



ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

Diseño de Software

Tarea1-Travel Ease

INTEGRANTES:

Perdomo Ordoñez Paul Isaac

Herrera Nieto Christian Alexander

Muñoz Sanchez Salvador Gabriel

Rosado Alcivar Enrique Gabriel

Grupo:

7

Profesora:

Jurado Mosquera David Alonso

Periodo Académico:

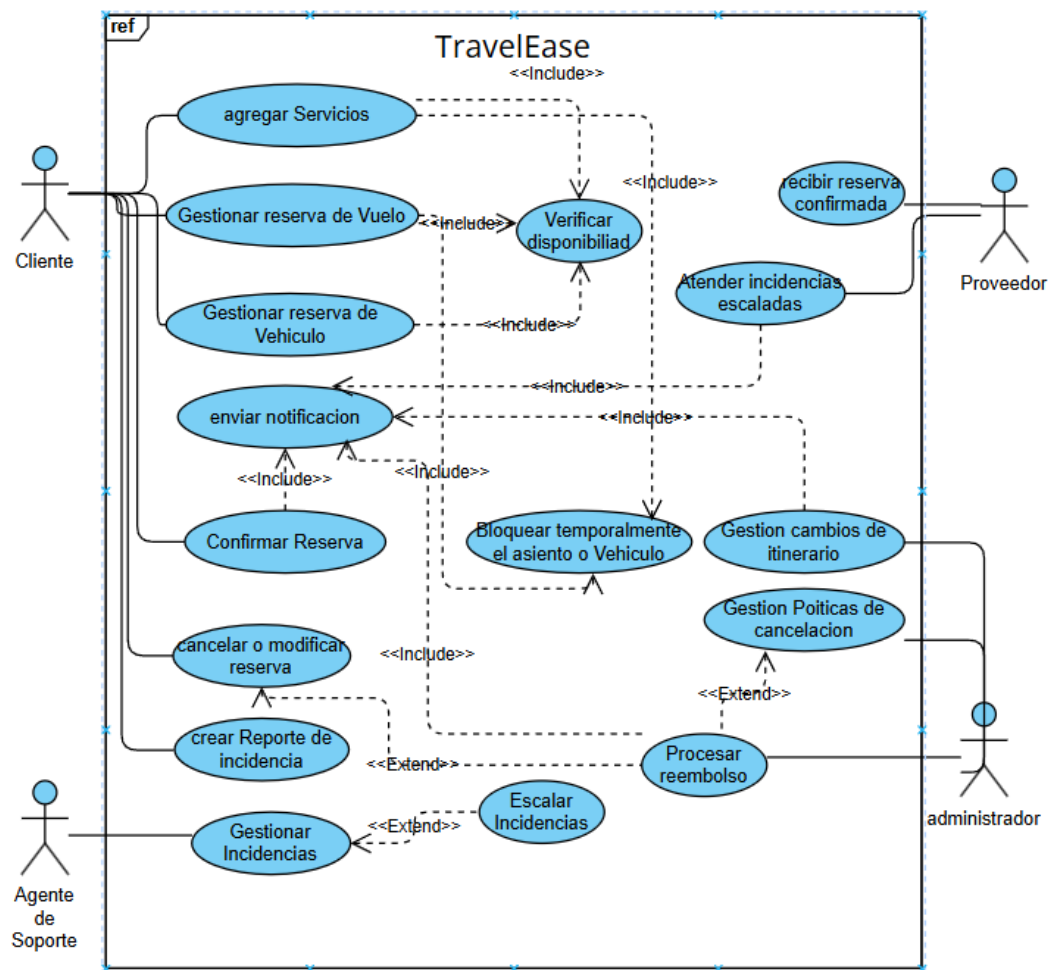
PAO II – 2024

Tabla de contenido

| | |
|-----------------------------------------------|----|
| Sección A | 3 |
| 1. Diagrama de Caso de Uso | 3 |
| 2. Caso de Uso #1..... | 4 |
| 3. Caso de uso #2..... | 5 |
| 4. Caso de uso #3..... | 6 |
| 5. Caso de uso #4..... | 7 |
| Sección B | 9 |
| 1. Diagrama de Clase..... | 9 |
| 2. Single Responsibility Principle (SRP)..... | 9 |
| 3. Open/Closed Principle (OCP) | 10 |
| 4. Liskov Substitution Principle (LSP) | 10 |
| 5. Interface Segregation Principle (ISP)..... | 10 |
| 6. Dependency Inversion Principle (DIP)..... | 11 |
| Sección C..... | 12 |
| 1. Diagrama De Secuencia #1 | 12 |
| 2. Diagrama De Secuencia #2..... | 12 |
| 3. Diagrama De Secuencia #3..... | 13 |
| 4. Diagrama De Secuencia #4..... | 13 |

Sección A

1. Diagrama de Caso de Uso



2. Caso de Uso #1

| Caso de Uso | |
|------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| ID | CDU-A01 |
| Título | Realizar Reserva integrada de vuelo y vehículo |
| Autor | Paúl Perdomo |
| Actor principal | Cliente |
| Actor secundario | |
| Incluye a | Notificar usuario |
| Extiende de | Agregar servicios adicionales |
| Descripción | Describe el flujo de acciones que realiza el usuario al reservar un vuelo con un vehículo |
| Precondición | 1. El usuario inicia sesión en el sistema |
| Secuencia normal | <ol style="list-style-type: none"> 1. El usuario selecciona el destino, fecha y preferencias de vuelo. 2. El usuario selecciona el tipo de asiento y la aerolínea. 3. El usuario selecciona el tipo de vehículo en el destino. 4. El sistema verifica la disponibilidad de asientos y vehículos. 5. El sistema bloquea temporalmente los asientos vuelo y el vehículo. 6. El usuario ingresa los datos de pago y confirma la reserva. 7. El sistema confirma la reserva y envía una notificación. |
| Secuencia alternativa | <p>En el paso 5 si no está disponible el asiento o el vehículo se le presentan alternativas al usuario</p> <p>En el paso 6, si el usuario desea puede agregar servicios adicionales</p> <p>En el paso 7 si el método de pago falla, desbloquea los asientos en los vuelos y los vehículos en el destino</p> |
| Postcondición | El vuelo y el vehículo quedan en estado "confirmado". |

3. Caso de uso #2

| Caso de Uso | |
|------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| ID | CDU-A02 |
| Título | Cancelar reserva |
| Autor | Paúl Perdomo |
| Actor principal | Cliente |
| Actor secundario | Administrador |
| Incluye a | Notificar usuario |
| Extiende de | |
| Descripción | Describe el flujo de acciones que realizan los profesores y alumnos para reportar problemas del sistema en la plataforma. El equipo de soporte técnico se encarga de resolver los problemas técnicos y, en caso de problemas de gestión, los derivan a los administradores correspondientes. |
| Precondición | Debe existir una reserva en estado "reservado" o "confirmado". |
| Secuencia normal | <ol style="list-style-type: none"> 1. El usuario selecciona una reserva. 2. Solicita la cancelación de la reserva. 3. El sistema verifica las políticas de cancelación aplicables. 4. Si procede, el sistema cancela la reserva y procesa Reembolso si es necesario. 5. El sistema notifica al Usuario sobre el estado de la cancelación. |
| Secuencia alternativa | En el paso 4 si la cancelación no es posible según las políticas, el sistema muestra un mensaje de error. |
| Postcondición | El vuelo y el vehículo quedan en estado "cancelado".} El usuario recibe una notificación sobre el estado de vuelo |

4. Caso de uso #3

| Caso de Uso | |
|-------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| ID | CDU-A03 |
| Título | Reportar incidente |
| Autor | Paúl Perdomo |
| Actor principal | Cliente |
| Actor secundario | Agente de soporte |
| Incluye a | Notificar usuario |
| Extiende de | |
| Descripción | Describe el flujo de acciones que realiza el usuario para reportar problemas del sistema en la plataforma. El agente de soporte se encarga de resolver los problemas y, en caso de no resolverse, son escalados al departamento de servicio al cliente de la aerolínea o la empresa de alquiler. |
| Precondición | Debe existir una reserva en estado "confirmado" o "reservado". Debe existir un problema |
| Secuencia normal | <ol style="list-style-type: none"> 1. El usuario reporta una incidencia (ej. problema con el horario o vehículo). 2. El agente de soporte revisa la incidencia. 3. El agente resuelve la incidencia. 4. Se notifica al usuario. |

| | |
|------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Secuencia alternativa | En el paso 3 si el agente no puede resolver la incidencia, esta es escalada al departamento de servicio al cliente de la aerolínea o la empresa de alquiler |
| Postcondición | Si el problema no pudo ser resuelto, es escalada al departamento de servicio al cliente de la aerolínea o la empresa de alquiler, en caso contrario se resolvió el conflicto. El usuario recibe notificaciones sobre el estado y la resolución de su problema |

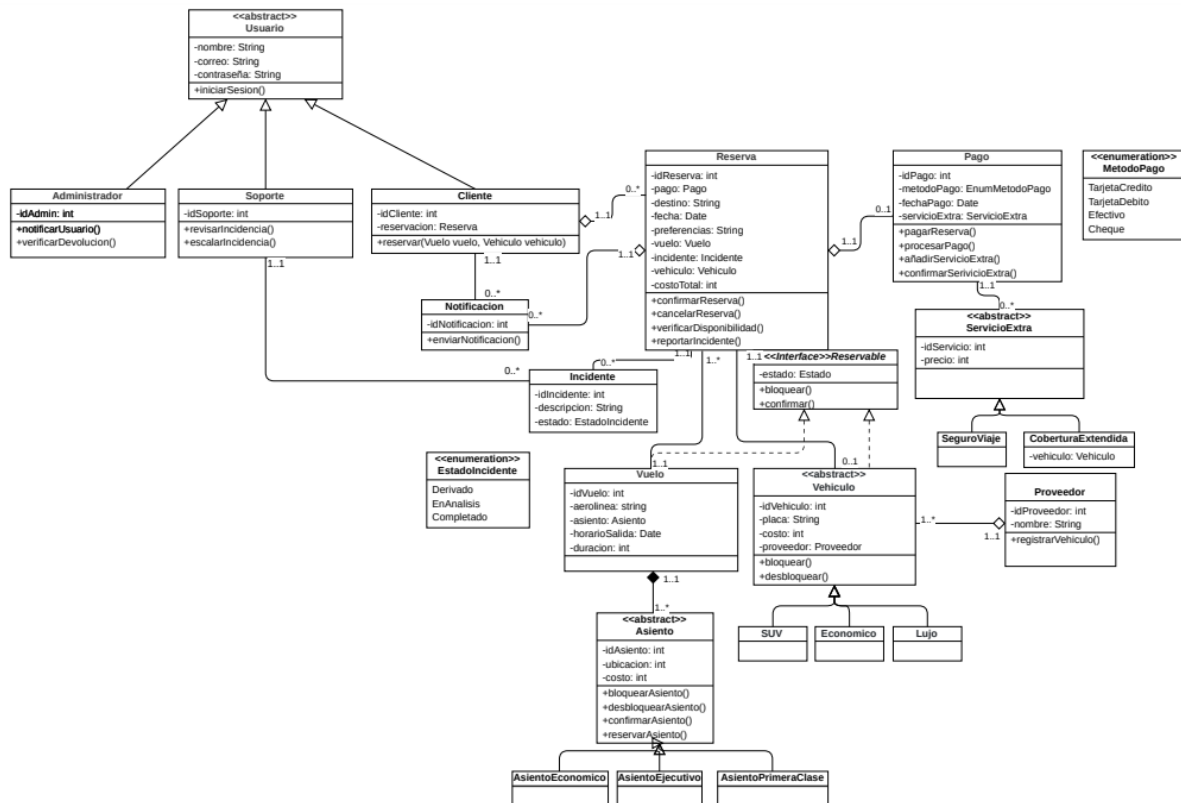
5. Caso de uso #4

| Caso de Uso | |
|-------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| ID | CDU -A04 |
| Título | Agregar Servicios Adicionales |
| Autor | Paúl Perdomo |
| Actor principal | Cliente |
| Actor secundario | |
| Incluye a | |
| Extiende de | Realizar reserva |
| Descripción | Permite al usuario añadir servicios adicionales a su reserva, como seguro de viaje o cobertura extendida para el vehículo alquilado. |
| Precondición | El usuario tiene una reserva existente o en proceso. |
| Secuencia normal | <ol style="list-style-type: none"> 1. El usuario selecciona la opción de agregar servicios adicionales. 2. El sistema muestra las opciones de servicios disponibles. 3. El usuario elige los servicios adicionales deseados. |

| | |
|------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| | <ul style="list-style-type: none"> 4. El sistema actualiza el costo total. 5. El usuario confirma el pago de los servicios adicionales. 6. El sistema confirma los servicios y envía una notificación al usuario. |
| Secuencia alternativa | En el paso 5, si el pago falla, el sistema cancela la adición de los servicios. |
| Postcondición | <p>Los servicios adicionales quedan confirmados en la reserva.</p> <p>La reserva obtiene un aumento de precio</p> |

Sección B

1. Diagrama de Clase



2. Single Responsibility Principle (SRP)

Justificación:

- El usuario encapsula la información y conductas generales de cualquier tipo de usuario (nombre, correo electrónico, contraseña y método de iniciar sesión()).
- Las subclases Administrador, Soporte y Cliente proporcionan características particulares dependiendo del puesto: Administrador tiene la capacidad de informar y confirmar devoluciones; Soporte gestiona las incidencias; y el Cliente efectúa reservas.
- Clases como Reserva, Pago, Incidencia y Notificación poseen obligaciones claras, lo que simplifica el mantenimiento al restringir las razones para transformar cada clase en una sola responsabilidad.

3. Open/Closed Principle (OCP)

Justificación:

- Se emplean clases abstractas (Usuario, ServicioExtra, Vehículo, Asiento) con el objetivo de expandir el sistema con nuevos tipos de usuarios, servicios extra, vehículos o asientos, sin alterar las clases base.
- Listas como EstadoIncidente y MetodoPago facilitan la incorporación de nuevos estados o métodos de pago sin la necesidad de alterar el código existente, solo ampliando estas estructuras.
- Esto garantiza que el sistema pueda ser expandido sin perjudicar su operatividad actual, respetando el principio de apertura para ampliación y cierre para alteración.

4. Liskov Substitution Principle (LSP)

Justificación:

- Las clases derivadas de Usuario (Cliente, Soporte, Administrador) mantienen las conductas establecidas en la clase base Usuario, lo que permite su implementación en el sistema donde se prevea la presencia de un Usuario.
- La interfaz Reservable asegura que cualquier clase que la implemente (como Asiento y Vehículo) cumpla con las funciones esperadas para reservas, como bloquear y confirmar, de manera que pueden emplearse de manera independiente siempre que se requiera un objeto reservable.
- Esto garantiza que las variantes de clases derivadas sean capaces de sustituir a sus superclases sin alterar la conducta del sistema.

5. Interface Segregation Principle (ISP)

Justificación:

- La interfaz Reservable solo incluye los métodos requeridos (bloquear() y confirmar()), y solo se aplica a clases que se identifican como reservables (como Asiento y Vehículo).
- Esto previene la acumulación de métodos superfluos en las clases, posibilitando que cada clase únicamente aplique métodos directamente pertinentes a su tarea, siguiendo el principio de separación de interface.

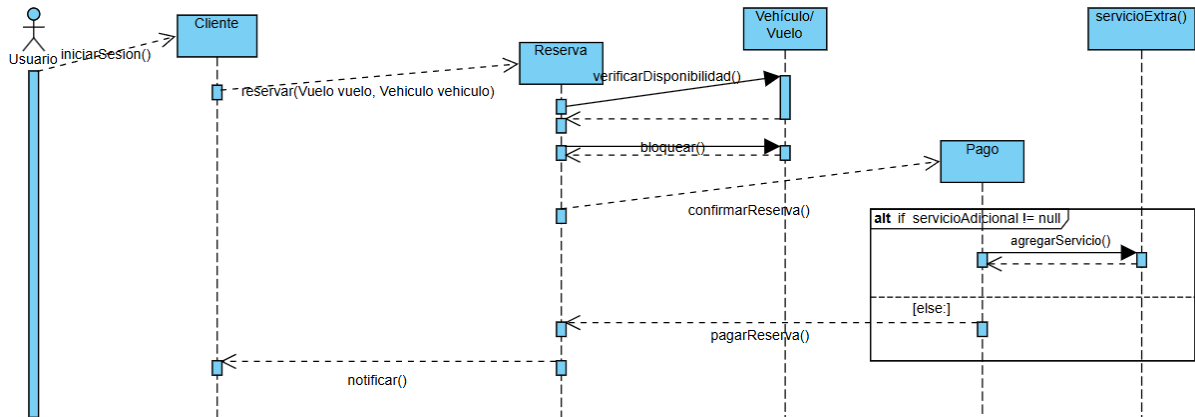
6. Dependency Inversion Principle (DIP)

Justificación:

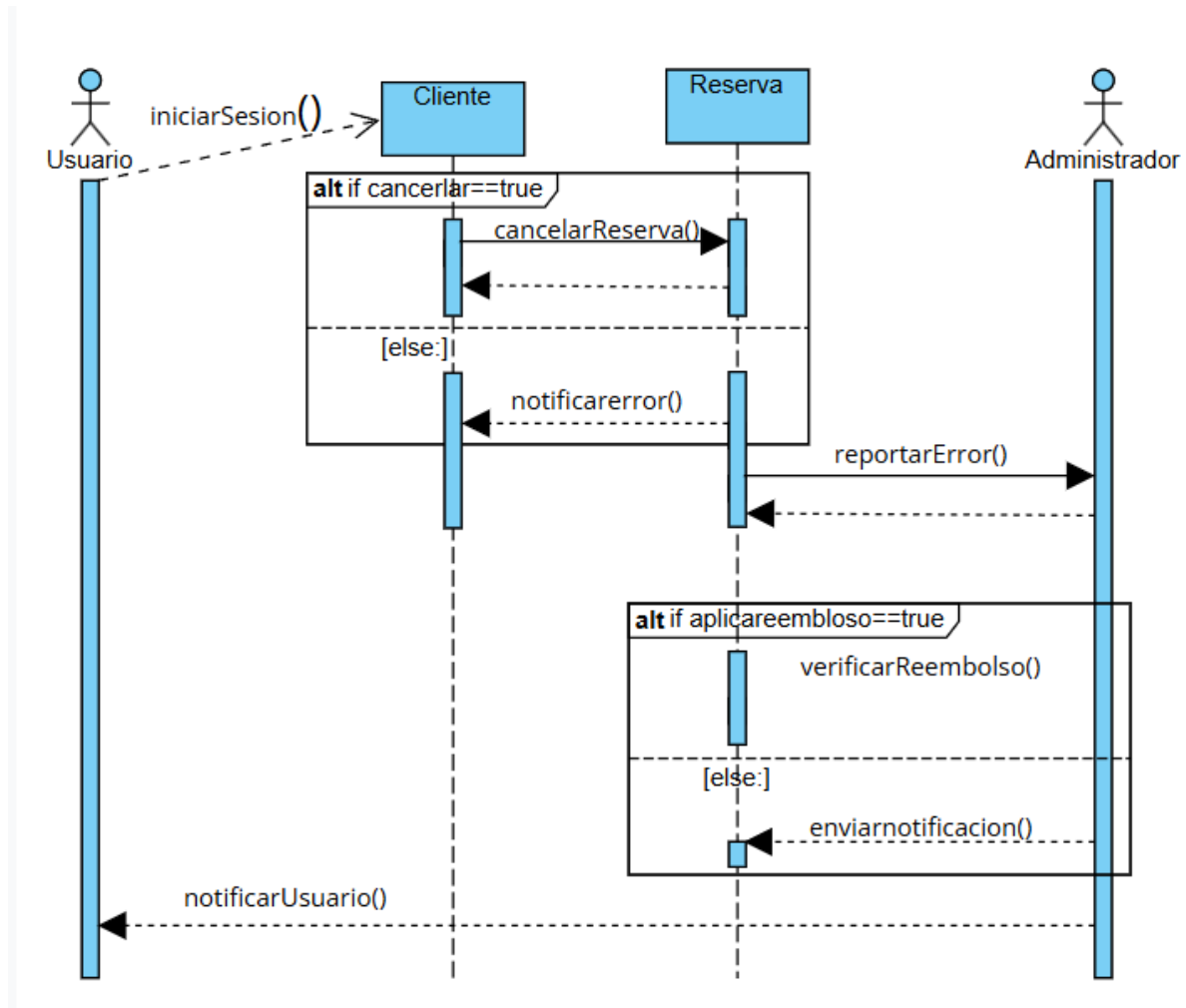
- Reserva y Pago se basan en la abstracción ServicioExtra y no en servicios específicos, lo que facilita la incorporación de nuevos tipos de servicios extras sin alterar estas clases.
- La implementación de MetodoPago como lista hace que Pago se base en una abstracción de métodos de pago en vez de métodos específicos, lo que favorece la inclusión de nuevas modalidades de pago en el futuro.
- Esto disminuye el vínculo y permite la ampliación del sistema a través de la utilización de abstracciones, respetando el principio de inversión de dependencias.

Sección C

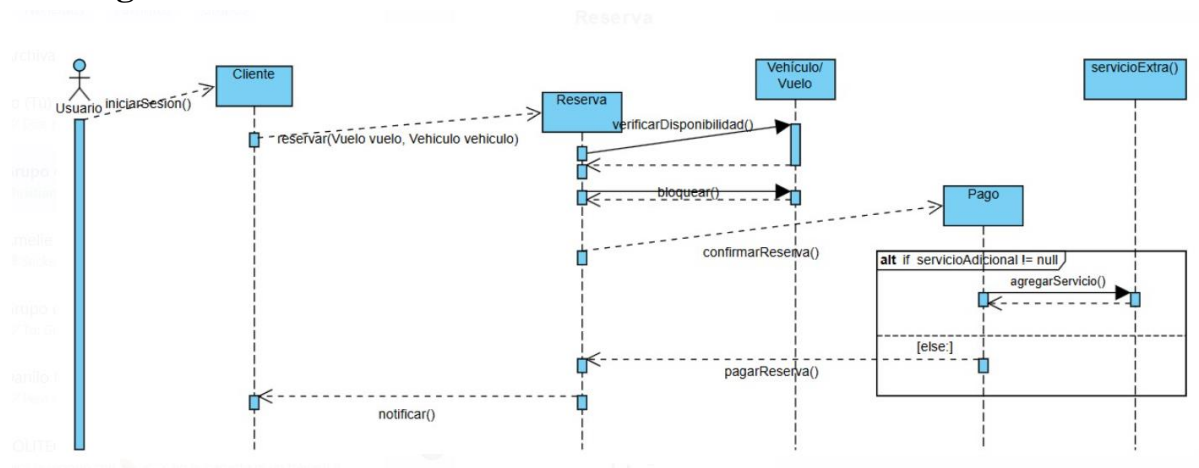
1. Diagrama De Secuencia #1



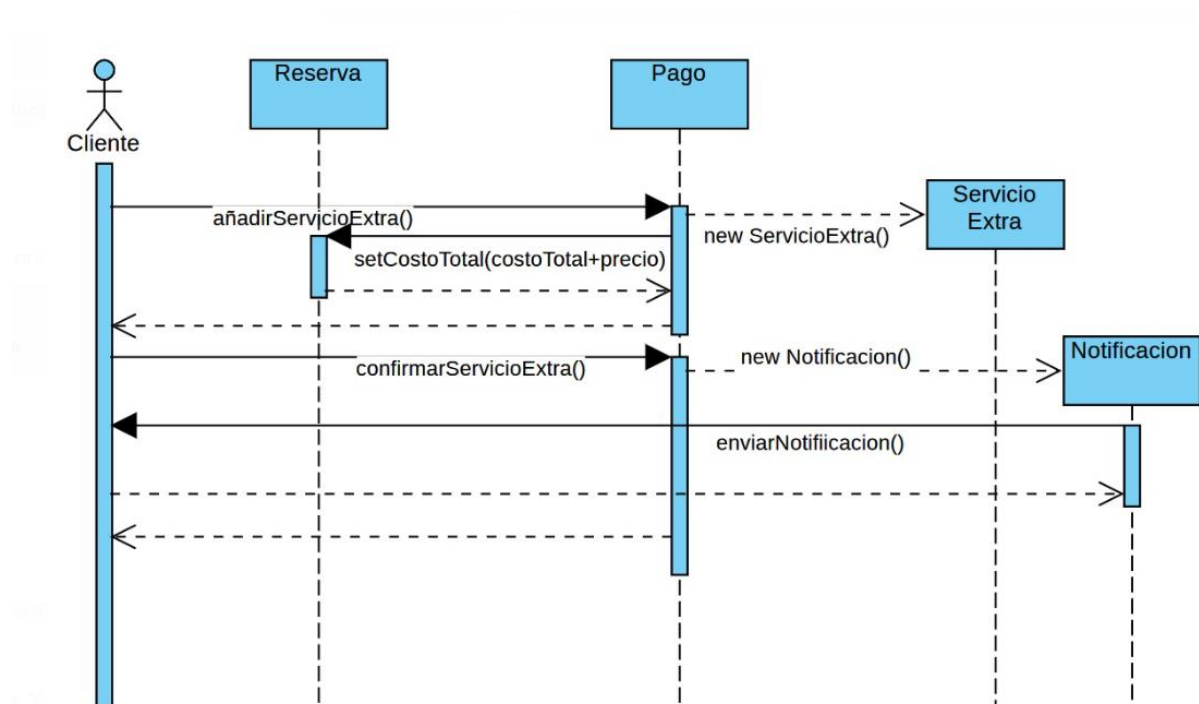
2. Diagrama De Secuencia #2



3. Diagrama De Secuencia #3



4. Diagrama De Secuencia #4



Sección D

Github: <https://github.com/salva1984/tarea1.git>