

---

# **Especificación de requisitos de software**

**Para**

## **Restaurant Chain**

**Preparado por**

**Luis Daniel Arias Marrugo**

**Michel Castellano Severiche**

**Brandold Vega Pérez**

**Andrés Felipe García Sosa**

**Universidad Tecnológica de Bolívar**

**25-11-2024**

## Tabla de contenido

<b>1. Introducción.....</b>	<b>3</b>
<b>2. Descripción general .....</b>	<b>5</b>
2.1 Características del producto .....	5
2.2 Clases de usuarios y características .....	6
2.3 Limitaciones de diseño y aplicación .....	6
2.4 Documentación del Usuario .....	8
2.5 Supuestos y dependencias .....	8
<b>3. Características del sistema .....</b>	<b>9</b>
3.1 Características de Sistema .....	9
3.1.1 Descripción y prioridad.....	9
3.1.2 Secuencias de estímulo/respuesta.....	10
3.1.3 Requisitos funcionales .....	10
3.2 Requisitos de la interfaz .....	12
3.3 Interfaces de Usuario.....	12
3.4 Interfaces de hardware .....	12
3.5 Interfaces de software.....	12
3.6 Interfaces de Comunicación .....	12
<b>4. Otros requisitos No funcionales .....</b>	<b>13</b>
4.1 Requerimientos de desempeño .....	13
4.2 Requisitos de seguridad.....	13
4.3 Atributos de calidad del software .....	13
<b>5. Otros Requerimientos .....</b>	<b>15</b>
5.1 Requerimientos del sistema .....	15
5.2 Requerimientos del usuario .....	15
<b>6. Casos de uso .....</b>	<b>17</b>

## Tabla de Diagramas

DIAGRAMA I DE PROCESO DE NEGOCIOS.....	23
DIAGRAMA II DE CLASES .....	24
DIAGRAMA III DE CASOS DE USO .....	25
DIAGRAMA IV ACTIVIDADES GESTIÓN DE PEDIDOS .....	1
DIAGRAMA V SECUENCIA GESTIÓN DE PEDIDOS .....	2
DIAGRAMA VI MAQUINA DE ESTADOS GESTIÓN DE PEDIDOS .....	1
DIAGRAMA VII ACTIVIDADES GESTIÓN DE INVENTARIO .....	1
DIAGRAMA VIII SECUENCIA GESTIÓN DE INVENTARIO.....	2
DIAGRAMA IX MAQUINA DE ESTADO GESTIÓN DE INVENTARIO .....	1
DIAGRAMA X ACTIVIDADES PAGOS.....	1
DIAGRAMA XI SECUENCIA PAGOS .....	1
DIAGRAMA XII MAQUINA ESTADO DE PAGOS.....	1
DIAGRAMA XIII ACTIVIDADES CONFIGURACIÓN DEL SISTEMA .....	1
DIAGRAMA XIV SECUENCIA CONFIGURACIÓN DEL SISTEMA.....	2
DIAGRAMA XV MAQUINA DE ESTADO CONFIGURACIÓN DEL SISTEMA .....	3
DIAGRAMA XVI ACTIVIDADES GESTIÓN API .....	4
DIAGRAMA XVII SECUENCIA GESTIÓN API.....	5
DIAGRAMA XVIII MAQUINA DE ESTADO GESTIÓN API .....	6
DIAGRAMA XIX ACTIVIDADES RECOMENDACIONES .....	7
DIAGRAMA XX SECUENCIA RECOMENDACIONES.....	8
DIAGRAMA XXI MAQUINA DE ESTADO RECOMENDACIONES .....	9

## Historial de revisiones

Nombre	Fecha	Motivo de los cambios	Versión
Descripción del caso	27-09-24	Inicio para la realización y descripción del caso escogido	1.0
Adicción de requerimientos funcionales	28-09-24	Descripción y adicción de los requerimientos funcionales para el caso escogido	2.0
Inicio con la descripción del general	29-09-24	Inicio y culminación del inciso 2 (descripción general)	3.0
Adicción de requerimientos no funcionales, del sistema y usuario	29-09-24	Descripción y adicción de los requerimientos para el caso escogido	4.0
Inclusión diagramas de clase, proceso de negocio y casos de uso	9-10-24	Adicción del diagrama de clases para el caso de estudio	5.0
Inclusión diagramas caso de uso gestión de pedidos	10-10-24	Inclusión diagramas de secuencia, actividades y máquina de estado para el caso gestión de pedidos	6.0
Inclusión diagramas para los casos de uso restantes	11-10-24	Inclusión diagramas de secuencia, actividades y máquina de estado para todos los casos de uso	7.0
Descripción de los anexos requisitos interfaz	24-11-24	Descripción de los anexos requisitos interfaz	8.0
Anexo supuestos y dependencias	24-11-24	Anexo supuestos y dependencias	9.0

# 1. Introducción

El software especificado es una plataforma integral basada en microservicios para una cadena de restaurantes que actualmente opera con sistemas fragmentados y heredados. Este nuevo sistema tiene como objetivo unificar y optimizar la operación de los restaurantes mediante la integración de funciones críticas, como el punto de venta, los programas de fidelización, las aplicaciones móviles y la gestión del inventario. Al descomponer los sistemas monolíticos existentes en microservicios desacoplados, se facilita la interoperabilidad entre los diferentes canales y se mejora la capacidad de la empresa para ofrecer experiencias omnicanal coherentes y personalizadas.

El propósito principal es proporcionar un software con una solución robusta y escalable que permita a la cadena de restaurante centralizar y compartir los datos en tiempo real a través de diferentes canales, como puntos físicos, aplicaciones móviles y programas de fidelización. Permitiendo así a la empresa mejorar la experiencia del cliente para un mayor prestigio y calidad del software.

Algunos beneficios que brindara este software, encontramos:

- Experiencia Omnicanal Consistente la cual, al integrar los diferentes sistemas, se garantiza que las promociones, pedidos y programas de fidelización ofrezcan una experiencia uniforme en todos los puntos de contacto con el cliente.
- Personalización de Interacciones contando La capacidad de compartir y analizar datos entre canales permite ofrecer interacciones personalizadas basadas en el comportamiento y las preferencias del cliente.
- Optimización operacional dejando la centralización y estandarización de datos que permite una mejor coordinación del inventario, lo que reduce costos operativos y mejora la eficiencia.
- Escalabilidad y flexibilidad en la arquitectura basada en microservicios, lo que permite a la empresa agregar nuevas funcionalidades y escalar el sistema sin las complejidades asociadas a la replicación de cambios en sistemas monolíticos.

Así mismo, objetivos y metas relevantes como:

- Integración de Sistemas, lo que consolidaría los sistemas de punto de venta, programas de fidelización, aplicaciones móviles y gestión de inventario en una plataforma unificada basada en microservicios.
- Mejora de la Experiencia del Cliente, al Implementar un sistema que permita interacciones omnicanal fluidas y consistentes para mejorar la satisfacción y lealtad del cliente.

- Optimización de Procesos, lo que reduciría los gastos generales y mejorar la eficiencia operacional mediante la centralización de datos y la automatización de procesos clave.
- Apalancamiento de Análisis Avanzados, al utilizar una plataforma de análisis en la nube para procesar y analizar los datos compartidos, permitiendo optimizar estrategias de marketing, gestión de inventario y toma de decisiones.

## 2. Descripción general

En un entorno empresarial cada vez más competitivo y digitalizado, este software se desarrolla como una evolución estratégica dentro de las cadenas de restaurante, con el objetivo de superar las limitaciones de sus sistemas actuales, que incluyen puntos de venta heredados, programas de fidelización independientes, aplicaciones móviles aisladas y una gestión ineficiente de los gastos operativos. Estos sistemas fragmentados han generado silos de información que impiden a la empresa obtener una visión integral de sus operaciones y de sus clientes, lo que limita la capacidad de personalizar las interacciones y optimizar los procesos de negocio.

El producto especificado en este SRS es una plataforma de microservicios que reemplaza a los sistemas monolíticos existentes, integrando funciones críticas como la gestión de pedidos, programas de fidelización, inventarios y análisis de datos en un entorno unificado. Este nuevo sistema no solo sustituye a los sistemas existentes, sino que también se posiciona como una solución escalable y flexible, diseñada para soportar la expansión futura de la cadena de restaurantes en términos de capacidades tecnológicas y de mercado.

### 2.1 Características del producto

El sistema propuesto para la cadena de restaurantes tiene como objetivo modernizar y optimizar las operaciones mediante la implementación de una arquitectura de microservicios y la integración de diversos sistemas. A continuación, se presenta un resumen de alto nivel de las principales características del producto:

1. **Arquitectura de Microservicios:** Descomposición de sistemas monolíticos en microservicios específicos para capacidades del restaurante como pedidos, programas de lealtad y gestión de inventario.
2. **Gateway de API:** Un punto de acceso unificado para gestionar todas las APIs, facilitando el acceso a datos de clientes, ventas y menú.
3. **Integración Omnicanal:** Integración completa entre sistemas de POS, aplicaciones móviles y programas de lealtad para ofrecer una experiencia de usuario coherente y sin interrupciones.
4. **Plataforma de Análisis en la Nube:** Transmisión de datos a una plataforma de análisis en la nube para optimización mediante aprendizaje automático (ML).
5. **APIs de Datos:** Exposición de APIs para acceder a datos de clientes, ventas y menú, facilitando la integración y el acceso a la información.
6. **Seguridad y Autenticación:** Implementación de medidas de seguridad robustas, incluyendo encriptación de datos y autenticación multifactorial.
7. **Alta Disponibilidad y Escalabilidad:** Garantía de alta disponibilidad y capacidad de escalar horizontalmente para manejar incrementos en la carga de trabajo.
8. **Interfaz de Usuario Intuitiva:** Interfaces de usuario intuitivas y fáciles de usar para empleados y clientes, tanto en POS como en aplicaciones móviles.

9. **Mantenibilidad y Actualizaciones:** Facilidad de mantenimiento y actualización del sistema con un tiempo de inactividad mínimo.
10. **Personalización y Recomendaciones:** Capacidad de ofrecer recomendaciones personalizadas y promociones basadas en el historial de compras y preferencias del cliente.

Para más detalles sobre cada característica, consulte la sección 3.

## 2.2 Clases de usuarios y características

La cadena de restaurante está diseñada para clases de usuarios favorecidos y clases de usuarios menos importante, por tanto:

### Clases de Usuarios Favorecidas

- Administradores del Sistema: Alta prioridad debido a su rol crítico en la gestión y mantenimiento del sistema.
- Gerentes de Restaurante: Alta prioridad para asegurar la eficiencia operativa y la toma de decisiones informadas.
- Clientes: Alta prioridad para garantizar una experiencia de usuario satisfactoria y coherente.

### Clases de Usuarios Menos Importantes

- Personal de Caja (Cajeros): Media prioridad, ya que su interacción con el sistema es más transaccional.
- Analistas de Datos: Media prioridad, ya que su uso del sistema es más especializado.
- Desarrolladores: Media prioridad, ya que su interacción es más técnica y menos frecuente en comparación con otros usuarios.

## 2.3 Limitaciones de diseño y aplicación

1. Políticas Corporativas o Normativas:
  - 1.1. Cumplimiento con las políticas corporativas de la cadena de restaurantes y las normativas legales aplicables, como las regulaciones de protección de datos (GDPR, CCPA).
  - 1.2. Los desarrolladores deben asegurarse de que todas las funcionalidades y el manejo de datos cumplan con estas políticas y normativas.
2. Limitaciones de Hardware:
  - 2.1. Requisitos de Tiempo: El sistema debe ser capaz de procesar transacciones en tiempo real para no afectar la experiencia del cliente.
  - 2.2. Requisitos de Memoria: El sistema debe ser eficiente en el uso de memoria para funcionar en el hardware existente de los puntos de venta (POS).
3. Interfaces con Otras Aplicaciones:
  - 3.1. Integración con sistemas de POS existentes, aplicaciones móviles y programas de lealtad.
  - 3.2. Los desarrolladores deben asegurarse de que las APIs y microservicios sean compatibles con estos sistemas y puedan comunicarse eficazmente.
4. Tecnologías, Herramientas y Bases de Datos Específicas:

- 4.1. Uso de tecnologías específicas como contenedores Docker para microservicios y bases de datos relacionales o no relacionales, como PostgreSQL o MongoDB.
- 4.2. Los desarrolladores deben estar familiarizados con estas tecnologías y herramientas para implementar y mantener el sistema.
5. Operaciones Paralelas:
  - 5.1. El sistema debe soportar operaciones paralelas para manejar múltiples transacciones simultáneamente.
  - 5.2. Los desarrolladores deben diseñar el sistema para ser concurrente y escalable, utilizando técnicas como el procesamiento asíncrono y la gestión de hilos.
6. Requisitos de Lenguaje:
  - 6.1. Uso de lenguajes de programación específicos como, por ejemplo. Java, Python o Node.js para el desarrollo de microservicios.
  - 6.2. Los desarrolladores deben tener experiencia en estos lenguajes y seguir las mejores prácticas de programación.
7. Protocolos de Comunicación:
  - 7.1. Uso de protocolos de comunicación estándar como HTTP/HTTPS para APIs RESTful y gRPC para comunicación interna entre microservicios.
  - 7.2. Los desarrolladores deben asegurarse de que las comunicaciones sean seguras y eficientes.
8. Consideraciones de Seguridad:
  - 8.1. Implementación de medidas de seguridad robustas, incluyendo encriptación de datos, autenticación multifactor y gestión de acceso.
  - 8.2. Los desarrolladores deben seguir las mejores prácticas de seguridad y cumplir con las normativas de protección de datos.
9. Convenciones de Diseño o Normas de Programación:
  - 9.1. Adopción de convenciones de diseño y normas de programación establecidas por la organización, como el uso de patrones de diseño, principios SOLID y metodologías ágil
  - 9.2. Los desarrolladores deben seguir estas convenciones y normas para asegurar la mantenibilidad y calidad del código.
10. Responsabilidad del Mantenimiento:
  - 10.1. La organización del cliente será responsable del mantenimiento del software entregado.
  - 10.2. Los desarrolladores deben documentar adecuadamente el código y proporcionar guías de mantenimiento para facilitar la transición y el soporte continuo.



## 2.4 Documentación del Usuario

*Se proporcionará un manual de usuario para cada clase de usuario identificada (administradores, gerentes, personal de caja y clientes). Este manual incluirá:*

- *Guías de uso general: Instrucciones paso a paso sobre las funciones principales.*
- *Resolución de problemas comunes: Soluciones para errores frecuentes como fallos en pedidos o consultas de inventario.*
- *Materiales de referencia rápida: Diagramas y ejemplos de flujos de trabajo.*
- *Capacitaciones digitales: Videos y simuladores para empleados y clientes.*

## 2.5 Supuestos y dependencias

*Supuestos:*

- *Los puntos de venta (POS) existentes serán compatibles con el nuevo sistema.*
- *La conexión a internet estará disponible en todas las ubicaciones para garantizar el funcionamiento en tiempo real.*
- *Los empleados recibirán capacitación antes de usar el sistema.*

*Dependencias:*

- *Uso de Docker para la gestión de microservicios.*
- *Integración con plataformas de análisis en la nube.*
- *APIs externas para pagos y programas de fidelización.*

## 3. Características del sistema

### 3.1 Características de Sistema

1. Arquitectura de microservicios
2. Gateway de API
3. Integración omnicanal
4. Plataforma de análisis en la nube
5. APIs de datos
6. Seguridad y autenticación
7. Escalabilidad del sistema
8. Personalización y recomendaciones

#### 3.1.1 Descripción y prioridad

1. **Arquitectura de Microservicios**
  - **Descripción:** El sistema debe estar basado en una arquitectura de microservicios, donde cada servicio está orientado a una capacidad específica del restaurante.
  - **Prioridad:** Alta
  - **Beneficio:** 9
  - **Penalización:** 8
  - **Coste:** 7
  - **Riesgo:** 6
2. **Gateway de API**
  - **Descripción:** Un Gateway centralizado para gestionar todas las APIs, proporcionando un punto de acceso unificado para los datos de clientes, ventas y menú.
  - **Prioridad:** Alta
  - **Beneficio:** 8
  - **Penalización:** 7
  - **Coste:** 6
  - **Riesgo:** 5
3. **Integración Omnicanal**
  - **Descripción:** Integración completa entre los sistemas de POS, aplicaciones móviles y programas de lealtad para ofrecer una experiencia de usuario coherente y sin interrupciones.
  - **Prioridad:** Alta
  - **Beneficio:** 9
  - **Penalización:** 8
  - **Coste:** 7
  - **Riesgo:** 6
4. **Plataforma de Análisis en la Nube**
  - **Descripción:** Transmisión de datos a una plataforma de análisis en la nube para optimización mediante aprendizaje automático (ML).
  - **Prioridad:** Media
  - **Beneficio:** 7
  - **Penalización:** 6
  - **Coste:** 6
  - **Riesgo:** 5
5. **APIs de Datos**
  - **Descripción:** Exposición de APIs para acceder a datos de clientes, ventas y menú, facilitando la integración y el acceso a la información.

- **Prioridad:** Alta
  - **Beneficio:** 8
  - **Penalización:** 7
  - **Coste:** 5
  - **Riesgo:** 4
6. **Seguridad y Autenticación**
- **Descripción:** Implementación de medidas de seguridad robustas, incluyendo encriptación de datos y autenticación multifactorial.
  - **Prioridad:** Alta
  - **Beneficio:** 9
  - **Penalización:** 8
  - **Coste:** 6
  - **Riesgo:** 5
7. **Escalabilidad del sistema**
- **Descripción:** El sistema debe ser altamente disponible y capaz de escalar horizontalmente para manejar incrementos en la carga de trabajo.
  - **Prioridad:** Alta
  - **Beneficio:** 9
  - **Penalización:** 8
  - **Coste:** 7
  - **Riesgo:** 6
8. **Personalización y Recomendaciones**
- **Descripción:** Capacidad de ofrecer recomendaciones personalizadas y promociones basadas en el historial de compras y preferencias del cliente.
  - **Prioridad:** Alta
  - **Beneficio:** 9
  - **Penalización:** 8
  - **Coste:** 6
  - **Riesgo:** 5

### 3.1.2 Secuencias de estímulo/respuesta

*Estímulo: Un cliente realiza un pedido desde la aplicación.*

*Respuesta:*

- *El sistema verifica la disponibilidad del inventario.*
- *Genera una confirmación del pedido con el tiempo estimado de entrega.*
- *Notifica al cliente y actualiza el inventario en tiempo real.*
- *Si el pago falla, se solicita al cliente un nuevo método de pago.*

### 3.1.3 Requisitos funcionales

1. RF-1 Gestionar pedidos
  - Descripción: Permitir a los clientes realizar pedidos a través de diferentes canales (POS, aplicaciones móviles, sitio web).
  - Datos de entrada: Información del cliente, detalles del pedido (productos, cantidades), método de pago.
  - Datos de salida: Confirmación del pedido, tiempo estimado de entrega/preparación.
  - Excepciones: Fallar en el procesamiento del pago, productos no disponibles.
2. RF-2 Gestionar programas de lealtad

- Descripción: Permitir a los clientes acumular y canjear puntos de lealtad.
  - Datos de entrada: Información del cliente, transacción de compra.
  - Datos de salida: Saldo de puntos de lealtad, historial de transacciones.
  - Excepciones: Error en la acumulación de puntos, canjes no validos
3. RF-3 Gestionar precios y promociones
- Descripción: Administrar y aplicar precios y promociones a los productos.
  - Datos de entrada: detalles de producto, reglas de precio y promociones.
  - Datos de salida: precios actualizados, promociones aplicadas.
  - Excepciones: conflicto entre promociones, error en la actualización de precios.
4. RF-4 Exponer datos de clientes, menú y ventas vía APIs
- Descripción: proveer acceso a datos del cliente, menú y ventas a través de APIs.
  - Datos de entrada: solicitudes API (consulta de datos).
  - Datos de salida: datos de los clientes, menú y ventas en formato JSON.
  - Excepciones: error en la autenticación, datos no encontrados.
5. RF-5 Transmitir datos a una plataforma de análisis en la nube
- Descripción: enviar datos de las APIs a una plataforma de análisis en la nube para optimización.
  - Datos de entrada: datos del cliente, ventas, menú.
  - Datos de salida: datos transmitidos a la plataforma de análisis.
  - Excepciones: falla en la transmisión de datos, error en la conexión.

### 3.2 Requisitos de la interfaz

- Interfaz del usuario: Diseño intuitivo con navegación clara para clientes y empleados.
- Compatibilidad: Funcionalidad en dispositivos móviles, terminales POS y navegadores web.
- Accesibilidad: Opciones de idioma y soporte para usuarios con discapacidades.

### 3.3 Interfaces de Usuario

- Clientes: Aplicación móvil con opciones de pedidos, historial de compras y promociones.
- Cajeros: Interfaz POS para transacciones rápidas y consultas de inventario.
- Gerentes: Panel de control para la gestión de inventarios y reportes.

### 3.4 Interfaces de hardware

- Navegador de su preferencia
- Windows 10 en Adelante /
- macOS 15.1.1
- Android 10 en Adelante

### 3.5 Interfaces de software

- *FastAPI para la interacción con aplicaciones externas.*
- *Sistemas de gestión de bases de datos relacionales y no relacionales (MongoDB).*

### 3.6 Interfaces de Comunicación

- Internas: Uso de FastAPI para la comunicación entre microservicios.
- Externas: Protocolo HTTPS para la comunicación segura con aplicaciones de terceros (pagos, programas de lealtad).

## 4. Otros requisitos No funcionales

### 4.1 Requerimientos de desempeño

1. RNF-1 Garantizar disponibilidad
  - Descripción: el sistema debe encontrarse disponible en el 90% del tiempo para asegurar continuidad en el negocio.
  - Datos de entrada: solicitudes de acceso al sistema
  - Datos de salida: informe de disponibilidad.
  - Excepciones: interrupciones por mantenimiento o fallos técnicos.
2. RNF-2 Escalar microservicios
  - Descripción: permitir que los sistemas fragmentados se desacoplen para el manejo de múltiples servicios de forma simultánea.
  - Datos de entrada: datos que manejen los microservicios.
  - Datos de salida: respuesta del sistema ante los microservicios.
  - Excepciones: sobrecarga de solicitudes o limitación de recursos.
3. RNF-3 Garantizar respuesta de las APIs
  - Descripción: garantizar que las APIs respondan en menos de 200 ms para asegurar experiencia fluida.
  - Datos de entrada: solicitudes de las APIs.
  - Datos de salida: informe de respuesta de las APIs.
  - Excepciones: retrasos en la red o interrupciones por sobrecarga de solicitudes.
4. RNF-4 Facilitar mantenibilidad del sistema
  - Descripción: Facilitar que el sistema sea fácil de mantener y actualizar sin interrumpir el servicio.
  - Datos de entrada: Actualizaciones y parches de software.
  - Datos de salida: Sistema actualizado y funcional.
  - Excepciones: Tiempo de inactividad prolongado, errores en la actualización.

### 4.2 Requisitos de seguridad

1. RNF-4 Garantizar seguridad de datos
  - Descripción: Los datos deben ser transmitidos y almacenados de manera segura utilizando cifrado AES-256.
  - Datos de entrada: datos sensibles del cliente y transacciones.
  - Datos de salida: datos encriptados y autenticados.
  - Excepciones: brechas de seguridad, autenticaciones fallidas.

### 4.3 Atributos de calidad del software

- Disponibilidad: 99.9% de uptime garantizado mediante arquitectura escalable.

- Seguridad: Implementación de cifrado AES-256 y autenticación multifactorial.
- Rendimiento: Respuesta de APIs en menos de 200 ms bajo carga máxima.
- Mantenibilidad: Código modular documentado y adaptable a nuevas tecnologías.

## **5. Otros Requerimientos**

### **5.1 Requerimientos del sistema**

1. RS-1 implementar microservicios
  - Descripción: Implementar microservicios orientados a capacidades del restaurante como pedidos y programas de lealtad.
  - Datos de entrada: Solicitudes de servicio, datos de clientes y ventas.
  - Datos de salida: Respuestas de servicio, datos procesados.
  - Excepciones: Fallos en la comunicación entre microservicios.
2. RS2 - Gestionar APIs a través de un Gateway
  - Descripción: Gestionar todas las APIs a través de un Gateway centralizado para un acceso unificado.
  - Datos de entrada: Solicitudes de API.
  - Datos de salida: Respuestas de API.
  - Excepciones: Errores en el Gateway, fallos de autenticación.
3. RS3 - Transmitir datos a una plataforma de análisis en la nube
  - Descripción: Transmitir datos de ventas, inventario y clientes a una plataforma de análisis en la nube para optimización de datos.
  - Datos de entrada: Datos de ventas, inventario y clientes.
  - Datos de salida: Informes de análisis, recomendaciones.
  - Excepciones: Fallos en la transmisión de datos, errores en el análisis.

### **5.2 Requerimientos del usuario**

1. RU1 - Acceder a promociones integradas
  - Descripción: Permitir a los usuarios acceder a promociones integradas a través de POS, aplicaciones móviles y programas de lealtad.
  - Datos de entrada: Información de usuario, detalles de promociones.
  - Datos de salida: Confirmación de promociones aplicadas.
  - Excepciones: Promociones no disponibles, errores en la aplicación de promociones.
2. RU2 - Realizar pedidos a través de múltiples canales
  - Descripción: Permitir a los usuarios realizar pedidos a través de POS, aplicaciones móviles y programas de lealtad.
  - Datos de entrada: Detalles del pedido, información de pago.
  - Datos de salida: Confirmación del pedido, tiempos de entrega estimados.
  - Excepciones: Fallos en el procesamiento del pedido, errores en el pago.



3. RU3 - Consultar historial de compras y puntos de lealtad
  - Descripción: Permitir a los usuarios consultar su historial de compras y puntos de lealtad a través de cualquier canal.
  - Datos de entrada: Información de usuario.
  - Datos de salida: Historial de compras, saldo de puntos de lealtad.
  - Excepciones: Datos no encontrados, errores en la consulta.
  
4. RU4 - Recibir recomendaciones personalizadas
  - Descripción: Permitir a los usuarios recibir recomendaciones personalizadas basadas en su historial de compras y preferencias.
  - Datos de entrada: Historial de compras, preferencias del usuario.
  - Datos de salida: Recomendaciones de productos.
  - Excepciones: Datos incompletos o incorrectos, fallos en la personalización.

## 6. Casos de uso

<b>Use Case ID:</b>	1		
<b>Use Case Name:</b>	Gestion de pedidos		
<b>Created By:</b>		<b>Last Updated By:</b>	
<b>Date Created:</b>		<b>Date Last Updated:</b>	

<b>Actors:</b>	cliente cajero
<b>Description:</b>	Este caso de uso describe el proceso mediante el cual un cliente puede realizar un pedido a través de diversos canales, y cómo el sistema gestiona dicho pedido asegurando la disponibilidad de productos, la verificación de inventario y el procesamiento del pago.
<b>Trigger:</b>	El cliente inicia la acción de realizar un pedido, ya sea en persona, por medio de una aplicación móvil, o a través del sitio web de la cadena de restaurantes.
<b>Preconditions:</b>	El cliente debe existir Catálogo, producto(s) y artículos en inventario
<b>Postconditions:</b>	<ul style="list-style-type: none"> <li>El pedido es registrado en el sistema y se proporciona una confirmación al cliente.</li> <li>El sistema actualiza el inventario para reflejar los productos vendidos.</li> <li>Se informa al cliente sobre el tiempo estimado de preparación o entrega.</li> </ul>
<b>Normal Flow:</b>	<ol style="list-style-type: none"> <li>El cliente selecciona productos del catálogo.</li> <li>El cliente especifica la cantidad de cada producto.</li> <li>El cliente selecciona un método de pago.</li> <li>El sistema verifica la disponibilidad de los productos en el inventario.</li> <li>El sistema procesa el pago.</li> <li>El sistema confirma el pedido y muestra el tiempo estimado de entrega o preparación.</li> </ol>
<b>Alternative Flows:</b>	<ul style="list-style-type: none"> <li>Si el cliente no está registrado, el sistema le solicita crear una cuenta.</li> <li>Si algunos productos no están disponibles, el sistema muestra alternativas o la posibilidad de ajustar el pedido.</li> </ul>
<b>Exceptions:</b>	<ul style="list-style-type: none"> <li>Si hay un error en el procesamiento del pago, el sistema notifica al cliente y solicita un nuevo intento o método de pago alternativo.</li> <li>Si no hay productos disponibles, el sistema muestra un mensaje de error o alternativas.</li> </ul>
<b>Priority:</b>	Parte de las siguientes características del sistema general, incluyendo la experiencia omnicanal y la integración con el inventario y el sistema de pagos.

<b>Use Case ID:</b>	2		
<b>Use Case Name:</b>	Procesar pagos		
<b>Created By:</b>		<b>Last Updated By:</b>	
<b>Date Created:</b>		<b>Date Last Updated:</b>	

<b>Actors:</b>	Cliente cajero Sistema de pagos
<b>Description:</b>	permite al cliente realizar el pago de su pedido utilizando diferentes métodos de pago, como tarjetas de crédito/débito, pagos en efectivo, o servicios de pago en línea. El sistema valida el pago y actualiza el estado del pedido en función de la confirmación recibida.
<b>Trigger:</b>	El cliente confirma su pedido y selecciona un método de pago.
<b>Preconditions:</b>	El pedido debe existir Sistema de pago debe estar disponible
<b>Postconditions:</b>	<ul style="list-style-type: none"> <li>El pago es procesado exitosamente.</li> <li>Se actualiza el estado del pedido a "pagado".</li> <li>Se emite un recibo de pago al cliente.</li> </ul>
<b>Normal Flow:</b>	<ul style="list-style-type: none"> <li>El cliente selecciona el método de pago (tarjeta de crédito/débito, efectivo, pago en línea).</li> <li>El sistema presenta al cliente el importe total del pedido.</li> <li>El cliente ingresa los detalles de pago o entrega el efectivo.</li> <li>El sistema de pagos valida la transacción.</li> <li>El sistema confirma el pago exitoso.</li> <li>Se actualiza el estado del pedido a "pagado" y se genera un recibo.</li> </ul>
<b>Alternative Flows:</b>	<ul style="list-style-type: none"> <li>Si el cliente desea cambiar el método de pago, el sistema permite la selección de otro método antes de procesar el pago.</li> <li>Si el cliente está utilizando un sistema de pago en línea, el sistema redirige al cliente a la pasarela de pagos externa y, tras la confirmación, vuelve al sistema principal.</li> </ul>
<b>Exceptions:</b>	<ul style="list-style-type: none"> <li>Si el pago falla (por fondos insuficientes, tarjeta rechazada, o problemas técnicos), el sistema notifica al cliente y ofrece intentarlo nuevamente o cambiar el método de pago.</li> <li>Si el sistema de pagos no está disponible, se le informa al cliente y se ofrece una opción de pago alternativo o diferido.</li> </ul>
<b>Priority:</b>	Alta prioridad, ya que es crucial para completar el ciclo de venta y garantizar la actualización en tiempo real de los estados de los pedidos y el inventario.

<b>Use Case ID:</b>	3		
<b>Use Case Name:</b>	Gestionar inventario		
<b>Created By:</b>		<b>Last Updated By:</b>	
<b>Date Created:</b>		<b>Date Last Updated:</b>	

<b>Actors:</b>	Gerente de Restaurante Administrador del Sistema Cajero Analista de Datos
<b>Description:</b>	Este caso de uso permite la gestión de inventario en tiempo real, registrando entradas y salidas de productos, actualizando el stock disponible y generando alertas cuando se alcanza un umbral mínimo. El gerente puede realizar pedidos de reabastecimiento y el sistema se integra con otros módulos de la cadena de restaurantes para mantener los datos sincronizados.
<b>Trigger:</b>	La gestión de inventario se activa cuando el Gerente de Restaurante accede al sistema para revisar o actualizar los niveles de stock, o cuando el sistema detecta que el inventario ha alcanzado un umbral mínimo.
<b>Preconditions:</b>	<ul style="list-style-type: none"> <li>El usuario debe estar autenticado en el sistema y tener los permisos necesarios para gestionar el inventario.</li> <li>El inventario de productos debe estar previamente cargado en la base de datos.</li> </ul>
<b>Postconditions:</b>	<ul style="list-style-type: none"> <li>El sistema actualiza los niveles de inventario en la base de datos.</li> <li>Se generan alertas automáticas para el reabastecimiento de productos si se alcanza el umbral mínimo.</li> <li>El informe de inventario se actualiza para reflejar los cambios.</li> </ul>
<b>Normal Flow:</b>	<ul style="list-style-type: none"> <li>El Gerente de Restaurante accede al módulo de gestión de inventario.</li> <li>Verifica los niveles actuales de los productos.</li> <li>Introduce los cambios de cantidades en productos utilizados o recibidos.</li> <li>El sistema actualiza la base de datos con los nuevos niveles.</li> <li>Si algún producto cae por debajo del umbral mínimo, el sistema envía una alerta al Gerente de Restaurante.</li> <li>El Gerente genera un pedido de reabastecimiento desde el sistema si es necesario.</li> </ul>
<b>Alternative Flows:</b>	<ul style="list-style-type: none"> <li>El Cajero reporta una escasez de producto durante una transacción, generando una notificación para el Gerente de Restaurante.</li> <li>El Administrador del Sistema puede actualizar el umbral mínimo de productos o configurar notificaciones adicionales en caso de que sea necesario.</li> </ul>
<b>Exceptions:</b>	<ul style="list-style-type: none"> <li>Falla del sistema: Si el sistema no responde, los cambios en el inventario no se reflejan y se notifica al Gerente de Restaurante para que reintente la operación.</li> <li>Producto no registrado: Si el sistema no reconoce un producto, se envía una notificación al Administrador del Sistema para corregir el inventario.</li> </ul>
<b>Priority:</b>	Alta

<b>Use Case ID:</b>	4		
<b>Use Case Name:</b>	Configurar Sistema		
<b>Created By:</b>		<b>Last Updated By:</b>	
<b>Date Created:</b>		<b>Date Last Updated:</b>	

<b>Actors:</b>	Administrador del Sistema
<b>Description:</b>	Este caso de uso permite al Administrador del Sistema realizar configuraciones generales del sistema, como la actualización de parámetros, la gestión de usuarios, la definición de reglas de seguridad y la configuración de umbrales de alertas. Estas configuraciones son fundamentales para asegurar que el sistema funcione de manera óptima y cumpla con los requisitos operacionales.
<b>Trigger:</b>	El Administrador del Sistema accede al módulo de configuración para realizar cambios o ajustes en los parámetros del sistema.
<b>Preconditions:</b>	<ul style="list-style-type: none"> <li>El administrador debe estar autenticado y tener permisos de acceso al módulo de configuración.</li> <li>Los parámetros de configuración deben estar disponibles para su modificación.</li> </ul>
<b>Postconditions:</b>	<ul style="list-style-type: none"> <li>El sistema refleja los cambios de configuración en tiempo real.</li> <li>Los usuarios afectados reciben las nuevas configuraciones o permisos.</li> <li>Se generan informes de auditoría sobre los cambios realizados.</li> </ul>
<b>Normal Flow:</b>	<ol style="list-style-type: none"> <li>El Administrador del Sistema accede al módulo de configuración.</li> <li>Selecciona las áreas del sistema que desea ajustar (ej. seguridad, umbrales, usuarios).</li> <li>Realiza los cambios en los parámetros de configuración.</li> <li>El sistema aplica los cambios en tiempo real y actualiza las configuraciones.</li> <li>El sistema confirma al Administrador que los cambios han sido aplicados con éxito.</li> </ol>
<b>Alternative Flows:</b>	<ul style="list-style-type: none"> <li>El Administrador programa actualizaciones o cambios para que se apliquen fuera del horario operativo del restaurante.</li> </ul>
<b>Exceptions:</b>	<ul style="list-style-type: none"> <li>Error en la actualización: Si el sistema falla al aplicar un cambio, el Administrador recibe una notificación para corregir el problema y reintentar la operación.</li> <li>Parámetro inválido: Si se ingresa un valor fuera de los límites aceptables, el sistema rechaza la entrada y solicita corrección.</li> </ul>
<b>Priority:</b>	Alta

<b>Use Case ID:</b>	5		
<b>Use Case Name:</b>	Gestionar APIs		
<b>Created By:</b>		<b>Last Updated By:</b>	
<b>Date Created:</b>		<b>Date Last Updated:</b>	

<b>Actors:</b>	Administrador del Sistema
<b>Description:</b>	Este caso de uso permite al Administrador del Sistema gestionar las APIs del sistema. Esto incluye la creación, actualización y eliminación de APIs, así como la supervisión de su rendimiento. Además, el Administrador puede configurar permisos de acceso y asegurarse de que las APIs sean seguras y respondan dentro de los tiempos esperados.
<b>Trigger:</b>	El Administrador del Sistema accede al módulo de gestión de APIs para crear, modificar, eliminar o monitorear el rendimiento de las APIs.
<b>Preconditions:</b>	<ul style="list-style-type: none"> <li>• El Administrador debe estar autenticado y tener permisos para gestionar las APIs.</li> <li>• Las APIs deben estar integradas en el sistema de microservicios.</li> </ul>
<b>Postconditions:</b>	<ul style="list-style-type: none"> <li>• Las APIs están actualizadas o creadas según las configuraciones realizadas por el Administrador.</li> <li>• Se genera un informe de rendimiento y monitoreo de las APIs.</li> <li>• Las APIs no autorizadas o fallidas son deshabilitadas automáticamente.</li> </ul>
<b>Normal Flow:</b>	6. El Administrador del Sistema accede al módulo de gestión de APIs. 7. Selecciona si desea crear, modificar, eliminar o monitorear una API. 8. Realiza los ajustes necesarios en la configuración de la API. 9. El sistema actualiza la API y registra los cambios. 10. El sistema genera un informe sobre el estado de la API y su rendimiento.
<b>Alternative Flows:</b>	<ul style="list-style-type: none"> <li>• El Administrador puede desactivar temporalmente una API por mantenimiento o actualización.</li> <li>• El Administrador configura límites de uso para evitar sobrecargas en las APIs.</li> </ul>
<b>Exceptions:</b>	<ul style="list-style-type: none"> <li>• Error de conexión: Si una API falla en su comunicación con otros sistemas, el Administrador recibe una alerta para corregir el problema.</li> <li>• Error de autenticación: Si el acceso a una API no es autorizado, el sistema bloquea la solicitud y notifica al Administrador.</li> </ul>
<b>Priority:</b>	Alta

<b>Use Case ID:</b>	6		
<b>Use Case Name:</b>	Recomendaciones personalizadas		
<b>Created By:</b>		<b>Last Updated By:</b>	
<b>Date Created:</b>		<b>Date Last Updated:</b>	

<b>Actors:</b>	Administrador del Sistema
<b>Description:</b>	Este caso de uso permite al sistema generar recomendaciones personalizadas para el cliente en función de su historial de compras, preferencias y comportamiento dentro de los canales de interacción (punto de venta, aplicación móvil, sitio web). Las recomendaciones pueden incluir productos, promociones especiales y ofertas basadas en el perfil del cliente.
<b>Trigger:</b>	El cliente accede al sistema (a través de POS, aplicación móvil o sitio web) o finaliza una compra, lo que desencadena la generación de recomendaciones.
<b>Preconditions:</b>	<ul style="list-style-type: none"> <li>El cliente debe tener un historial de compras o preferencias registradas en el sistema.</li> <li>El sistema debe estar integrado con los datos del cliente y los productos disponibles.</li> </ul>
<b>Postconditions:</b>	<ul style="list-style-type: none"> <li>El cliente recibe recomendaciones personalizadas basadas en su comportamiento previo.</li> <li>Se actualizan las recomendaciones en tiempo real cuando se realiza una nueva compra o se registra una nueva preferencia.</li> </ul>
<b>Normal Flow:</b>	<ol style="list-style-type: none"> <li>El cliente inicia sesión o accede al sistema a través de un canal de interacción.</li> <li>El sistema analiza el historial de compras, las preferencias y el comportamiento del cliente.</li> <li>El sistema genera una lista de recomendaciones personalizadas que incluyen productos y ofertas relevantes.</li> <li>El cliente visualiza las recomendaciones y puede agregar productos sugeridos a su carrito de compras o acceder a las promociones.</li> </ol>
<b>Alternative Flows:</b>	<ul style="list-style-type: none"> <li>Si el cliente no tiene un historial de compras significativo, el sistema puede basarse en datos genéricos o en tendencias populares para ofrecer recomendaciones.</li> <li>El cliente puede ajustar manualmente sus preferencias dentro de la plataforma para recibir recomendaciones más precisas en el futuro.</li> </ul>
<b>Exceptions:</b>	<ul style="list-style-type: none"> <li>Si no hay suficientes datos para generar recomendaciones personalizadas (por ejemplo, para un nuevo cliente), el sistema muestra recomendaciones basadas en tendencias generales o productos destacados.</li> <li>Si hay un error en la generación de recomendaciones (por ejemplo, datos de compras incompletos), el sistema muestra un mensaje de error y ofrece la opción de intentar nuevamente o mostrar productos populares.</li> </ul>
<b>Priority:</b>	Media a alta, ya que las recomendaciones personalizadas son una funcionalidad clave para mejorar la experiencia del cliente, fomentar la lealtad y aumentar las ventas al sugerir productos y promociones relevantes.

## Apéndice B: Modelos de análisis

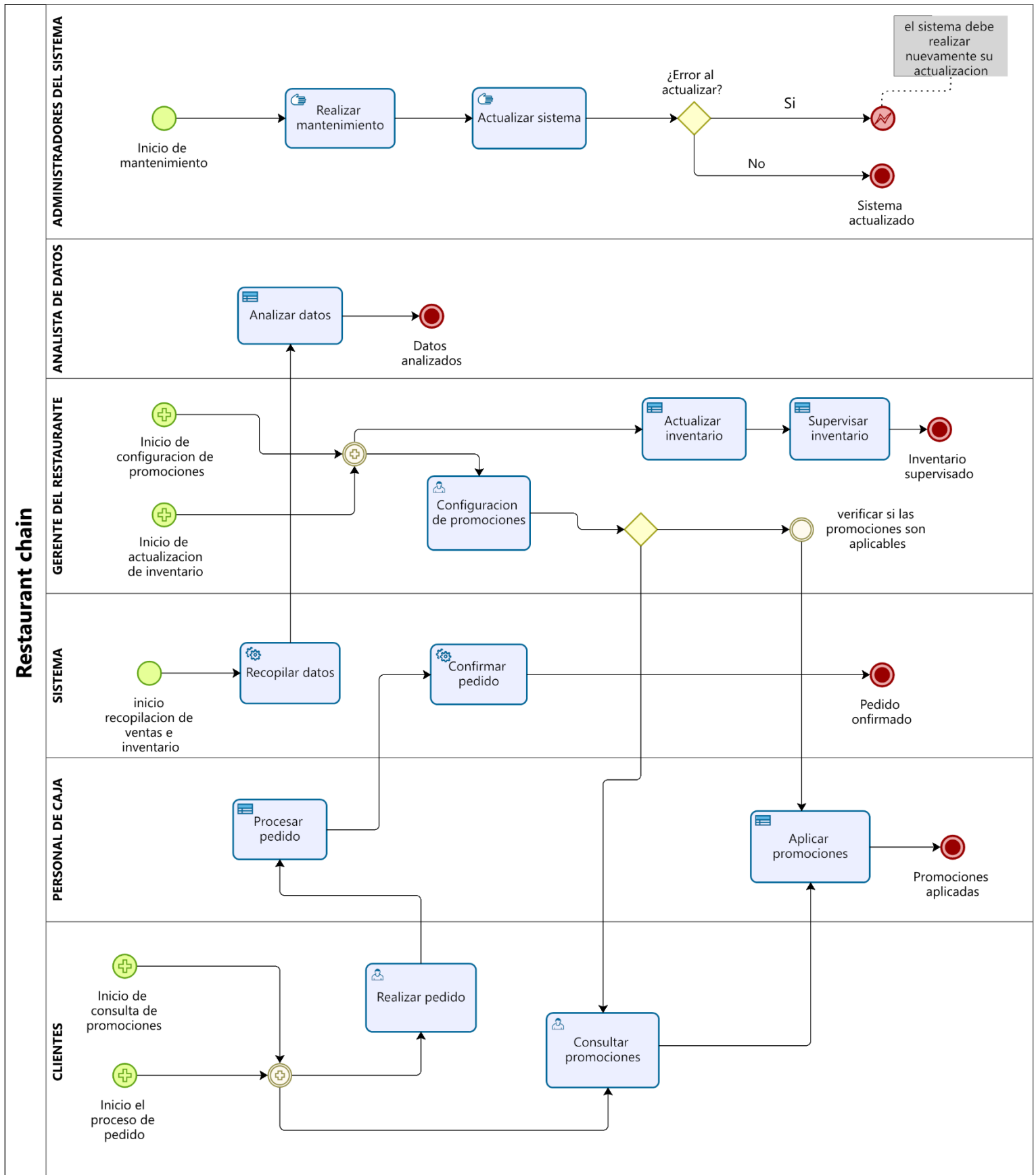


Diagrama I de Proceso de Negocios



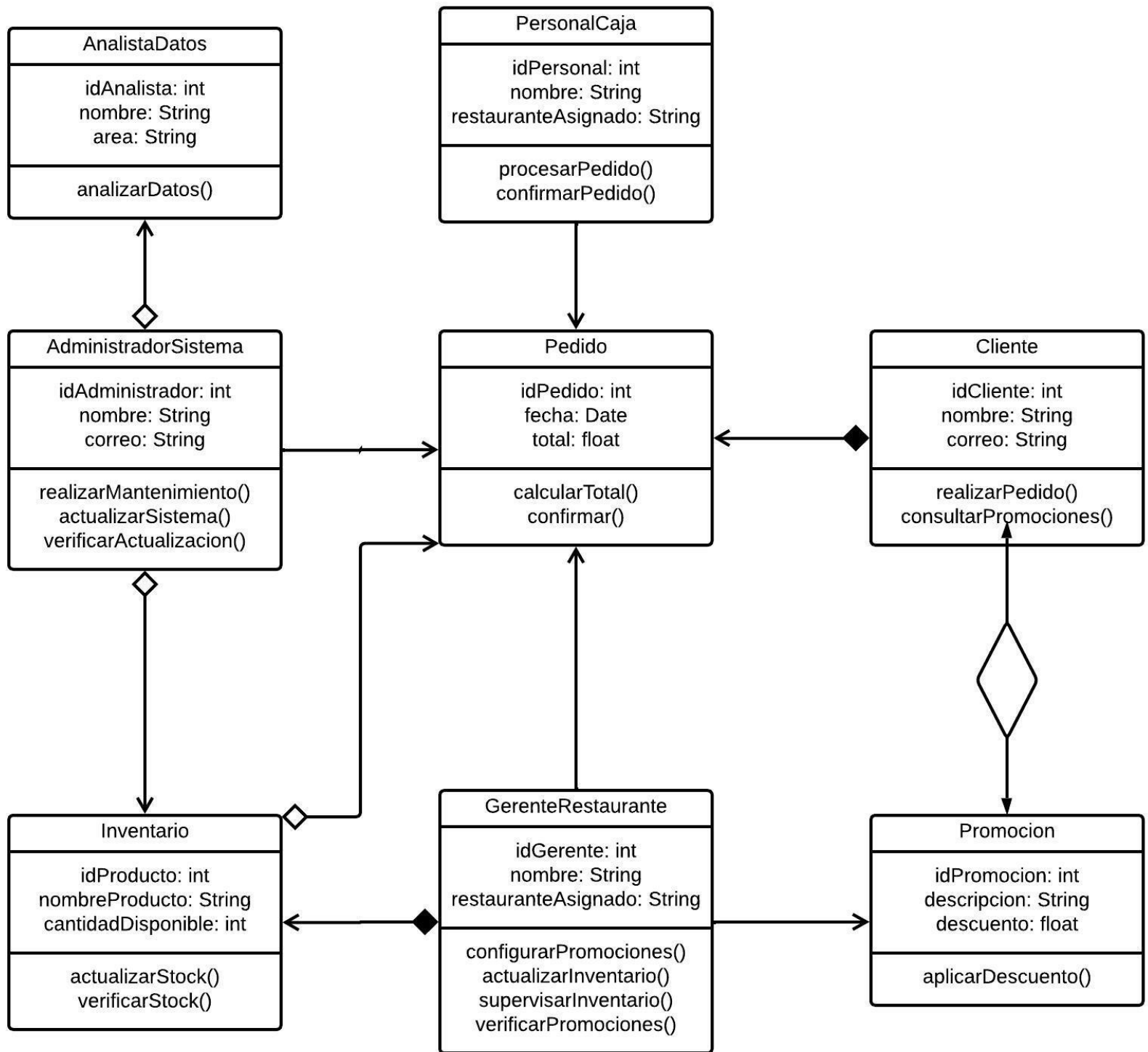


Diagrama II de Clases

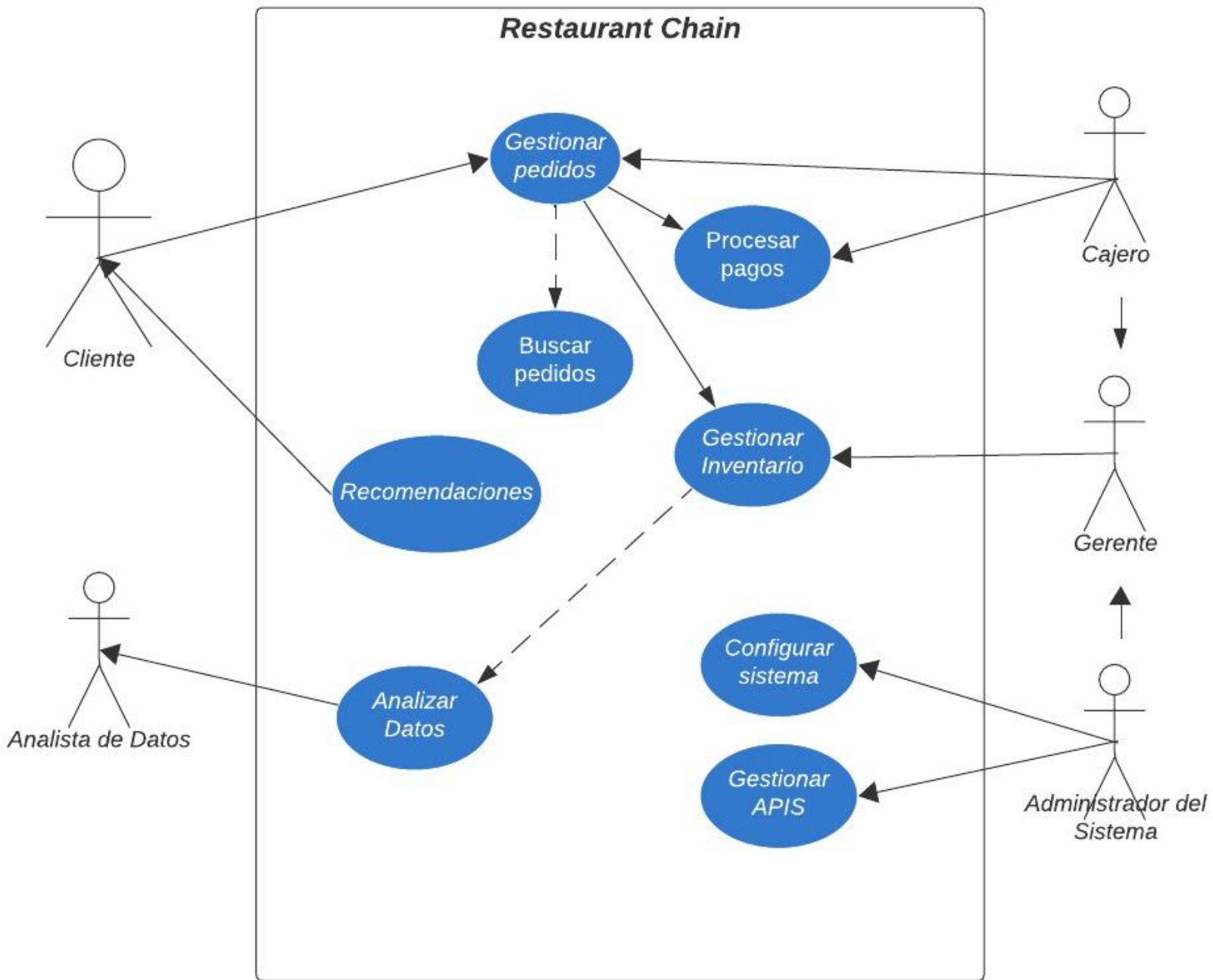


Diagrama III de Casos de Uso

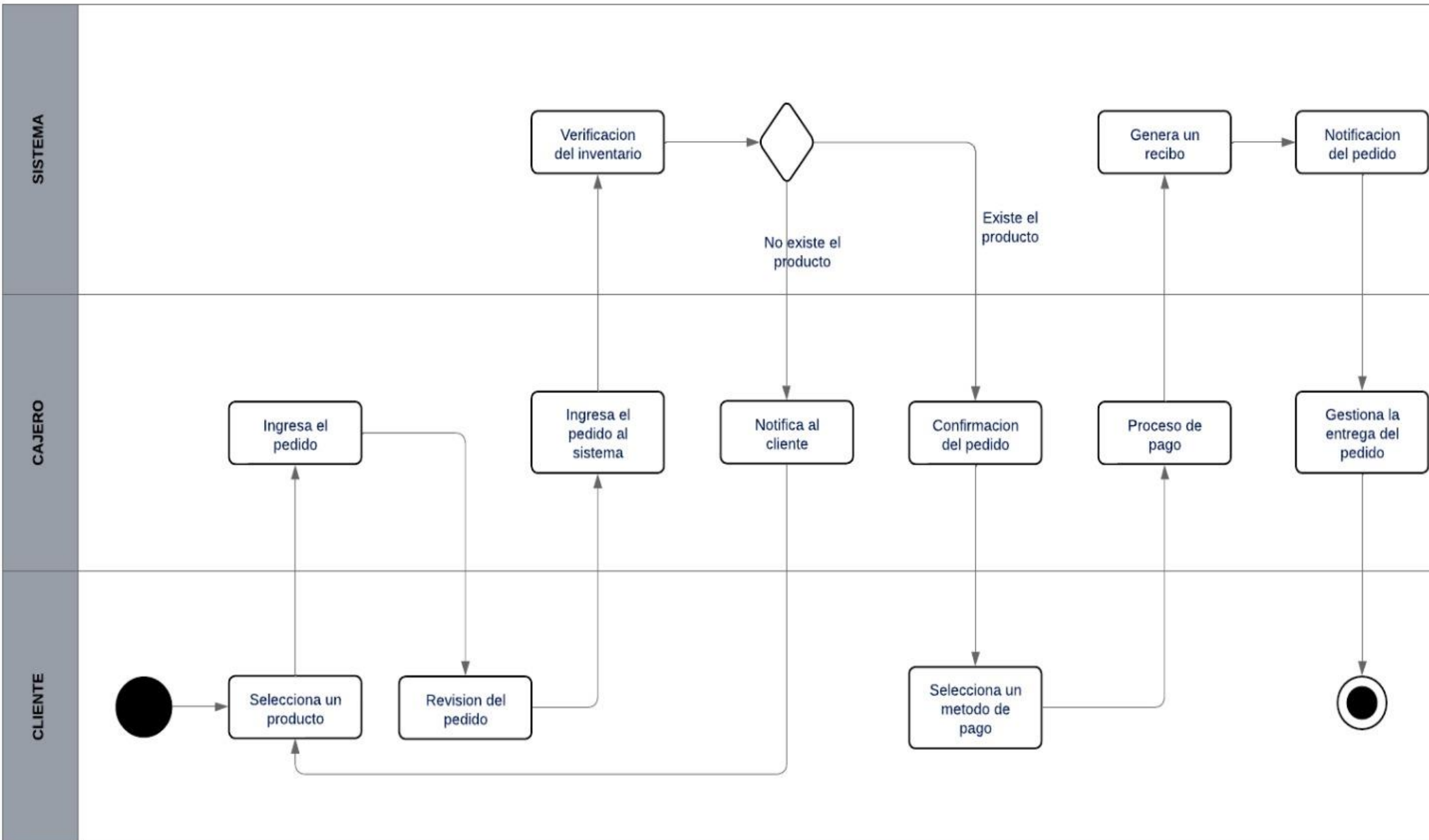


Diagrama IV Actividades Gestión de pedidos

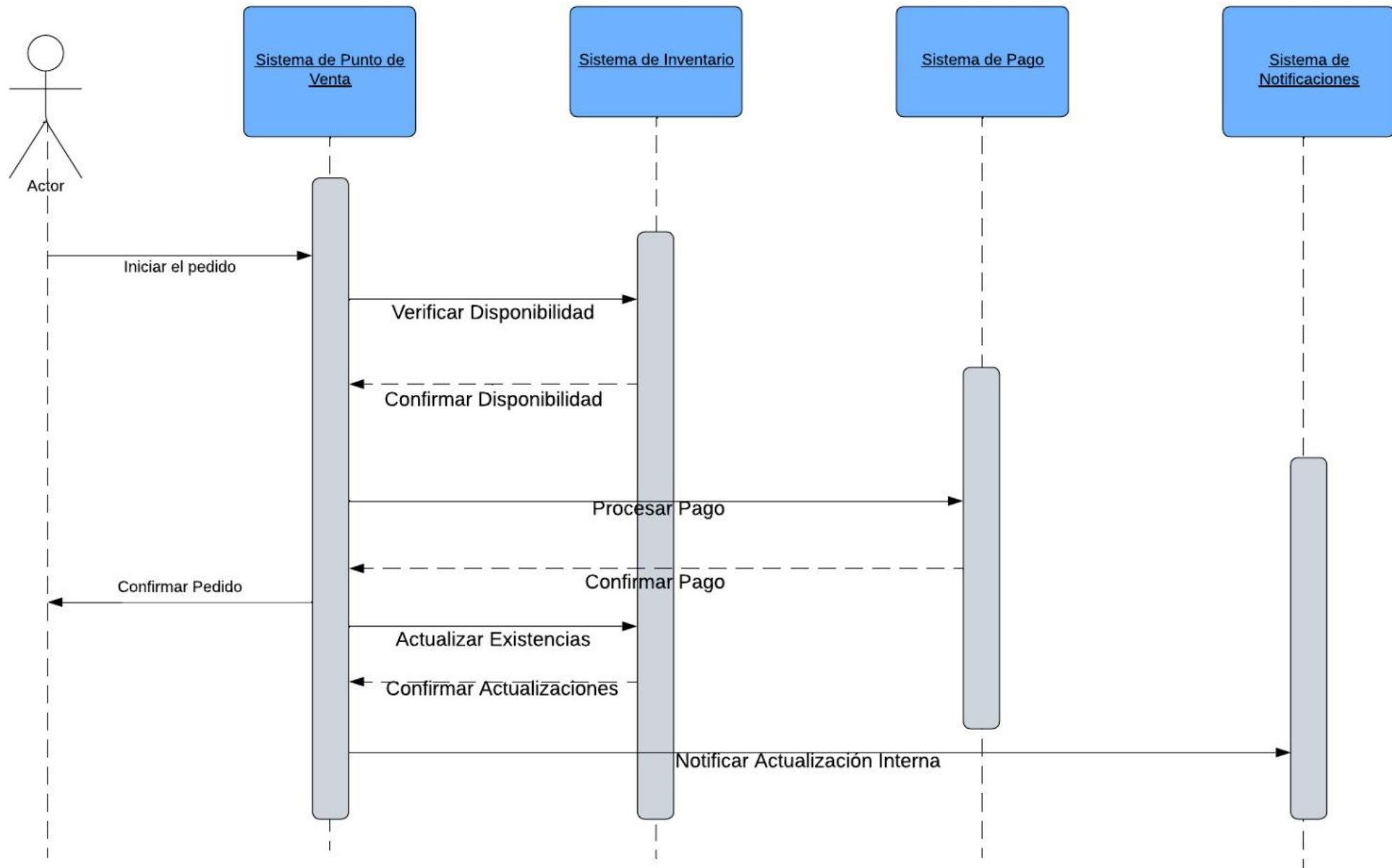


Diagrama V Secuencia Gestión de Pedidos

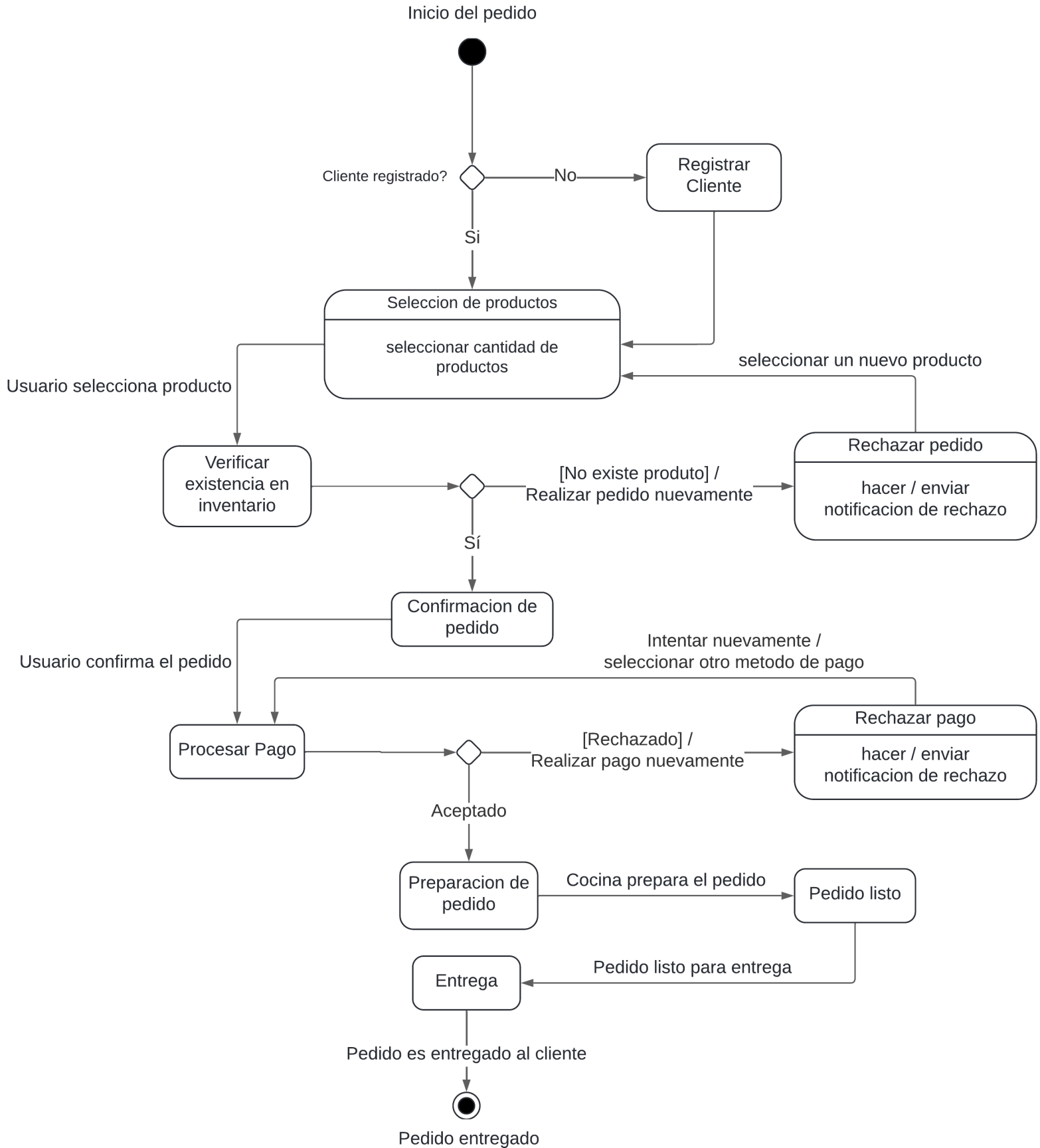


Diagrama VI Maquina de estados Gestión de pedidos

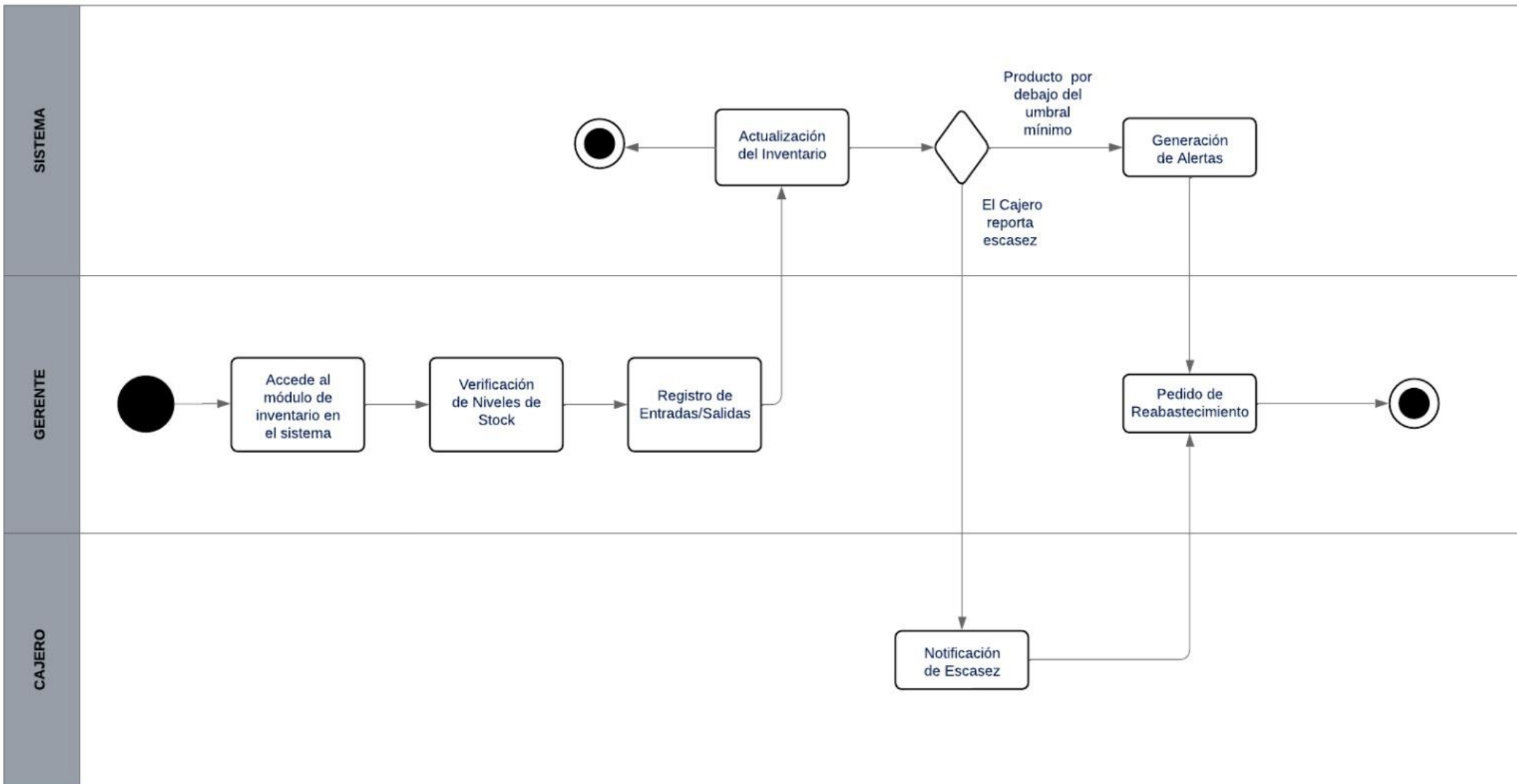


Diagrama VII Actividades Gestión de Inventario

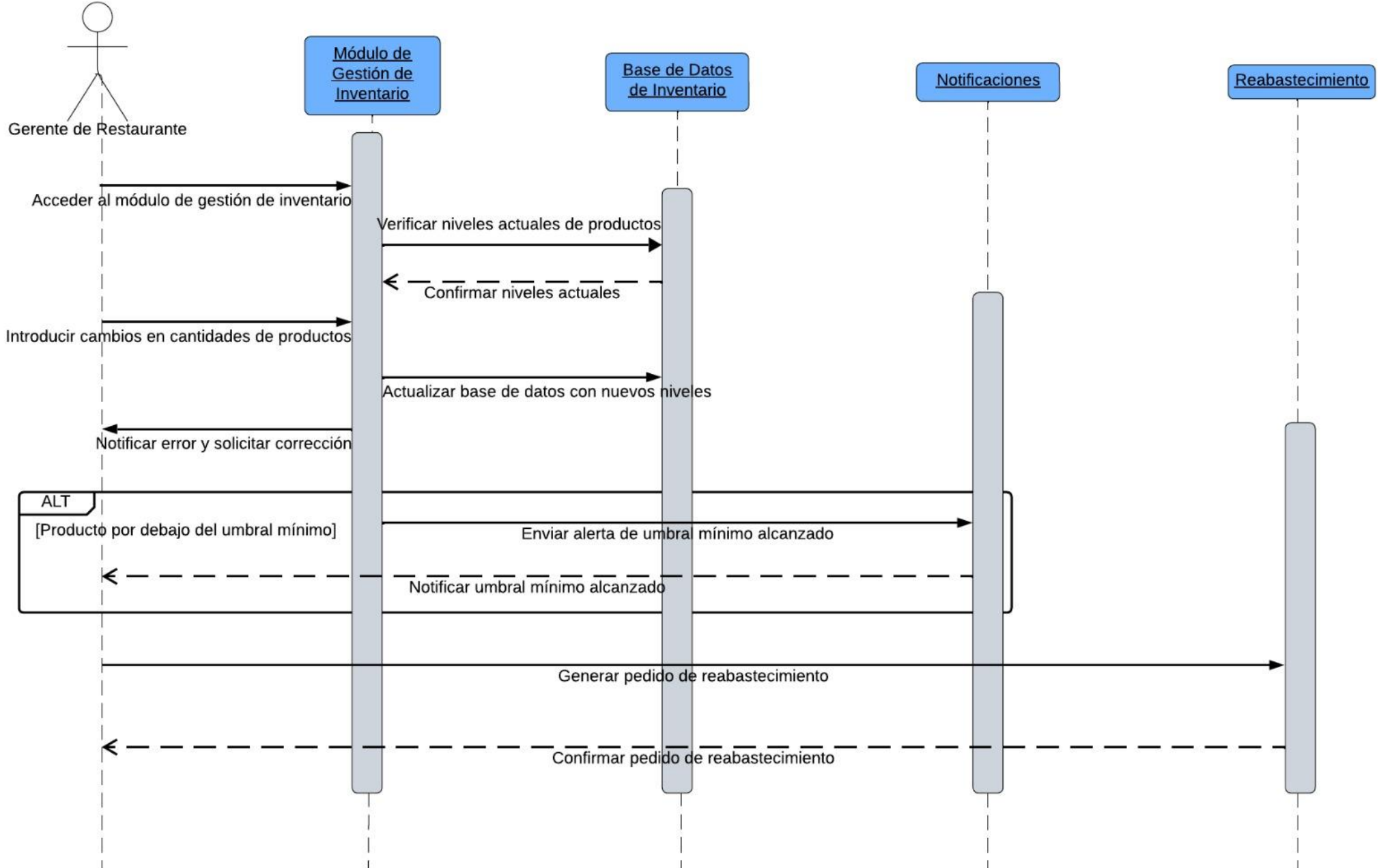


Diagrama VIII Secuencia Gestión de Inventario

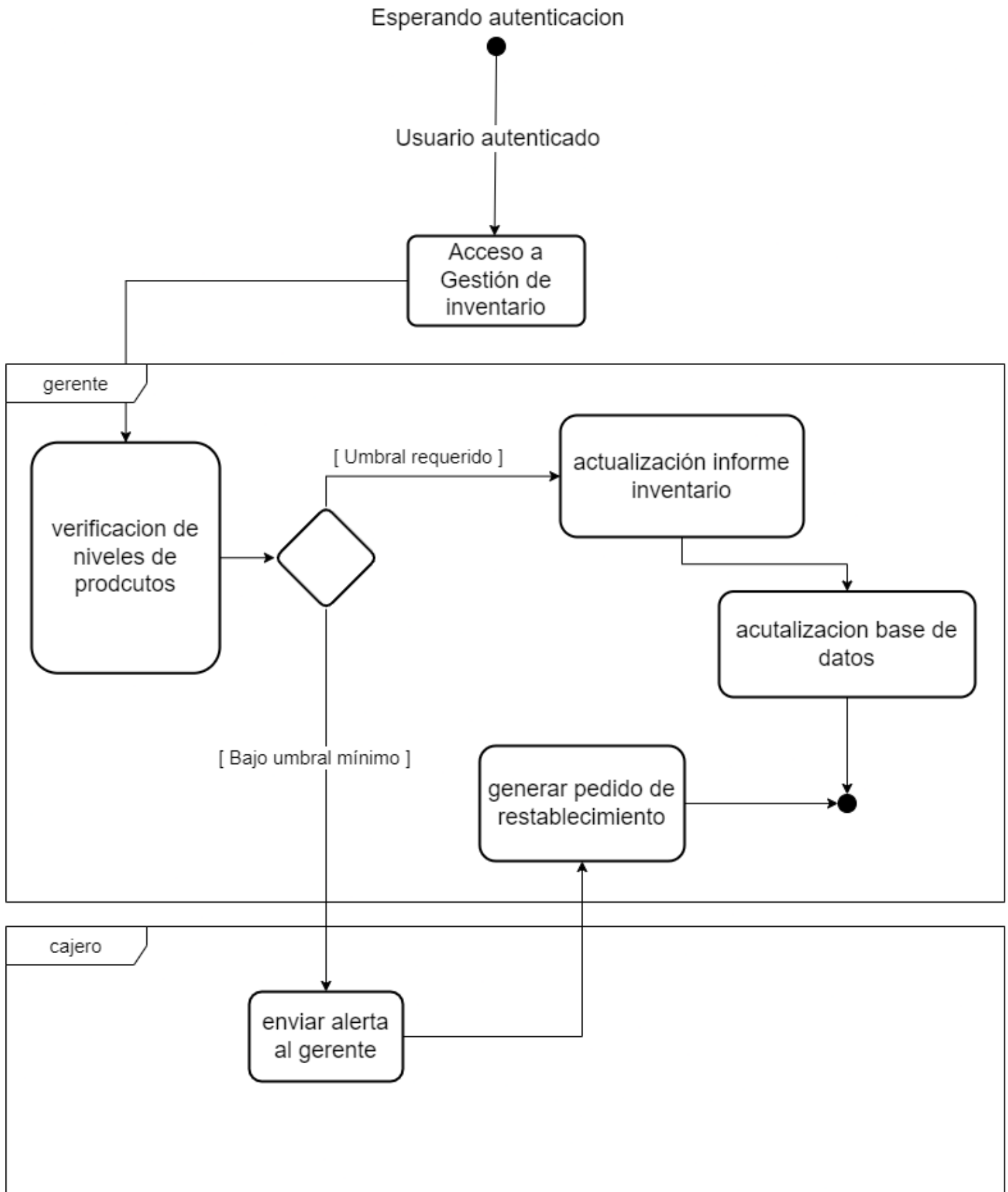


Diagrama IX Maquina de Estado Gestión de Inventario



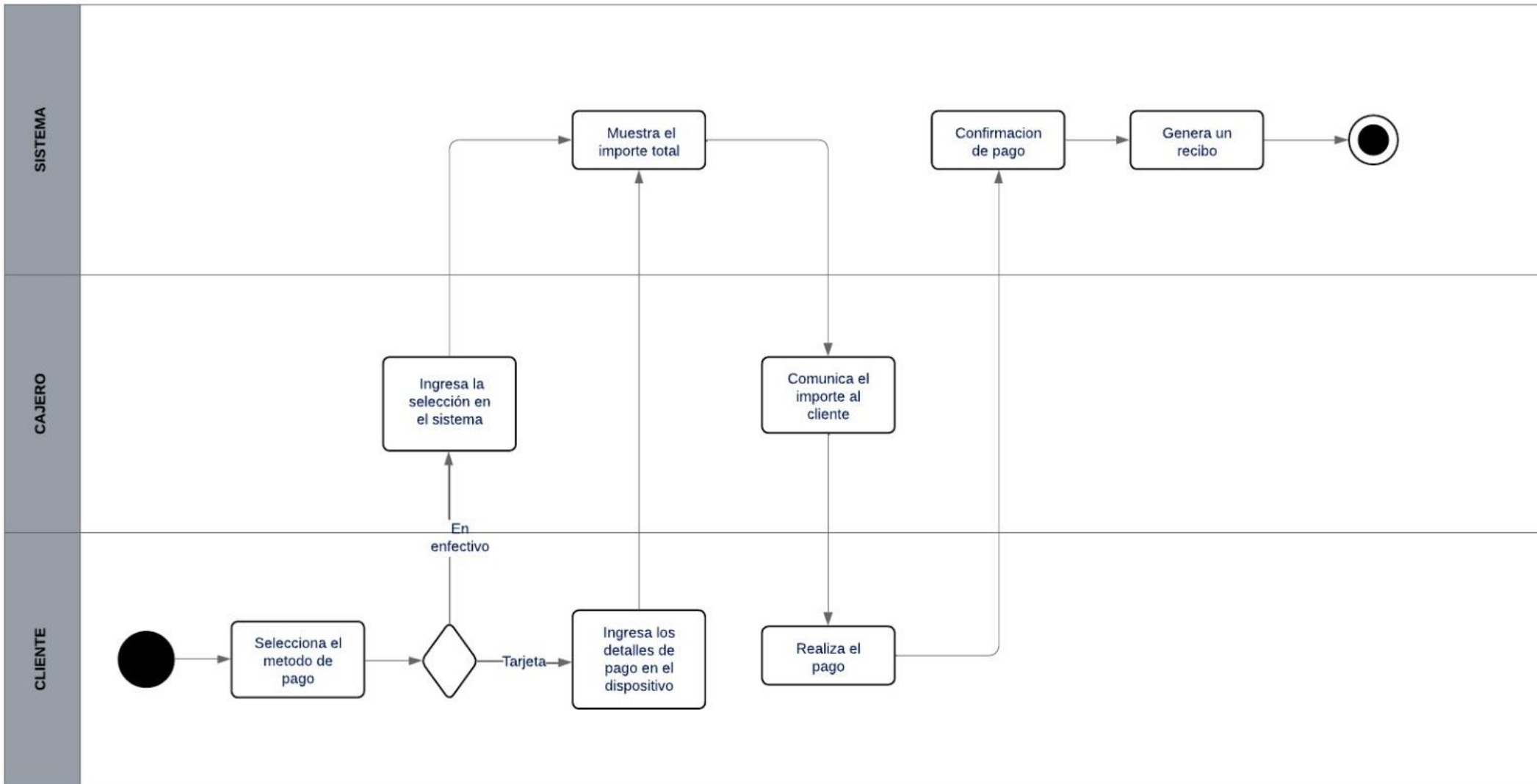


Diagrama X Actividades Pagos

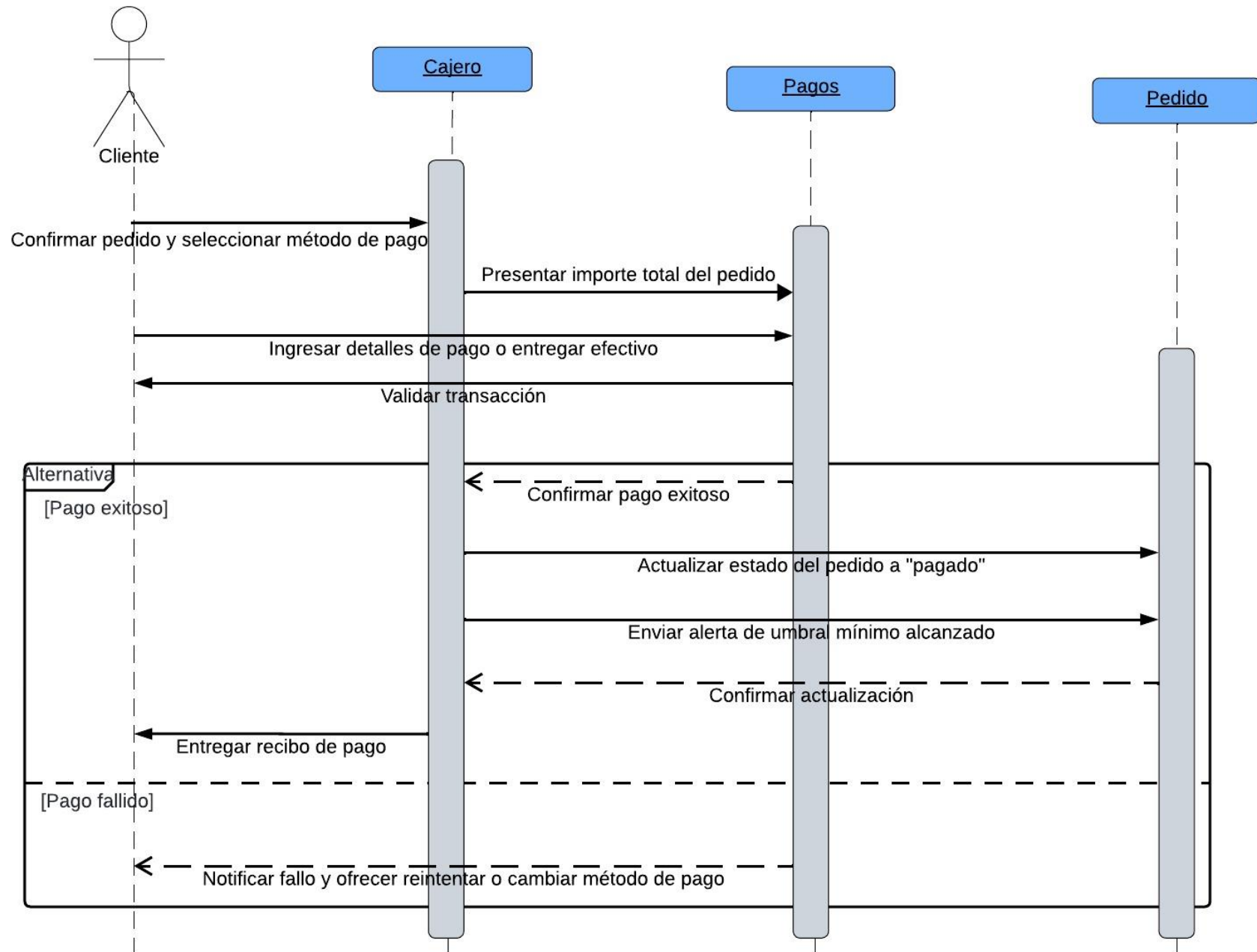


Diagrama XI Secuencia Pagos

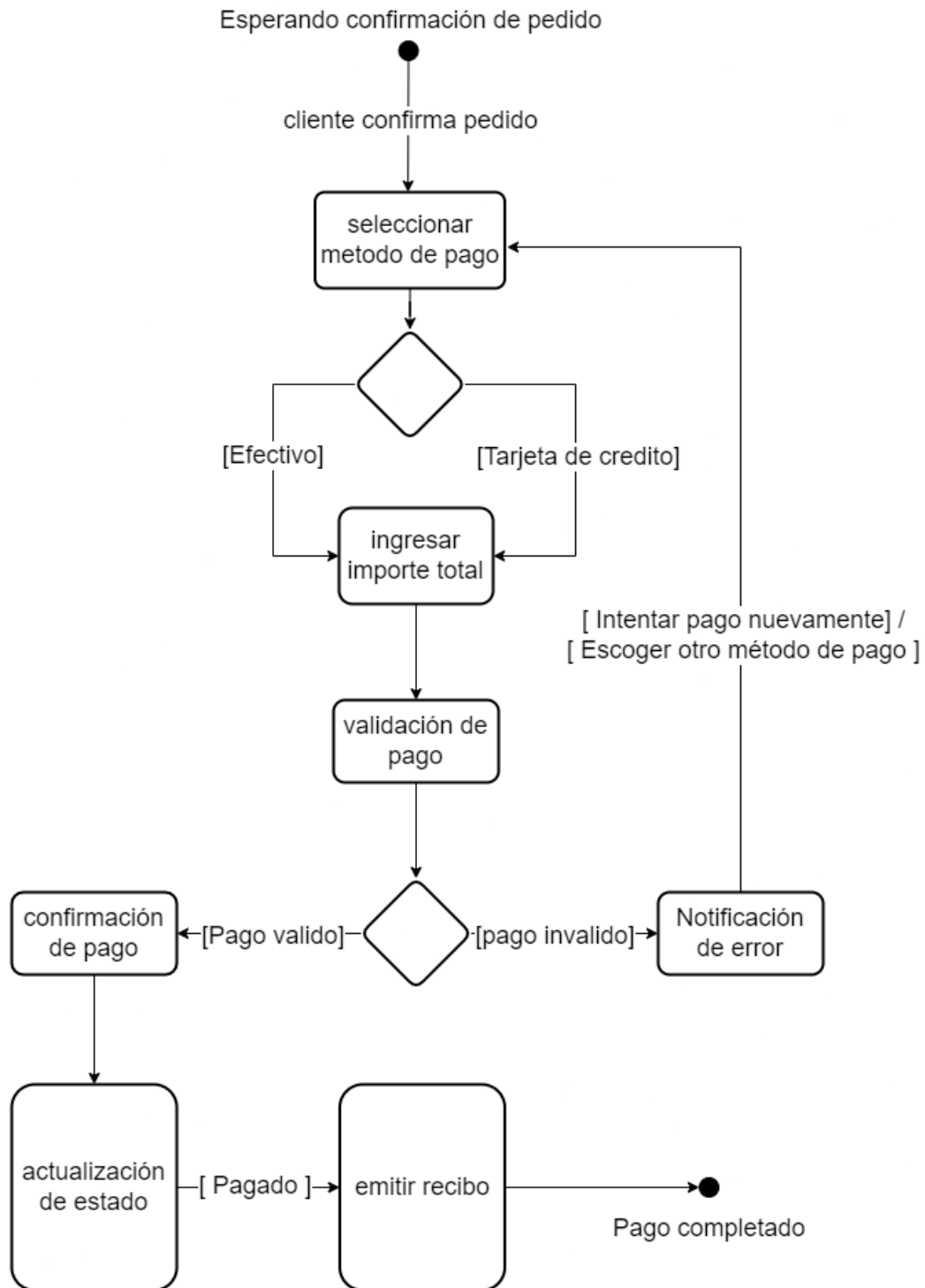


Diagrama XII Maquina Estado de Pagos

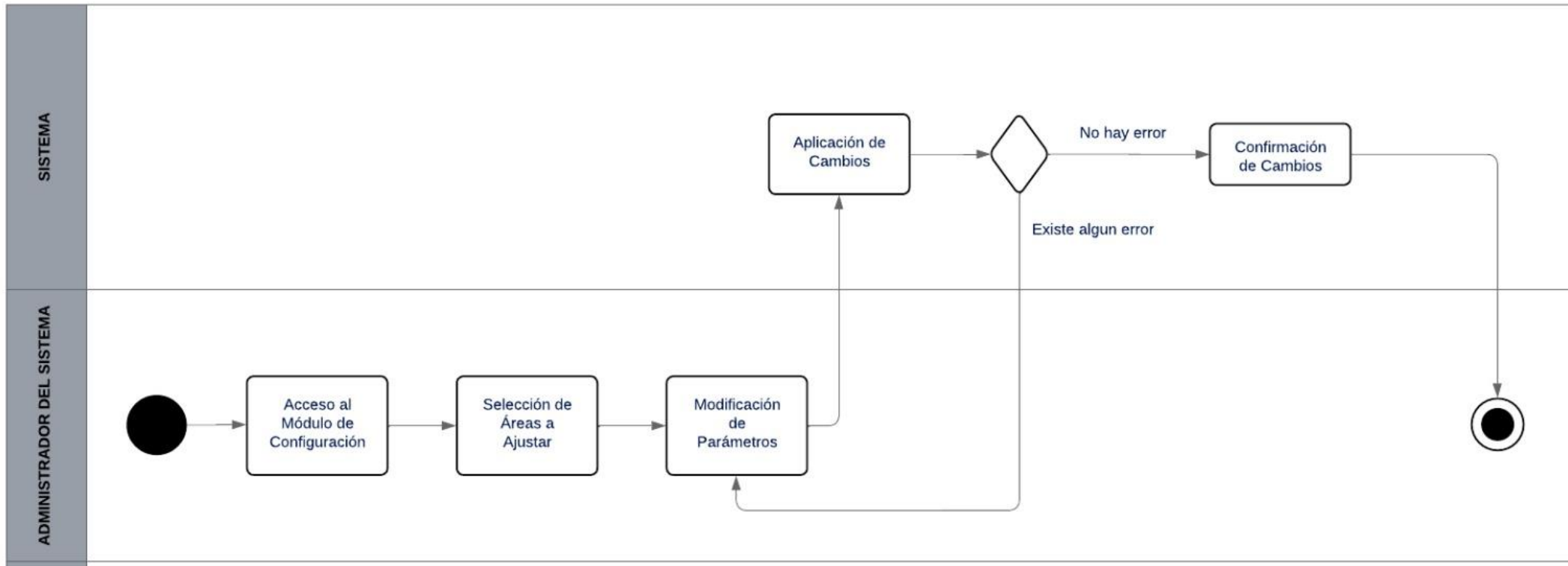


Diagrama XIII Actividades Configuración del Sistema

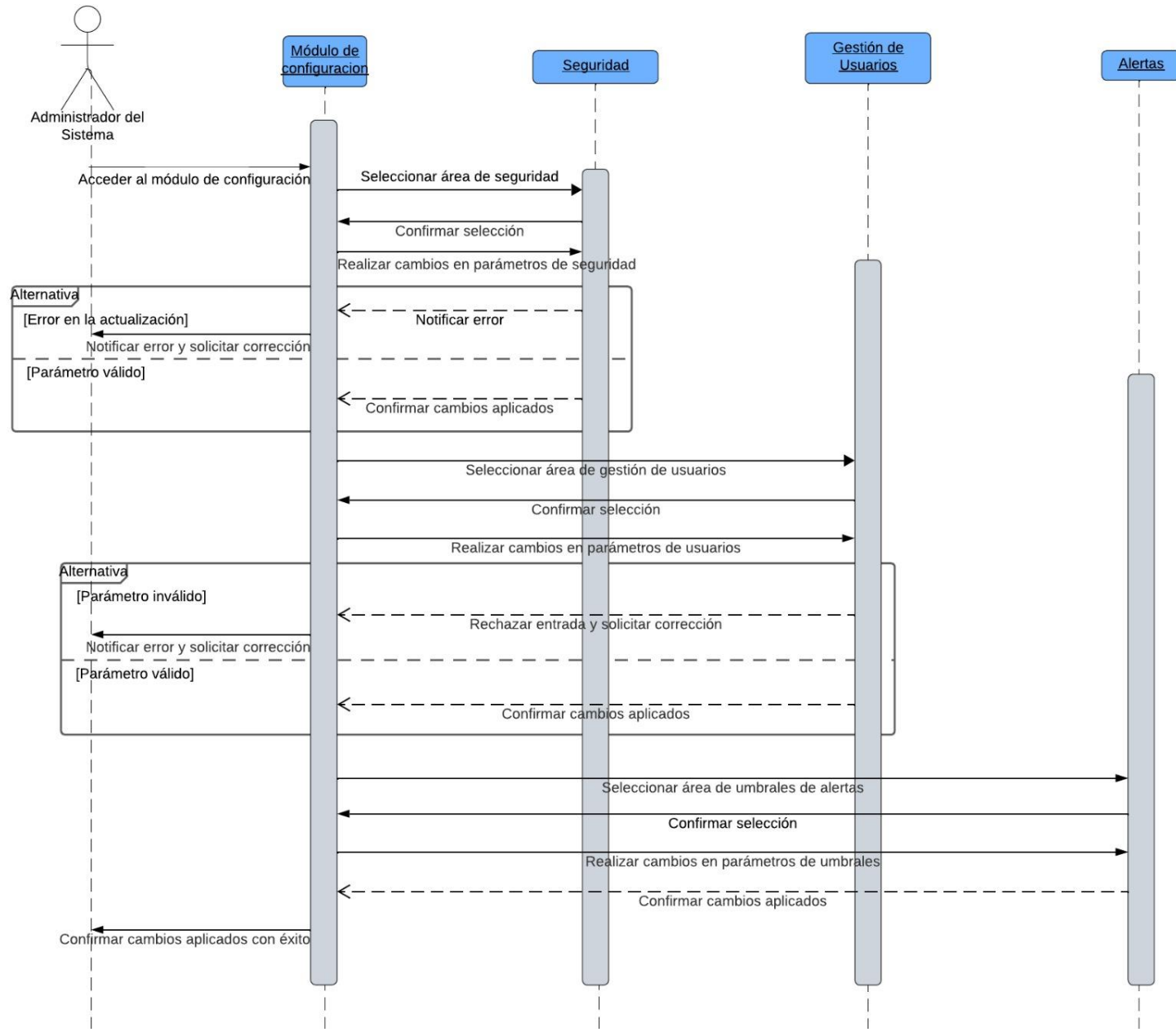


Diagrama XIV Secuencia Configuración del Sistema

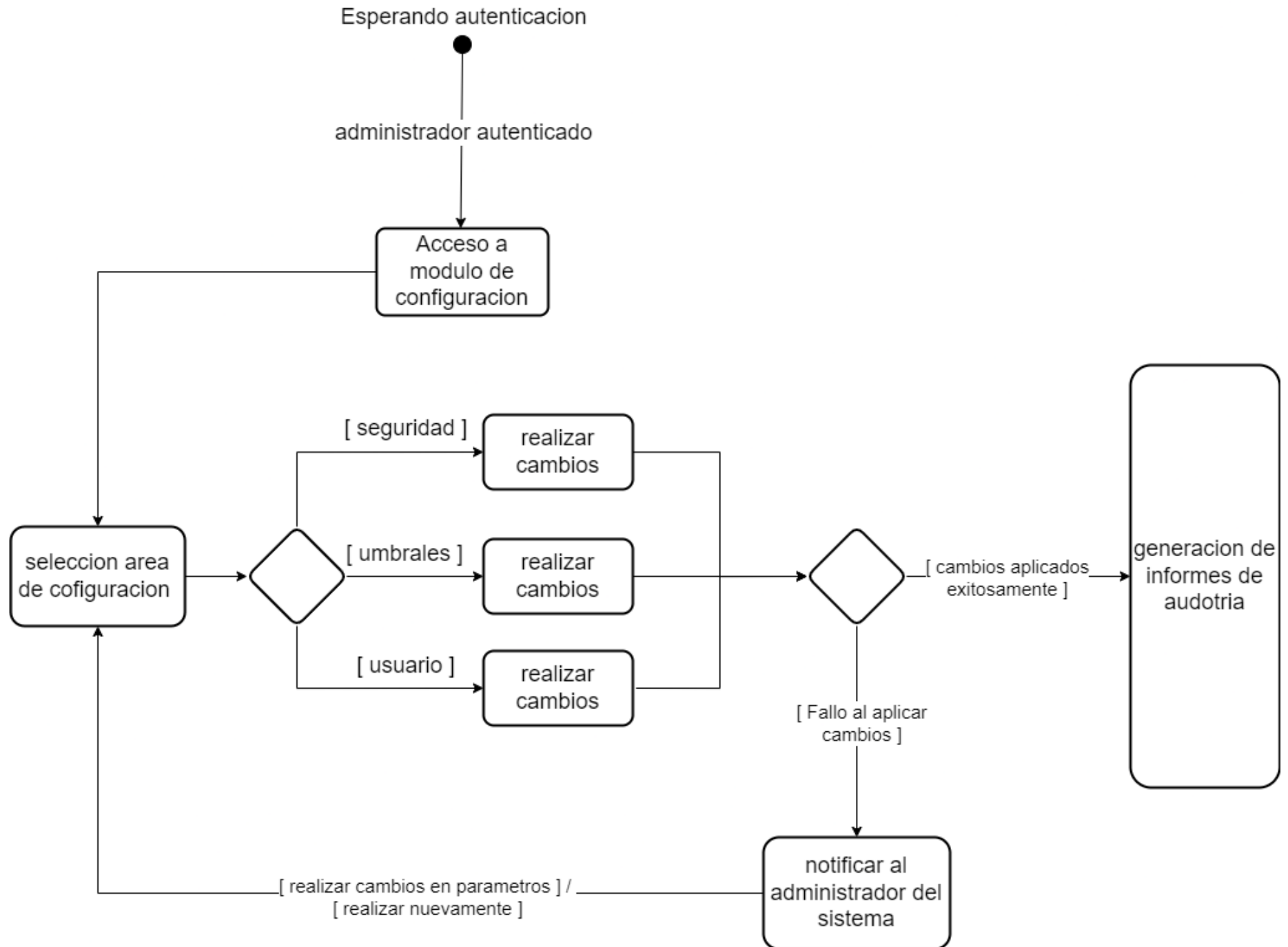


Diagrama XV Maquina de Estado Configuración del Sistema

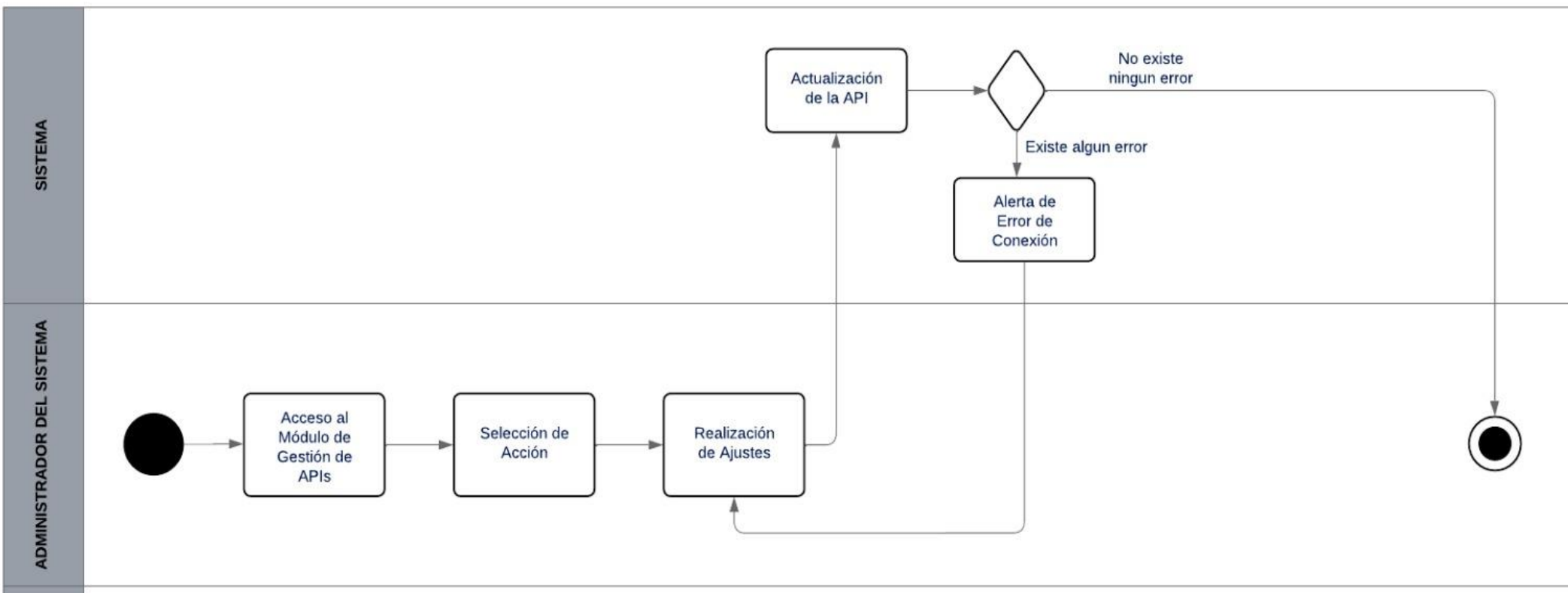


Diagrama XVI Actividades Gestión API

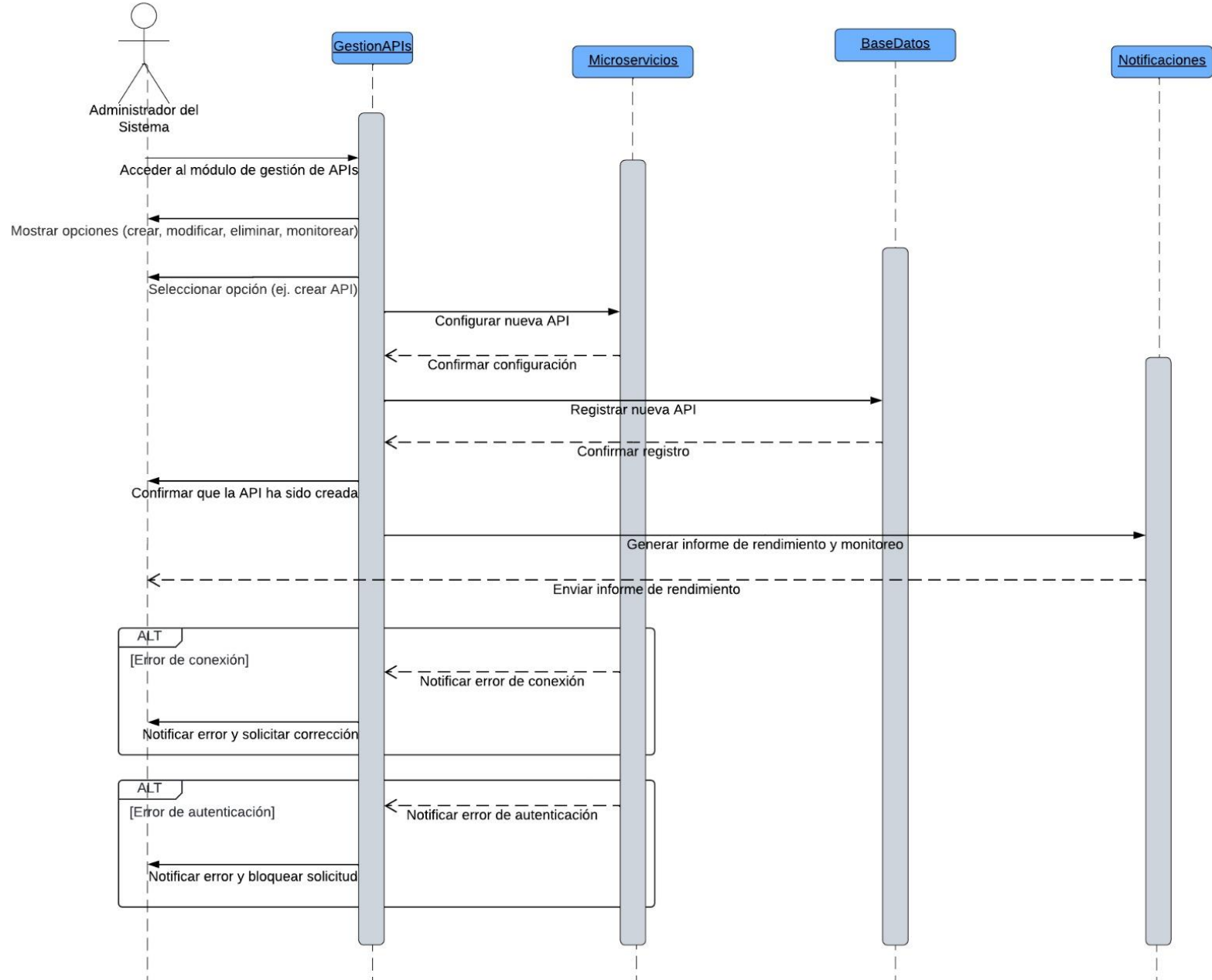


Diagrama XVII Secuencia Gestión API



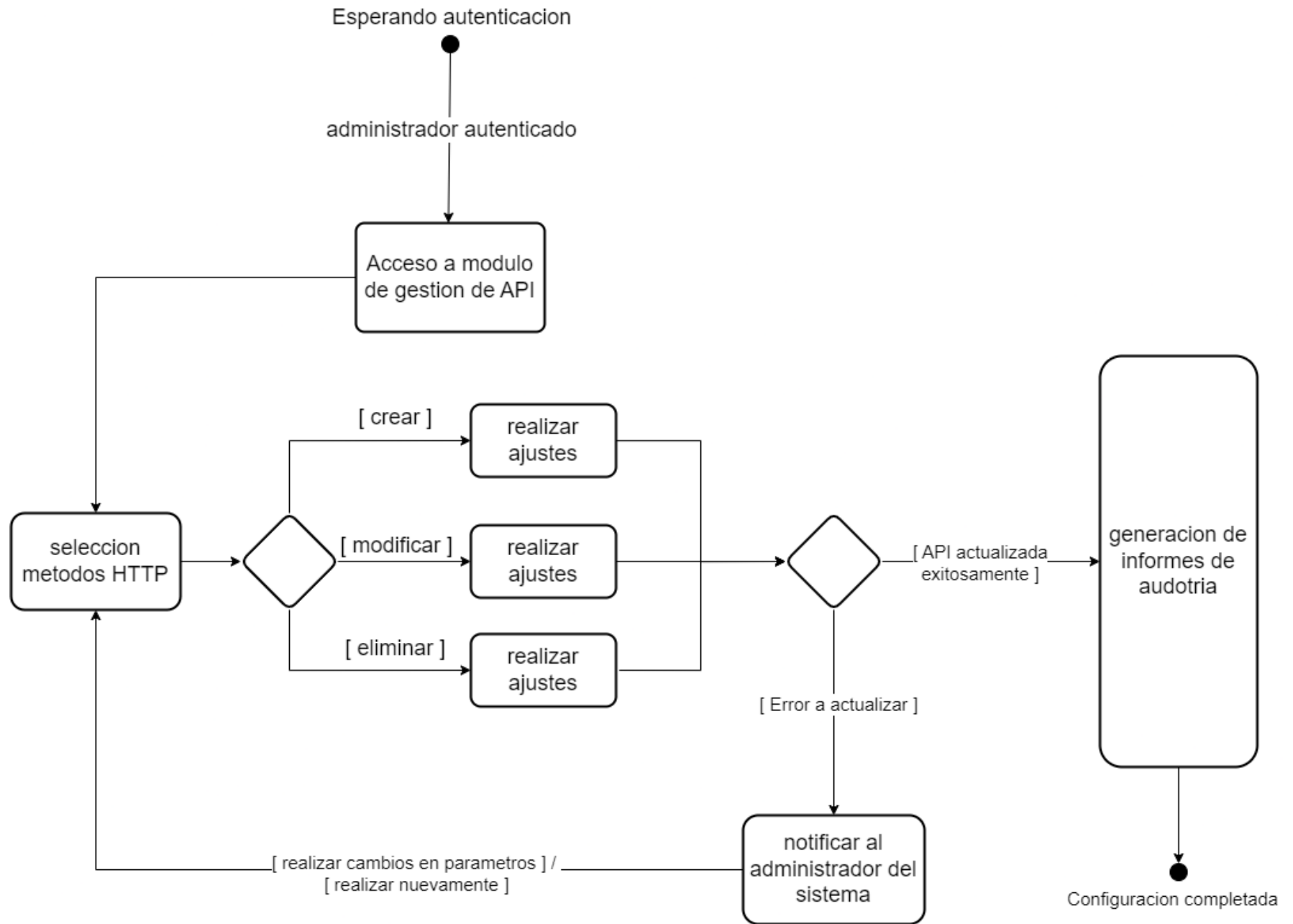


Diagrama XVIII Maquina de Estado Gestión API

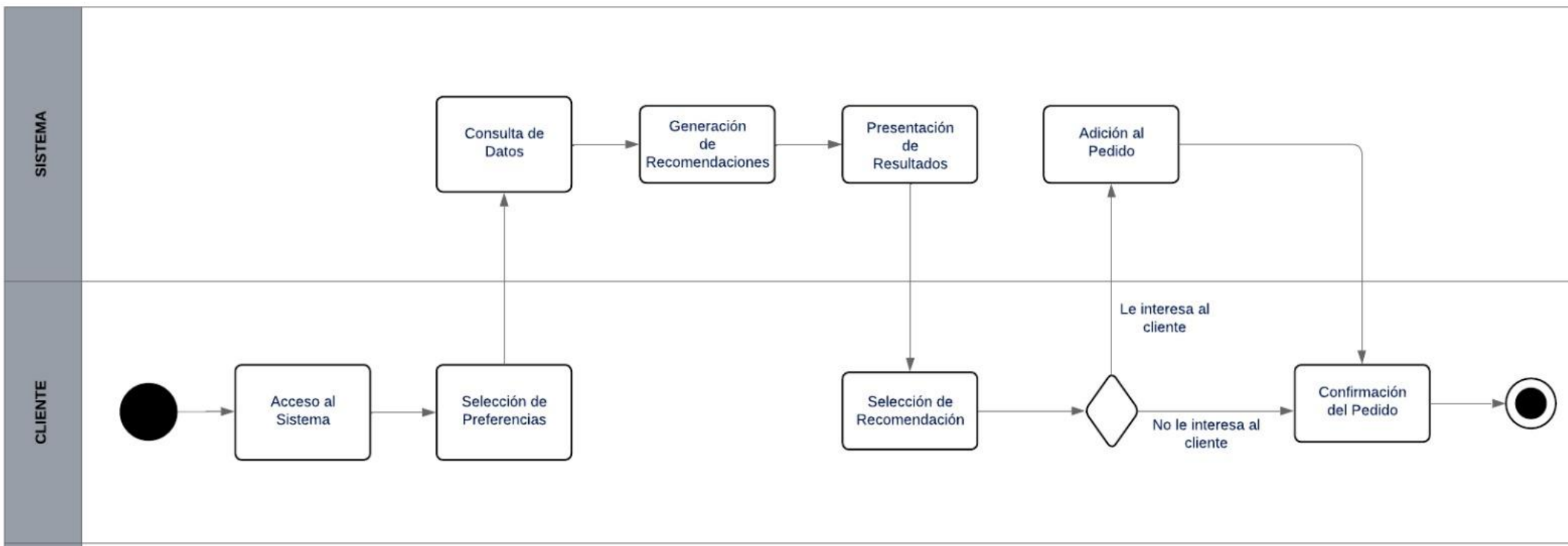


Diagrama XIX Actividades Recomendaciones

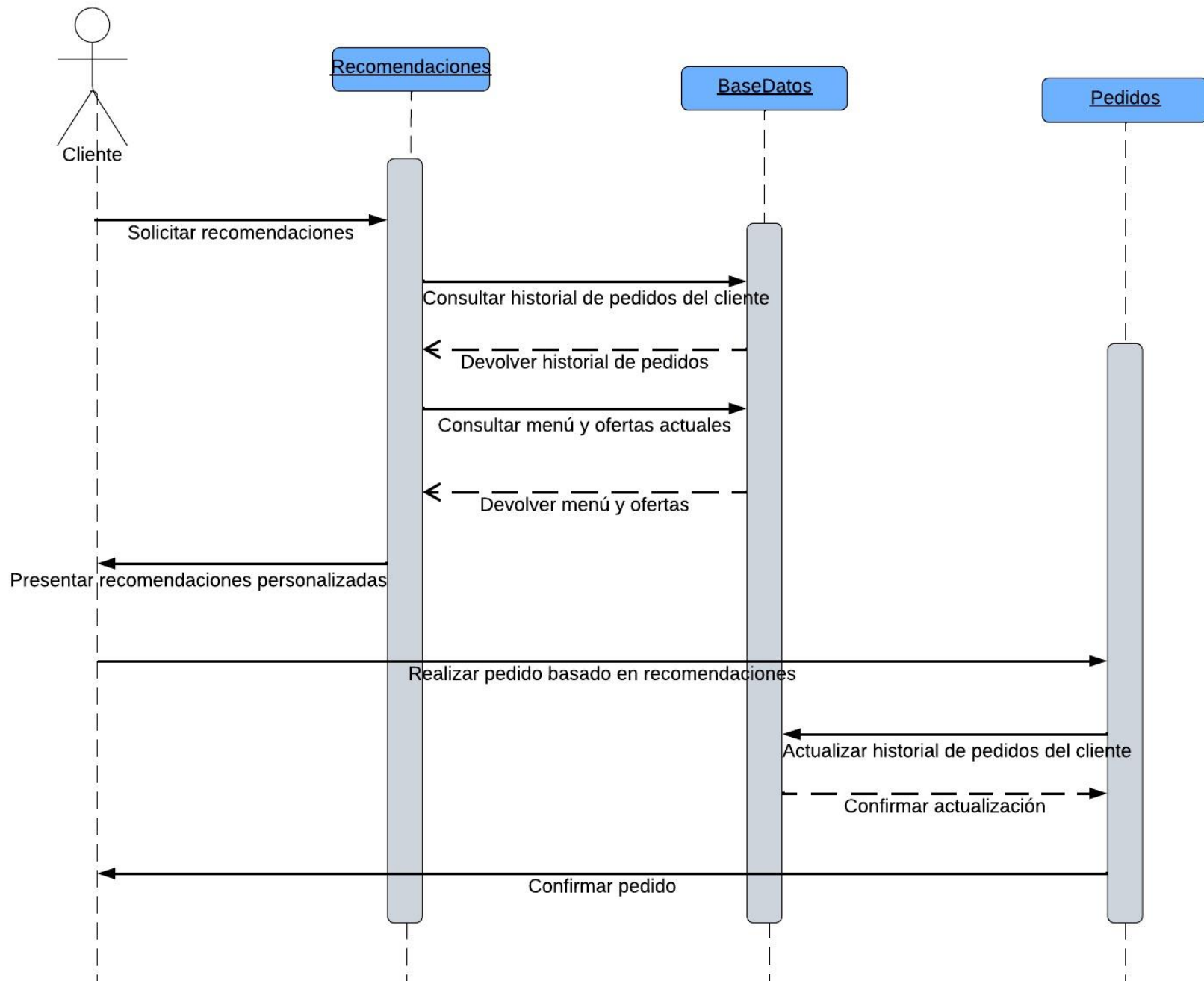


Diagrama XX Secuencia Recomendaciones

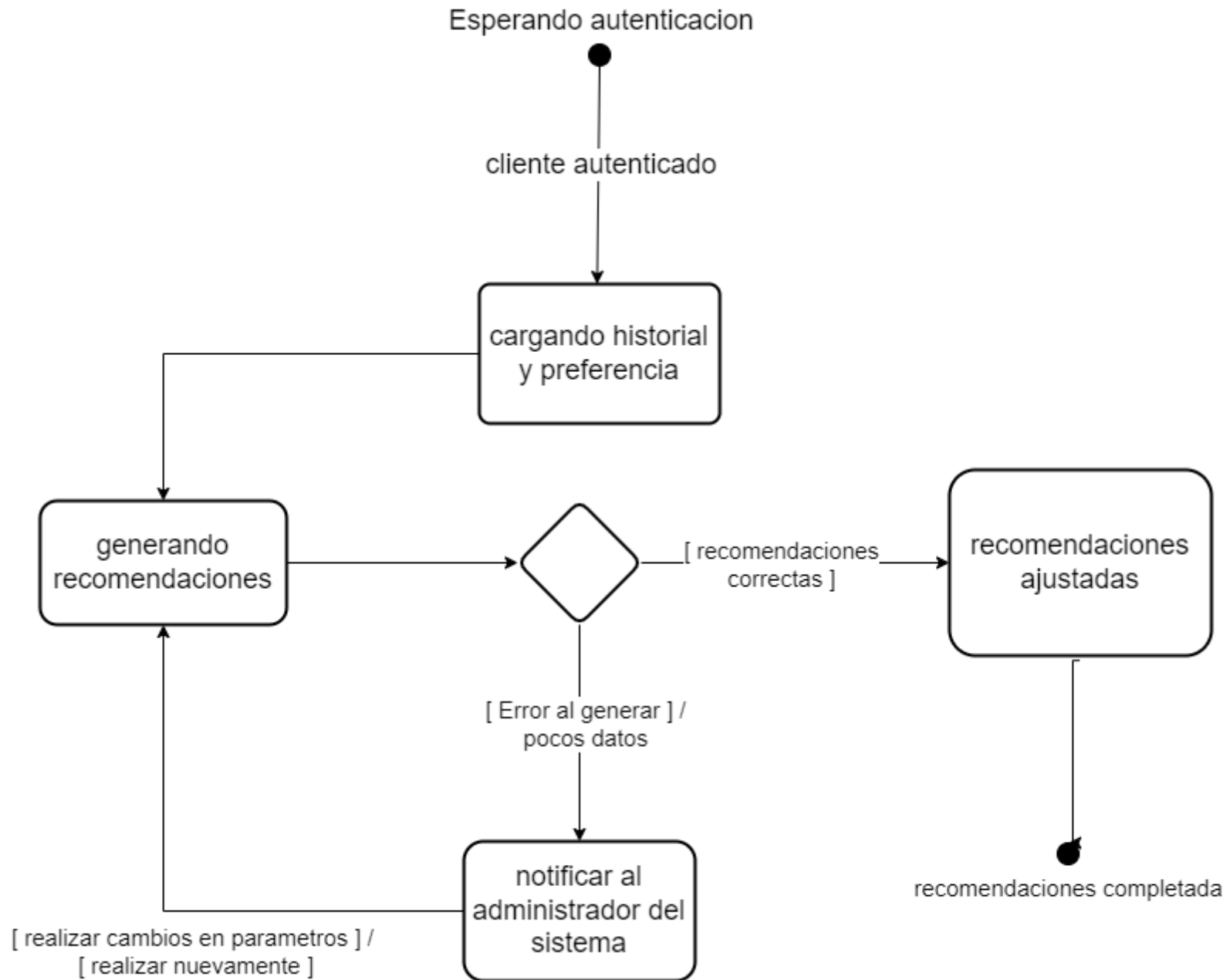


Diagrama XXI Maquina de Estado Recomendaciones