo. |
| RF-03 | Registro de abonos y notas | Alta | Abono válido, descuento automático y auditoría. |
| RF-04 | Historial por cliente | Media | Filtro por fechas y exportación CSV. |
| RF-05 | Reportes de cartera | Alta | SalDOS, morosidad, totales. |
| RF-06 | Gestión de usuarios y | Alta | RBAC: propietario y |



| | | | |
|--|-------|--|-----------|
| | roles | | auxiliar. |
|--|-------|--|-----------|

15.2 Requerimientos no funcionales (RQNF)

| ID | Atributo de calidad | Especificación | Verificación |
|--------|---------------------|---------------------------|------------------------|
| RNF-01 | Rendimiento | Búsqueda < 500 ms (P90) | Pruebas de rendimiento |
| RNF-02 | Disponibilidad | SLA 99,5% mensual | Monitoreo |
| RNF-03 | Seguridad | TLS, hash, RBAC y logs | Pruebas/pentest básico |
| RNF-04 | Usabilidad | UI responsiva y accesible | Pruebas con usuarios |
| RNF-05 | Mantenibilidad | Cobertura $\geq 70\%$ | CI con reportes |
| RNF-06 | Portabilidad | Contenedores Docker | Guía de despliegue |

16. Arquitectura propuesta y modelo de datos

Stack sugerido: Frontend (bootstrap), Backend (Node.js + PHP + laravel), Base de datos

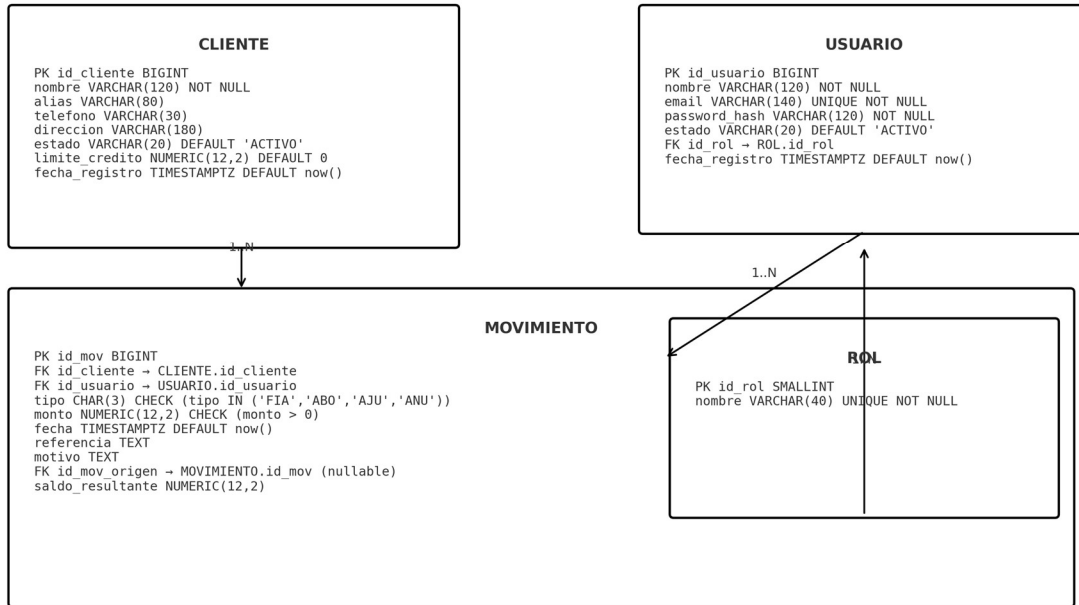
(MYSQL). Control de versiones con Git; CI opcional con GitHub Actions. Modelo ER: Cliente(id,

nombre, alias, contacto, estado); Movimiento(id, cliente_id, tipo, monto, fecha, usuario_id,

referencia); Usuario(id, nombre, rol). Vista saldo_por_cliente.



Modelo Entidad-Relación - Libro Electrónico de Créditos



Notas: FIA=Fiado, ABO=Abono, AJU=Ajuste, ANU=Anulación (reverso). 'saldo_resultante' es útil para auditoría rápida; el saldo por cliente se deriva en una VISTA.

17. Plan de pruebas

| ID Prueba | Caso | Traza (RF) | Entrada | Resultado esperado |
|-----------|------------------------|------------|------------------|--------------------------|
| PF-01 | Crear cliente válido | RF-01 | Datos correctos | Cliente creado y listado |
| PF-02 | Registrar compra fiada | RF-02 | Monto > 0 | Saldo aumenta |
| PF-03 | Registrar abono | RF-03 | Abono ≤ saldo | Saldo disminuye |
| PF-04 | Reporte de | RF-05 | Datos existentes | CSV con totales |



| | | | | |
|-------|-----------------|-------|--------------|------------------------|
| | cartera | | | |
| PF-05 | Permisos de rol | RF-06 | Rol auxiliar | No elimina movimientos |

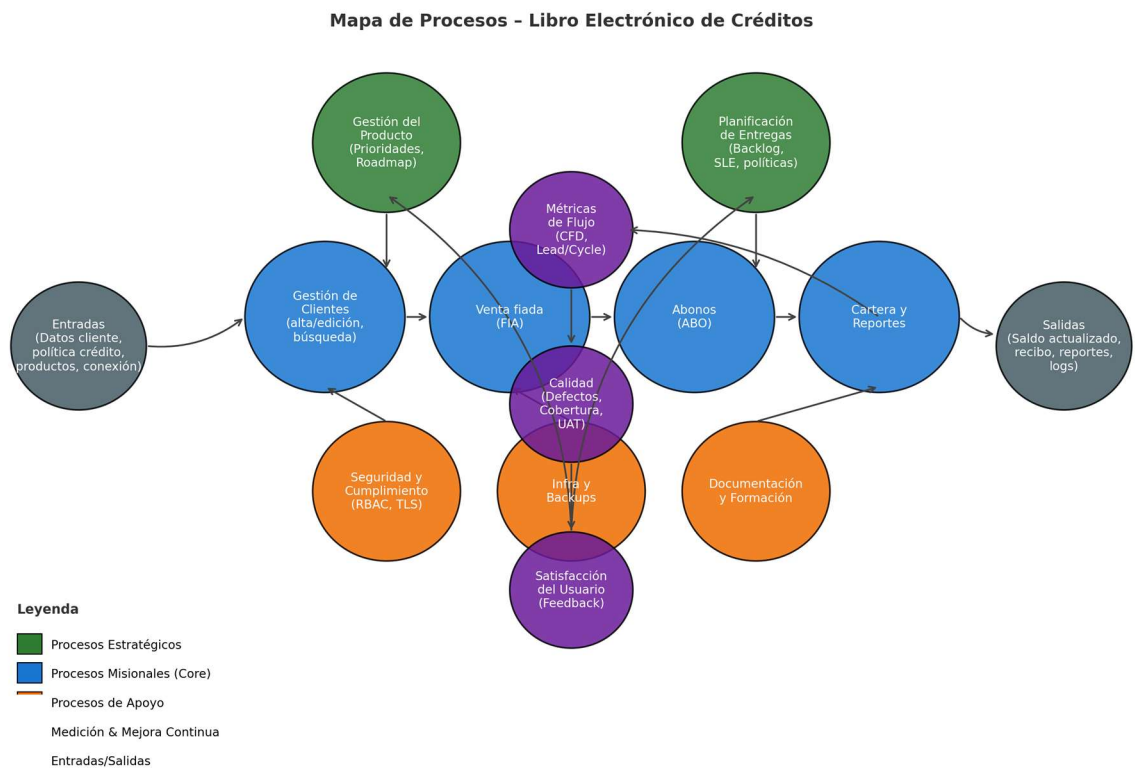
18. Plan de comunicación y cronograma

actualización del tablero; issues y PR en el repositorio; chat del equipo y correo. Cronograma 8 semanas: S1 (Análisis), S2 (Diseño), S3 (Desarrollo MVP), S4 (Pruebas y despliegue).

19. Conclusiones

La propuesta resuelve la problemática de gestión de créditos con un enfoque ágil, priorizando facilidad de uso, trazabilidad y seguridad básica, y habilitando decisiones con reportes de cartera.

20. MAPA DE PROCESOS





SIPOC – Proceso principal (Venta fiada / Abonos)

| S (Suppliers / Proveedores) | I (Inputs / Entradas) | P (Process / Proceso) | O (Outputs / Salidas) | C (Customers / Clientes) |
|--|---|--|--|---|
| Tendero (dueño/aux.) Cliente Proveedor de hosting/DB | Datos del cliente Política de crédito Productos y montos Conexión y credenciales | 1) Autenticación 2) Seleccionar cliente 3) Registrar FIA/ABO 4) Actualizar saldo 5) Emitir recibo 6) (Opcional) Reporte | Saldo actualizado Recibo de operación Reporte de cartera Logs/Auditoría | Tendero (dueño/aux.) Cliente (recibo) Docente/stakeholders (reportes) |

Bibliografía

Pressman, R. S., & Maxim, B. R. (2021). Ingeniería del software: Un enfoque práctico (9.^a ed.). McGraw-Hill.

Sommerville, I. (2020). Software Engineering (10th ed.). Pearson.

Schwaber, K., & Sutherland, J. (2020). The Scrum Guide. Scrum.org.

PMI. (2021). Guía del PMBOK (7.^a ed.). Project Management Institute.

Cohn, M. (2004). User Stories Applied: For Agile Software Development. Addison-Wesley.

ISO/IEC 25010:2011. Systems and software engineering — Quality Requirements and Evaluation (SQuaRE).

IEEE 29148-2011. Systems and Software Engineering — Requirements Engineering.