

**Universidad Tecnológica de Panamá**

**Facultad de Ingeniería en Sistemas Computacionales**

**Ingeniería en Sistemas de Información**

**Ingeniería de Software Aplicada**

**Proyecto Final**

**ÜPay: Aplicación de pago móvil.**

**Facilitadora**: Ing. Jeanette Riley

**Integrantes**

Johel Batista 8-914-587

Sebastián Maldonado 8-984-1276

Alessandro Vergara 8-971-1446

Mariana Zabala 20-70-5218

**I Semestre Académico**

**2023**

**Índice de Contenidos**

[**Introducción** 4](#_Toc140698672)

[**Matriz de Trazabilidad** 6](#_Toc140698673)

[**Diagrama de Casos de Uso** 10](#_Toc140698674)

[**Especificación de Casos de Uso** 11](#_Toc140698675)

[**Especificación de Caso de Uso: Validar Perfil** 11](#_Toc140698676)

[**Especificación de Caso de Uso: Informar sobre Servicios** 13](#_Toc140698677)

[**Especificación de Caso de Uso: Notificación de Pagos Realizados y Pendientes** 16](#_Toc140698678)

[**Especificación de Caso de Uso: Verificación de Identidad** 19](#_Toc140698679)

[**Especificación de Caso de Uso: Mostrar Historial de Transacciones del Usuario** 22](#_Toc140698680)

[**Escenarios de Casos de Uso** 25](#_Toc140698681)

[**Caso de Uso: Validar Perfil** 25](#_Toc140698682)

[**Caso de Uso: Informar sobre Servicios** 26](#_Toc140698683)

[**Caso de Uso: Notificación de Pagos Realizados y Pendientes** 27](#_Toc140698684)

[**Caso de Uso: Verificación de Identidad** 28](#_Toc140698685)

[**Caso de Uso: Mostrar Historial de Transacciones del Usuario** 29](#_Toc140698686)

[**Prototipos de Pantallas** 30](#_Toc140698687)

[**Caso de Uso: Validar Perfil** 30](#_Toc140698688)

[**Caso de Uso: Informar sobre Servicios** 31](#_Toc140698689)

[**Caso de Uso: Notificación de Pagos Realizados y Pendientes** 33](#_Toc140698690)

[**Caso de Uso: Verificación de Identidad** 34](#_Toc140698691)

[**Caso de Uso: Mostrar Historial de Transacciones** 35](#_Toc140698692)

[**Arquitectura del Sistema** 36](#_Toc140698693)

[**Descripción de la Arquitectura** 36](#_Toc140698694)

[**Vista Casos de Uso** 38](#_Toc140698695)

[**Caso de Uso: Validar Perfil** 38](#_Toc140698696)

[**Escenario #1: Validar Perfil (Exitoso)** 38](#_Toc140698697)

[**Escenario #2: Validar Perfil (Fallido)** 40](#_Toc140698698)

[**Caso de Uso: Informar sobre Servicios** 42](#_Toc140698699)

[**Escenario #1: Informar sobre Servicios (Exitoso)** 42](#_Toc140698700)

[**Escenario #2: Informar sobre Servicios (Fallido)** 44](#_Toc140698701)

[**Caso de Uso: Notificación de Pagos Realizados y Pendientes** 46](#_Toc140698702)

[**Escenario #1: Notificación de Pagos Realizados y Pendientes (Exitoso)** 46](#_Toc140698703)

[**Escenario #1: Notificación de Pagos Realizados y Pendientes (Fallido)** 48](#_Toc140698704)

[**Caso de Uso: Verificación de Identidad** 50](#_Toc140698705)

[**Escenario #1: Verificación de Identidad (Exitoso)** 50](#_Toc140698706)

[**Escenario #2: Verificación de Identidad (Fallido)** 52](#_Toc140698707)

[**Caso de Uso: Mostrar Historial de Transacciones** 55](#_Toc140698708)

[**Escenario #1: Mostrar Historial de Transacciones (Exitoso)** 55](#_Toc140698709)

[**Escenario #2: Mostrar Historial de Transacciones (Fallido)** 57](#_Toc140698710)

[**Vistas de la Aplicación** 59](#_Toc140698711)

[**Vista Lógica** 59](#_Toc140698712)

[**Vista de Despliegue** 60](#_Toc140698713)

[**Vista Puesta en Marcha** 61](#_Toc140698714)

[**Vista de Procesos** 62](#_Toc140698715)

[**Vista de Datos** 63](#_Toc140698716)

[**Casos de Prueba** 64](#_Toc140698717)

[**Caso de Uso:** Validar Perfil 65](#_Toc140698718)

[**Caso de Uso:** Informar sobre Servicios 66](#_Toc140698719)

[**Caso de Uso:** Notificación de Pagos Realizados y Pendientes 67](#_Toc140698720)

[**Caso de Uso:** Verificación de Identidad 68](#_Toc140698721)

[**Caso de Uso:** Mostrar Historial de Transacciones del Usuario 69](#_Toc140698722)

[**Lecciones Aprendidas** 74](#_Toc140698723)

[**Referencias Bibliográficas** 76](#_Toc140698724)

[**Anexos** 77](#_Toc140698725)

[**Glosario de Términos** 77](#_Toc140698726)

# **Introducción**

El principal objetivo de este proyecto reside en la exploración exhaustiva y la aplicación de las técnicas, procesos y conocimientos fundamentales en el campo de la Ingeniería de Software dentro de un contexto real y palpable.

Para lograr esto, haremos uso de una serie de herramientas y técnicas claves en la ingeniería de software.

Las herramientas incluyen diagramas de clases, que nos proporcionan un método visual para representar las relaciones entre las diversas partes del software. Utilizaremos casos de uso para definir las interacciones entre el software y sus usuarios, esclareciendo qué puede hacer cada tipo de usuario con el sistema.

Los diagramas de secuencia y colaboración nos permitirán describir cómo se comunican los diferentes componentes del software para realizar una tarea.

Para garantizar que mantenemos el enfoque del proyecto y que comprendemos completamente su alcance, también implementaremos una matriz de trazabilidad.

Esta matriz nos ayudará a seguir el rastro de cada requisito a lo largo del ciclo de vida del desarrollo, asegurándonos de que todas las necesidades estén cubiertas y que el proyecto esté en camino.

A lo largo del proyecto, describiremos de manera meticulosa y rigurosa el software, de tal forma que se especifique con precisión el comportamiento esperado del mismo.

Además, presentaremos escenarios detallados de los casos de uso, que son representaciones de situaciones específicas que podrían ocurrir durante el uso normal del software. Cada escenario será acompañado de prototipos de pantallas, proporcionando una visión más clara de la interacción del usuario con el sistema.

Uno de los elementos fundamentales de nuestro enfoque para este proyecto es la adopción de la arquitectura 4+1 de Kruchten. Este modelo de arquitectura se utiliza para describir un sistema desde cinco diferentes puntos de vista, o 'vistas'.

Estas vistas son: lógica, de desarrollo, de proceso, física y de escenarios. Cada una de estas vistas aborda un conjunto específico de preocupaciones de los stakeholders del sistema.

* La vista lógica se centra en la funcionalidad que el sistema proporciona a sus usuarios. La vista de desarrollo describe el sistema desde la perspectiva de un programador y cómo se organiza el código.
* La vista de proceso se ocupa del rendimiento, la escalabilidad y otros aspectos no funcionales.
* La vista física representa el sistema desde la perspectiva de la infraestructura, mostrando cómo se mapea el software en el hardware.
* La vista de escenarios, también conocida como vista de casos de uso, se utiliza para describir secuencias de interacciones que ocurren entre los usuarios y el sistema.

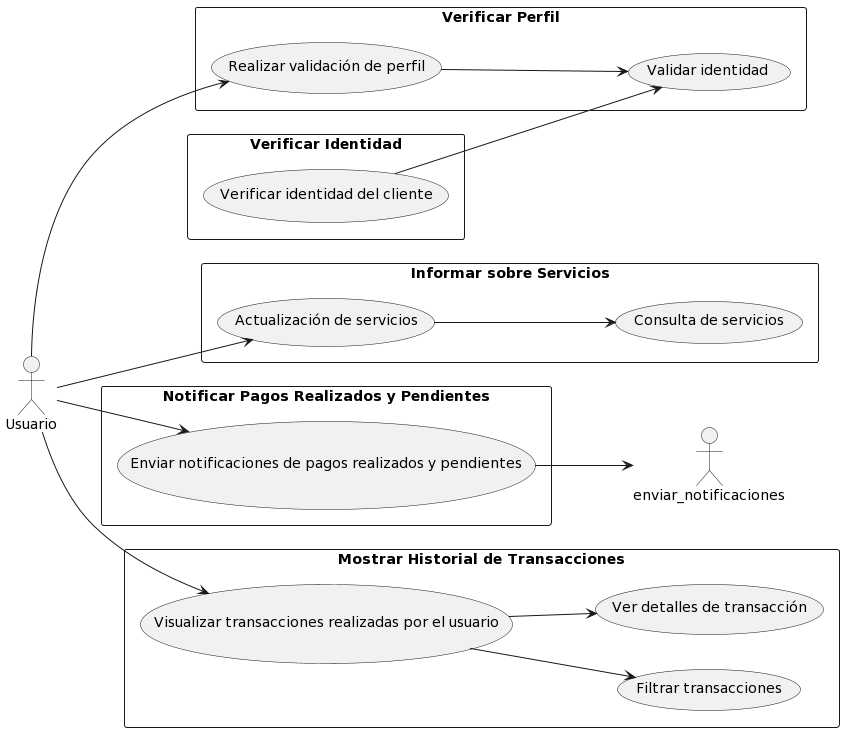
El uso de la arquitectura 4+1 nos ayudará a identificar y abordar las soluciones a las preocupaciones de los Stakeholders que impactan en la organización. La arquitectura también nos ayudará a crear un software eficaz que pueda manejar problemas como la falta de planificación de actividades y la necesidad de almacenar registros de ingresos y gastos.

Finalmente, como parte esencial de nuestro proceso de desarrollo, presentaremos casos de prueba para los casos de uso, garantizando que el software funcione correctamente y que cumpla con las especificaciones establecidas.

# **Matriz de Trazabilidad**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Necesidad** | **Característica** | **Procesos** | **Caso de Uso** | **Requerimientos No Funcionales** |
| Verificación de perfil de usuario para registro en aplicación. | Realizar una validación de perfil en el primer registro de usuario mediante el uso de cédula y foto de rostro. | Realizar el primer ingreso en la plataforma, con información personal (foto personal, foto de cédula, correo electrónico institucional, copia de recibo de pago de matrícula). Realizar doble autenticación con un código de 6 dígitos enviado al correo electrónico institucional. | **Validar Perfil** | **Seguridad**: La información personal del usuario debe estar segura y protegida. |
| Informar sobre los servicios que ofrece la universidad actualmente. | Actualización de servicios  Consulta de servicios | Poner a disposición una sección informativa sobre las funcionalidades del Sistema.  Mostrar al usuario nuevas actualizaciones que se realicen en el sistema.  Obtener información sobre los procesos que involucran pagos dentro la universidad (pagos en cafetería, pago de matrícula, pago en las librerías del complejo, pago en biblioteca, centro de copiado). | **Informar sobre Servicios** | **Usabilidad**: La interfaz de usuario debe ser fácil de navegar para permitir a los usuarios encontrar rápidamente la información que necesitan. |
| Notificación de pagos realizados y pendientes Recordatorios de pago para los usuarios. | Recordar a los usuarios sus pagos pendientes. | Brindar recordatorios a cada usuario sobre sus pagos pendientes, así pueden evitar recargos por morosidad.  Enviar notificaciones instantáneas al realizar transacciones (estudiantes, profesores y administrativos) | **Notificar sobre Pagos Realizados y Pendientes** | **Rendimiento**: Las notificaciones deben ser enviadas en tiempo real para mantener a los usuarios al día. |
| Verificación de identidad del cliente. | Verificar a través de datos biométricos o el uso de pines la identidad de la persona, y el uso correcto y apropiado de su herramienta de pago. | Antes de que el pago sea procesado, la persona en cuestión deberá proporcionar al sistema una verificación de su identidad, a través del uso de tecnología biométrica (huella dactilar) o una contraseña, una vez que el sistema verifique la identidad del usuario exitosamente se podrá hacer el cobro del servicio. | **Verificación de Identidad** | **Seguridad**: Los datos biométricos y las contraseñas deben estar encriptados para proteger la identidad del usuario. |
| Los usuarios necesitan una forma fácil y segura de rastrear todas las transacciones realizadas en la plataforma | Visualización de todas las transacciones realizadas por el usuario en la plataforma.  Función de filtro para buscar transacciones específicas.  Opción para ver detalles específicos de cada transacción.  Capacidad para descargar recibos de transacciones exitosas. | Autenticación del usuario  Selección de la opción "Historial de Transacciones" por parte del usuario  Recuperación de todas las transacciones del usuario de la base de datos  Presentación de la lista de transacciones al usuario  Uso del filtro de búsqueda por parte del usuario para localizar transacciones específicas  Selección de una transacción específica para ver los detalles completos  Descarga de un recibo para una transacción exitosa  Regreso a la lista de transacciones o salida de la sección de historial por parte del usuario. | **Mostrar Historial de Transacciones de Usuario** | El sistema debe ser seguro y proteger la información financiera del usuario.  El sistema debe ser confiable, capaz de recuperar y mostrar todas las transacciones realizadas por el usuario.  El sistema debe ser fácil de usar, con una interfaz clara que permita a los usuarios navegar y utilizar las funciones sin dificultad.  El sistema debe ser eficiente y capaz de recuperar y mostrar la información de la transacción rápidamente.  El sistema debe ser compatible con varias plataformas (como desktop, mobile, etc.) para permitir el acceso de los usuarios en cualquier momento y lugar. |

## **Diagrama de Casos de Uso**



**Ilustración #1**: Diagrama de Casos de Uso a partir de la Matriz de Trazabilidad

# **Especificación de Casos de Uso**

## **Especificación de Caso de Uso: Validar Perfil**

1. **Validar Perfil**
   1. **Breve Descripción**
2. Este caso de uso permite a los usuarios (Administradores del sistema y usuarios registrados) en una plataforma de pago en línea validar su perfil mediante la verificación de su identidad.
3. **Flujo de Eventos**
   1. **Flujo Básico**
      1. El usuario registrado inicia sesión en la plataforma.
      2. El sistema verifica la autenticidad de las credenciales.
      3. El sistema muestra el perfil del usuario.
      4. El sistema revisa y valida el perfil del usuario si cumple con los requisitos establecidos.
      5. El sistema envía una notificación al usuario informando que su perfil ha sido validado.
   2. **Flujo de Excepciones**
      1. Si el perfil del usuario no cuenta con el parámetro de tipo de usuario, el sistema rechaza la validación.
      2. El sistema envía una notificación al usuario informando el rechazo y la razón de este.
      3. El usuario realiza las correcciones necesarias en su perfil y solicita nuevamente su validación.
      4. El administrador del sistema repite el proceso de revisión y validación.
   3. **Flujo Alternativo**
      1. Si el usuario desea validar su perfil, pero olvidó algunos detalles de su cuenta, el sistema puede dirigir al usuario a la funcionalidad "Recuperar Contraseña" o "Recuperar Nombre de Usuario".
      2. Si el usuario tiene problemas con la verificación de identidad durante el proceso de validación, el sistema puede ofrecer alternativas como la verificación a través de correo electrónico o un número de teléfono asociado a la cuenta.
4. **Requerimientos Especiales**
   1. Implementar medidas de autenticación de usuarios y encriptación de datos.
   2. El sistema debe manejar grandes cantidades de usuarios y perfiles permitiendo un acceso rápido y eficiente a los servicios y funcionalidades de la plataforma.
5. **Precondiciones**
   1. El usuario debe estar registrado en la plataforma.
   2. El usuario debe haber creado un perfil en la plataforma.
6. **Postcondiciones**
   1. El sistema muestra en pantalla la notificación de perfil validado o rechazado.
7. **Puntos de Ampliación**
   1. Durante el proceso de validación, el sistema podría implementar el caso de uso "Verificar Identidad".
   2. El sistema puede tener la capacidad de "Contactar al Cliente" para informar sobre la validación o el rechazo del perfil.

## **Especificación de Caso de Uso: Informar sobre Servicios**

1. **Informar sobre Servicios**
   1. **Breve Descripción**
2. Este caso de uso describe el proceso en el que la plataforma, con base en la información de los servicios de la universidad, informa al usuario (Estudiante, Profesor, Administrativo) de las características, novedades y cambios recientes en dichos servicios.
3. **Flujo de Eventos**
   1. **Flujo Básico**
      1. El usuario (Estudiante, Profesor o Administrativo) inicia sesión en la plataforma utilizando sus credenciales de acceso.
      2. Una vez autenticado, el usuario navega hasta la opción “Informar sobre Servicios” en el menú principal de la plataforma.
      3. Al seleccionar esta opción, el sistema realiza una consulta a la base de datos para extraer la información más reciente y relevante sobre los servicios de la universidad.
      4. El sistema procesa esta información y la presenta en una lista estructurada en la interfaz del usuario, categorizando los servicios y destacando las actualizaciones o cambios recientes.
      5. El usuario puede explorar la lista de servicios, pudiendo filtrar o buscar servicios específicos si lo desea.
      6. Al seleccionar un servicio específico, el sistema realiza una nueva consulta a la base de datos para obtener una descripción detallada de este servicio, incluyendo cualquier cambio o actualización reciente.
      7. El sistema presenta esta información al usuario en un formato fácil de entender, permitiéndole tomar decisiones informadas basadas en las características y cambios del servicio.
      8. Después de leer los detalles del servicio, el usuario puede optar por salir seleccionando la opción de finalizar, regresando al menú principal y completando el caso de uso.
   2. **Flujo de Excepciones**
      1. Si el usuario no puede autenticarse en la plataforma debido a credenciales incorrectas, el sistema muestra un mensaje de error y pide al usuario que reintente el inicio de sesión.
      2. Si la plataforma está en mantenimiento o hay un error con la base de datos durante la consulta de la información de los servicios, el sistema muestra un mensaje de error informando al usuario que el servicio no está disponible en ese momento y que se debe intentar más tarde.
      3. Si la información detallada de un servicio específico no puede ser recuperada por cualquier razón, el sistema informa al usuario de la situación y le pide que intente más tarde o que seleccione otro servicio para obtener detalles.
      4. Si ocurre un error durante el proceso de finalización del caso de uso, el sistema informa al usuario del error y proporciona opciones para corregirlo o reportarlo.
   3. **Flujo Alternativo**
      1. Si el usuario selecciona la opción "Suscribirse a las actualizaciones del servicio" mientras ve los detalles de un servicio específico, el sistema añade este servicio a la lista de suscripciones del usuario y le notifica sobre cualquier cambio o actualización en el futuro.
      2. Si el usuario selecciona la opción "Marcar como favorito" mientras ve los detalles de un servicio específico, el sistema añade este servicio a la lista de favoritos del usuario para un acceso más rápido en el futuro.
      3. Si el sistema detecta que hay una gran cantidad de servicios a mostrar, puede optar por presentarlos de manera paginada o categorizada para facilitar la navegación del usuario.
4. **Requerimientos Especiales**
   1. El sistema debe asegurarse de que la información presentada sea la más reciente y relevante.
   2. Los servicios deben estar categorizados para facilitar la búsqueda y selección por parte del usuario.
   3. El usuario debe estar autenticado en la plataforma.
5. **Precondiciones**
   1. El usuario debe estar registrado y autenticado en la plataforma.
   2. La información sobre los servicios debe estar disponible y actualizada en la base de datos del sistema.
6. **Postcondiciones**
   1. El usuario ha sido informado sobre los servicios y sus actualizaciones.
   2. Si el usuario se suscribió a las actualizaciones de un servicio, será notificado sobre cualquier cambio o actualización en el futuro.
7. **Puntos de Ampliación**
   1. El sistema podría implementar una funcionalidad de notificaciones, donde los usuarios reciben actualizaciones de servicios directamente en sus correos electrónicos.
   2. La plataforma podría implementar una funcionalidad de suscripción, donde los usuarios pueden suscribirse a actualizaciones de servicios específicos.

## **Especificación de Caso de Uso: Notificación de Pagos Realizados y Pendientes**

1. **Notificación de Pagos Realizados y Pendientes**
   1. **Breve Descripción**
2. Este caso de uso describe el proceso en el que la plataforma informa al usuario (Estudiante, Profesor, Administrativo) sobre sus pagos realizados y pendientes. El objetivo es ayudar al usuario a mantener un control efectivo de sus transacciones financieras y recordarle el cumplimiento oportuno de sus obligaciones financieras para evitar cargos adicionales por morosidad.
3. **Flujo de Eventos**
   1. **Flujo Básico**
      1. Una vez autenticado en la plataforma, el sistema recoge la información del perfil del usuario y accede a su historial de pagos.
      2. El sistema, basándose en el historial de pagos del usuario, determina si hay pagos realizados y/o pendientes.
      3. Si hay pagos pendientes, el sistema calcula la fecha límite para realizar el pago sin recargos y la fecha en la que se aplicaría el recargo.
      4. El sistema genera una notificación para el usuario con la información de los pagos realizados y pendientes, incluyendo las fechas relevantes en caso de pagos pendientes.
      5. Esta notificación se muestra en la plataforma y se envía por correo electrónico al usuario.
      6. El usuario revisa las notificaciones y toma las acciones correspondientes.
      7. Si el usuario no realiza el pago dentro del plazo establecido, el sistema aplica un recargo del 25% al monto pendiente y actualiza la notificación con esta información (punto 2.1.4).
      8. Si el usuario realiza el pago, el sistema actualiza la información de los pagos y finaliza este caso de uso.
   2. **Flujo de Excepciones**
      1. Si el sistema encuentra errores durante la recopilación de la información de pagos del usuario (por ejemplo, problemas con la conexión a la base de datos), mostrará un mensaje de error al usuario y pedirá que se intente más tarde.
      2. Si el usuario no ha habilitado la opción de recibir notificaciones, el sistema no enviará la notificación, pero mostrará la información en la plataforma.
      3. Si el sistema no puede enviar la notificación por correo electrónico (por ejemplo, si el correo electrónico del usuario no es válido o si hay problemas con el servidor de correo), mostrará un mensaje de error al usuario y le pedirá que revise y actualice su información de contacto.
      4. Si el usuario no realiza el pago dentro del plazo establecido y el sistema no puede aplicar el recargo por alguna razón (por ejemplo, problemas con la base de datos), el sistema mostrará un mensaje de error al usuario y pedirá que se intente más tarde.
   3. **Flujo Alternativo**
      1. En caso de que el usuario tenga un pago programado, el sistema verifica el estado del pago en la fecha programada.
      2. Si el pago programado se realiza con éxito, el sistema actualiza la información de pagos y envía una notificación al usuario confirmando la transacción.
      3. Si el pago programado falla (por ejemplo, debido a un problema con el método de pago), el sistema envía una notificación al usuario informándole del fallo y sugiere acciones correctivas.
4. **Requerimientos Especiales**
   1. El usuario debe tener una cuenta registrada y verificada en la plataforma.
   2. El usuario debe haber habilitado la opción de recibir notificaciones.
   3. El sistema debe poder acceder a la información de los pagos del usuario y calcular las fechas relevantes y los recargos de forma precisa.
5. **Precondiciones**
   1. El usuario debe tener una cuenta en la plataforma y haber iniciado sesión.
   2. La cuenta del usuario debe tener al menos un pago registrado (realizado o pendiente).
6. **Postcondiciones**
   1. El usuario recibirá información actualizada y precisa sobre sus pagos realizados y pendientes.
   2. Si hay pagos pendientes, el usuario estará informado de las fechas límites y posibles recargos.
7. **Puntos de Ampliación**
   1. El sistema podría implementar una funcionalidad de recordatorios personalizados, donde los usuarios pueden configurar la frecuencia y el formato de las notificaciones según sus preferencias.
   2. Se podría añadir una opción para que el usuario realice el pago directamente desde la notificación, mejorando así la experiencia de usuario y aumentando la probabilidad de un pago oportuno.

## **Especificación de Caso de Uso: Verificación de Identidad**

1. **Verificación de Identidad**
   1. **Breve Descripción**
2. Este caso de uso describe el proceso por el cual la plataforma verifica la identidad del usuario antes de autorizar una transacción, ya sea a través de datos biométricos o un PIN. Este proceso de verificación contribuye a prevenir el fraude y asegurar la privacidad del usuario.
3. **Flujo de Eventos**
   1. **Flujo Básico**
      1. Una vez autenticado, el usuario selecciona la opción "Verificación de Identidad" en la interfaz de la plataforma.
      2. El sistema solicita al usuario que se identifique a través de datos biométricos (como una huella dactilar) o introduciendo su PIN personal.
      3. El usuario proporciona los datos biométricos o introduce su PIN.
      4. El sistema procesa la información proporcionada por el usuario y la compara con la información guardada en su base de datos.
      5. Si la información proporcionada por el usuario coincide con la información almacenada, el sistema autoriza la transacción y notifica al usuario del éxito de la operación.
      6. Si la información no coincide, el sistema bloquea la transacción y envía una notificación al usuario informándole del fallo en la verificación. El usuario tiene la opción de intentar de nuevo o de recuperar/restablecer su PIN o datos biométricos.
   2. **Flujo de Excepciones**
      1. Si el sistema detecta un intento de acceso no autorizado (por ejemplo, introducción repetida de un PIN incorrecto o lecturas biométricas fallidas), el sistema bloquea la cuenta del usuario por seguridad y envía una notificación al usuario para informarle de la situación.
      2. Si el sistema no puede procesar los datos biométricos o el PIN debido a problemas técnicos (por ejemplo, una interrupción del servicio, un fallo en el hardware de lectura de huellas dactilares), el sistema muestra un mensaje de error al usuario y le pide que intente más tarde o que se ponga en contacto con el soporte técnico.
   3. **Flujo Alternativo**
      1. Si el usuario olvida su PIN o no puede proporcionar datos biométricos, puede seleccionar la opción "Recuperar PIN" o "Preguntas de seguridad".
      2. El sistema solicita al usuario que responda a las preguntas de seguridad preestablecidas o que confirme su dirección de correo electrónico para recuperar su PIN.
      3. Si las respuestas a las preguntas de seguridad son correctas o si el usuario confirma correctamente su correo electrónico, el sistema permite al usuario restablecer su PIN.
      4. El sistema solicita al usuario que establezca un nuevo PIN.
      5. El usuario introduce un nuevo PIN y lo confirma.
      6. El sistema actualiza el PIN en la base de datos y notifica al usuario que su PIN ha sido cambiado con éxito.
4. **Requerimientos Especiales**
   1. El usuario debe haber registrado previamente sus datos biométricos o su PIN en la plataforma.
   2. El sistema debe cumplir con las normativas de seguridad y privacidad vigentes, incluyendo el cifrado de los datos del usuario y la protección contra el acceso no autorizado.
5. **Precondiciones**
   1. El usuario debe tener una cuenta en la plataforma y debe haber iniciado sesión.
   2. El usuario debe haber configurado su método de verificación de identidad (datos biométricos o PIN).
6. **Postcondiciones**
   1. Si la verificación de identidad es exitosa, la transacción se autoriza y se registra en el historial de transacciones del usuario.
   2. Si la verificación de identidad no es exitosa, la transacción se bloquea y el usuario es notificado.
7. **Puntos de Ampliación**
   1. En casos en los que la verificación biométrica no sea posible (por ejemplo, debido a hardware inadecuado), el sistema podría permitir la verificación de identidad a través de preguntas de seguridad personales.
   2. Para mejorar la seguridad, el sistema podría implementar una autenticación de dos factores, en la que el usuario deba proporcionar dos formas de identificación (por ejemplo, un PIN y una respuesta a una pregunta de seguridad).

## **Especificación de Caso de Uso: Mostrar Historial de Transacciones del Usuario**

1. **Mostrar Historial de Transacciones del Usuario.**
   1. **Breve Descripción**
2. Este caso de uso permite a un usuario autenticado visualizar todas las transacciones que ha realizado en la plataforma. Puede ver detalles específicos de cada transacción y, en caso de transacciones pendientes o fallidas, puede tomar medidas para resolverlas.
3. **Flujo de Eventos**
   1. **Flujo Básico**
      1. Una vez autenticado, el usuario selecciona la opción "Historial de Transacciones" en la interfaz de la plataforma.
      2. El sistema recupera todas las transacciones del usuario de la base de datos.
      3. El sistema presenta la lista de transacciones al usuario.
      4. El usuario puede utilizar un filtro de búsqueda para localizar transacciones específicas.
      5. Al seleccionar una transacción de la lista, el sistema presenta los detalles completos de dicha transacción.
      6. Si la transacción fue exitosa, el sistema ofrece la opción de descargar un recibo.
      7. El usuario puede optar por regresar a la lista de transacciones o salir de la sección de historial.
   2. **Flujo de Excepciones**
      1. Si durante el proceso el sistema encuentra un error al intentar recuperar el historial de transacciones, el sistema genera un mensaje de error.
      2. Si el usuario intenta descargar un recibo de una transacción que está pendiente o que ha fallado, el sistema no puede generar el recibo.
      3. Si el usuario intenta filtrar el historial de transacciones con parámetros que no corresponden a ninguna transacción, el sistema muestra un mensaje.
      4. Si el usuario intenta acceder a los detalles de una transacción que ya no existe o ha sido borrada de la base de datos, el sistema muestra un mensaje.
   3. **Flujo Alternativo**
      1. El usuario puede decidir ordenar la lista de transacciones por una columna específica (por ejemplo, fecha, monto, estado).
      2. El sistema ordena la lista de transacciones de acuerdo con la columna seleccionada.
      3. Si el usuario desea regresar al orden original de la lista, selecciona la opción de eliminar ordenamiento. El sistema regresa la lista de transacciones a su estado original.
      4. En caso de que la plataforma cuente con una opción para exportar el historial de transacciones, el usuario puede seleccionar exportar historial de transacciones y el sistema genera un archivo descargable con los datos.
      5. En caso de que el usuario decida buscar una transacción específica, puede utilizar la opción de búsqueda e ingresar los datos de la transacción a buscar. El sistema filtra los resultados y muestra las transacciones que coinciden con la búsqueda.
4. **Requerimientos Especiales**
   1. El sistema debe ser capaz de filtrar las transacciones por fecha, monto, tipo y estado.
   2. El sistema debe ser capaz de generar y descargar recibos de las transacciones.
5. **Precondiciones**
   1. El usuario debe estar autenticado en la plataforma.
   2. El usuario debe haber realizado al menos una transacción en la plataforma.
6. **Postcondiciones**
   1. El usuario ha visto el historial completo de sus transacciones.
   2. El usuario ha visto los detalles de una transacción específica.
   3. El usuario ha tomado medidas sobre transacciones pendientes o fallidas, si es aplicable.
   4. El usuario ha descargado un recibo de una transacción, si lo deseaba.
7. **Puntos de Ampliación**
   1. El sistema podría permitir al usuario marcar transacciones como favoritas para un acceso más rápido en el futuro.
   2. El sistema podría proporcionar estadísticas y gráficos basados en el historial de transacciones del usuario.
   3. El sistema podría permitir al usuario programar alertas o recordatorios basados en sus patrones de transacciones.

# **Escenarios de Casos de Uso**

## **Caso de Uso: Validar Perfil**

**Escenario 1: Validación de Perfil (Exitoso)**

Juan inicia sesión en la plataforma y su perfil es verificado y validado con éxito por el sistema. Recibe una notificación informándole que su perfil ha sido validado satisfactoriamente. Se siente satisfecho y puede acceder a todas las funciones de la plataforma.

**Tiempo estimado:** Aproximadamente 2 minutos.

**Escenario 2: Validación de Perfil (Fallido)**

Juan inicia sesión en la plataforma, pero su perfil no cumple con los requisitos establecidos. Recibe una notificación indicándole los detalles del rechazo. Decide corregir los errores en su perfil y solicita nuevamente la validación.

El administrador revisa las correcciones realizadas por Juan y, después de un tiempo, se notifica a Juan si su perfil ha sido validado o rechazado nuevamente. En caso de ser validado, se siente aliviado y puede acceder a todas las funciones de la plataforma. En caso de ser rechazado nuevamente, Juan realiza las correcciones necesarias y solicita una nueva validación.

**Tiempo estimado:** Aproximadamente 5 minutos.

## **Caso de Uso: Informar sobre Servicios**

**Escenario 1: Informar sobre Servicios (Exitoso)**

El usuario, Juan, inicia sesión en la plataforma utilizando sus credenciales y accede a la opción "Informar sobre Servicios" en el menú principal. El sistema consulta la base de datos y recupera la información más reciente y relevante sobre los servicios de la universidad. Presenta la lista estructurada de servicios en la interfaz de Juan, resaltando las actualizaciones y cambios recientes. Juan explora la lista, puede aplicar filtros y realizar búsquedas para encontrar servicios específicos de su interés. Al seleccionar un servicio en particular, el sistema muestra una descripción detallada, incluyendo cualquier cambio o actualización reciente. Después de leer los detalles, Juan finaliza el caso de uso y regresa al menú principal de la plataforma.

**Tiempo estimado**: Aproximadamente 3 minutos.

**Escenario 2: Informar sobre Servicios (Fallido)**

El usuario, Juan, inicia sesión en la plataforma, pero no puede autenticarse debido a credenciales incorrectas. El sistema muestra un mensaje de error y solicita a Juan que reintente el inicio de sesión. Después de ingresar las credenciales correctas, Juan accede a la opción "Informar sobre Servicios" en el menú principal. Sin embargo, durante la consulta de la información de los servicios, el sistema encuentra un error en la base de datos o la plataforma está en mantenimiento. Se muestra un mensaje de error informando a Juan que el servicio no está disponible en ese momento y se le sugiere que lo intente nuevamente más tarde. Juan decide salir de la plataforma.

**Tiempo estimado**: Aproximadamente 2 minutos.

## **Caso de Uso: Notificación de Pagos Realizados y Pendientes**

**Escenario 1: Notificación de Pagos Realizados y Pendientes (Exitoso)**

El sistema recopila la información del perfil y el historial de pagos del usuario, identificando pagos realizados y pendientes. Calcula las fechas límite y los posibles recargos para los pagos pendientes. Luego, genera una notificación que muestra al usuario los pagos realizados y pendientes, incluyendo las fechas relevantes. La notificación se muestra en la plataforma y se envía por correo electrónico al usuario. Juan revisa las notificaciones, verifica los pagos y toma las acciones necesarias. Si realiza el pago dentro del plazo establecido, el sistema actualiza la información de los pagos y finaliza el caso de uso.

**Tiempo estimado:** Aproximadamente 3 minutos.

**Escenario 2: Notificación de Pagos Realizados y Pendientes (Fallido)**

El sistema intenta recopilar la información de pagos del usuario, pero encuentra un error durante el proceso (por ejemplo, problemas de conexión a la base de datos). Muestra un mensaje de error al usuario, solicitándole que intente nuevamente más tarde. Juan decide salir de la plataforma sin recibir la notificación de pagos realizados y pendientes.

**Tiempo estimado**: Aproximadamente 1 minuto.

## **Caso de Uso: Verificación de Identidad**

**Escenario 1: Verificación de Identidad (Exitoso)**

El usuario selecciona la opción "Verificación de Identidad" en la plataforma después de haber iniciado sesión. El sistema le solicita al usuario que se identifique proporcionando sus datos biométricos o introduciendo su PIN personal. El usuario elige utilizar su huella dactilar como método de verificación y coloca su dedo en el sensor. El sistema procesa la información biométrica y la compara con los datos almacenados. La verificación es exitosa y el sistema autoriza la transacción. El usuario recibe una notificación indicando que la verificación de identidad ha sido exitosa y puede proceder con la transacción.

**Tiempo estimado**: Aproximadamente 1 minuto.

**Escenario 2: Verificación de Identidad (Fallido)**

El usuario selecciona la opción "Verificación de Identidad" en la plataforma después de haber iniciado sesión. El sistema le solicita al usuario que se identifique proporcionando sus datos biométricos o introduciendo su PIN personal. El usuario intenta introducir su PIN, pero se equivoca varias veces. El sistema detecta múltiples intentos fallidos y bloquea temporalmente la cuenta del usuario por motivos de seguridad. El usuario recibe una notificación indicando que la verificación de identidad ha fallado y que su cuenta ha sido bloqueada temporalmente. Se le proporciona información sobre cómo desbloquear su cuenta y restablecer su PIN. El usuario toma las medidas necesarias para desbloquear su cuenta y restablecer su PIN.

**Tiempo estimado**: Aproximadamente 5 minutos.

## **Caso de Uso: Mostrar Historial de Transacciones del Usuario**

**Escenario 1: Mostrar Historial de Transacciones del Usuario (Exitoso)**

Una vez autenticado en la plataforma, el usuario selecciona la opción "Historial de Transacciones". El sistema recupera todas las transacciones del usuario de la base de datos y presenta la lista completa de transacciones. El usuario utiliza un filtro de búsqueda para localizar una transacción específica y selecciona una transacción de la lista. El sistema muestra los detalles completos de la transacción, incluyendo fecha, monto, tipo y estado. Si la transacción fue exitosa, el sistema ofrece la opción de descargar un recibo, el cual el usuario puede guardar para su referencia. Después de revisar los detalles de la transacción, el usuario decide regresar a la lista de transacciones o salir de la sección de historial.

**Tiempo estimado**: Aproximadamente 2 minutos.

**Escenario 2: Mostrar Historial de Transacciones del Usuario (Fallido)**

Una vez autenticado en la plataforma, el usuario selecciona la opción "Historial de Transacciones". Sin embargo, durante el proceso de recuperación del historial de transacciones, el sistema encuentra un error y muestra un mensaje de error al usuario. El usuario intenta nuevamente, pero el sistema sigue sin poder recuperar el historial debido a un problema técnico. El usuario recibe una notificación indicando que la operación no se puede completar en ese momento y se le sugiere intentar más tarde o ponerse en contacto con el soporte técnico para resolver el problema.

**Tiempo estimado**: Aproximadamente 1 minuto.

# **Prototipos de Pantallas**

## **Caso de Uso: Validar Perfil**

**A screenshot of a phone

Description automatically generated A screenshot of a phone

Description automatically generated A screenshot of a cell phone

Description automatically generated A screen shot of a cell phone

Description automatically generated**

## **Caso de Uso: Informar sobre Servicios**

A screenshot of a phone

Description automatically generated A screenshot of a phone

Description automatically generated A screenshot of a cell phone

Description automatically generated A screenshot of a phone

Description automatically generated

A pink rectangular sign with black text

Description automatically generated A screenshot of a cell phone

Description automatically generated A screenshot of a phone

Description automatically generated

## **Caso de Uso: Notificación de Pagos Realizados y Pendientes**

**A screenshot of a cell phone

Description automatically generated A screenshot of a cell phone

Description automatically generated**

## **Caso de Uso: Verificación de Identidad**

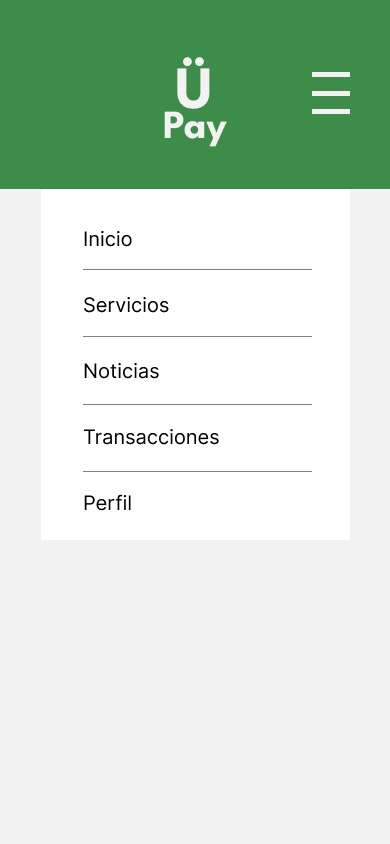
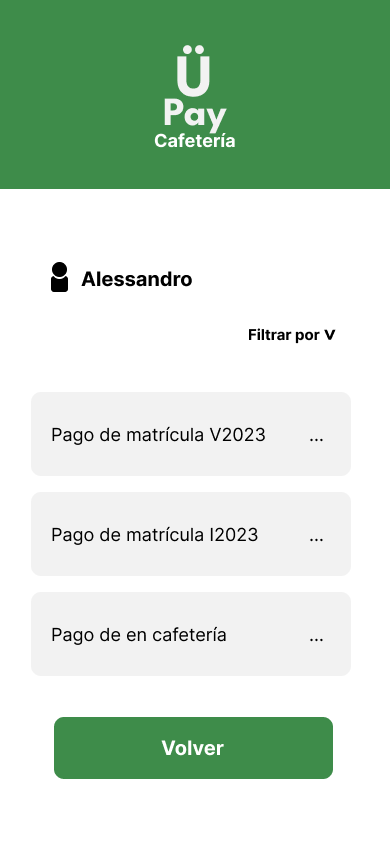
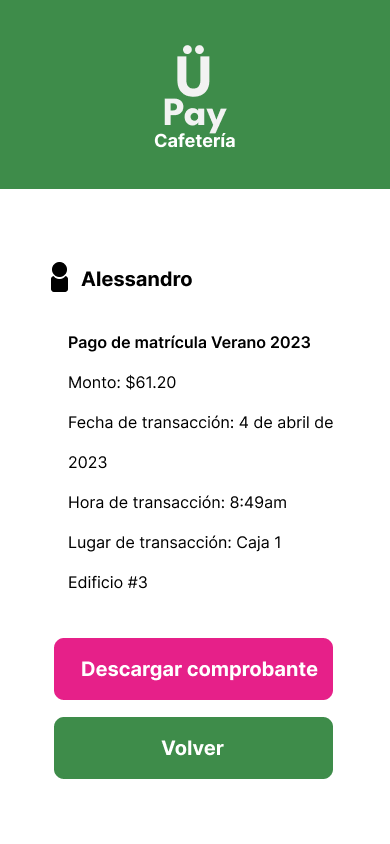
**A screenshot of a computer

Description automatically generated A white background with black text

Description automatically generated A screen shot of a phone

Description automatically generated**

## **Caso de Uso: Mostrar Historial de Transacciones**

**   **

# **Arquitectura del Sistema**

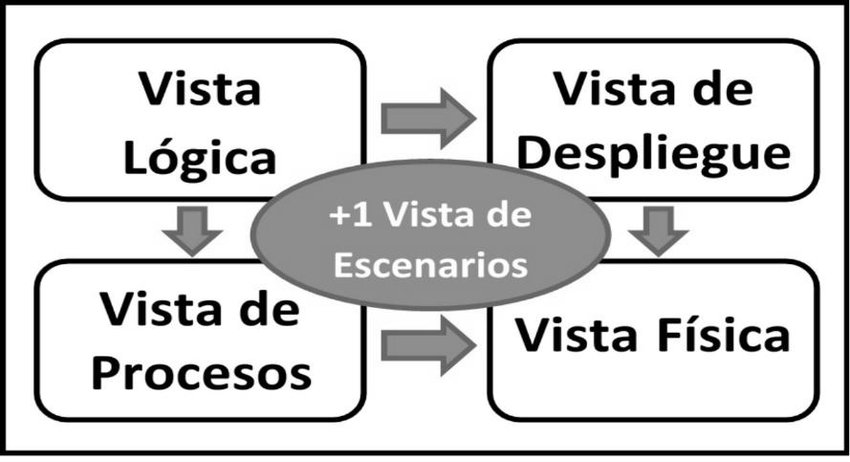
## **Descripción de la Arquitectura**

La arquitectura 4+1 es un modelo conceptual que describe la arquitectura de un sistema de software utilizando cinco vistas concurrentes.

Esta metodología proporciona una manera efectiva de visualizar, comprender y comunicar los aspectos clave de la arquitectura de un sistema, al tiempo que satisface las necesidades y preocupaciones de los diferentes stakeholders involucrados en el proyecto.

En la arquitectura de software, una vista es una representación de un aspecto específico del sistema. Cada vista se enfoca en un conjunto particular de preocupaciones y muestra diferentes aspectos de la arquitectura.

La arquitectura 4+1 se basa en la idea de que múltiples vistas son necesarias para abordar los intereses de los stakeholders de manera separada, lo que resulta en una comprensión más completa y precisa de la arquitectura del sistema.



***Ilustración #2****: Modelo “4+1” vistas*

El modelo 4+1 se compone de cinco vistas concurrentes, cada una de las cuales se representa mediante un diagrama con su notación particular:

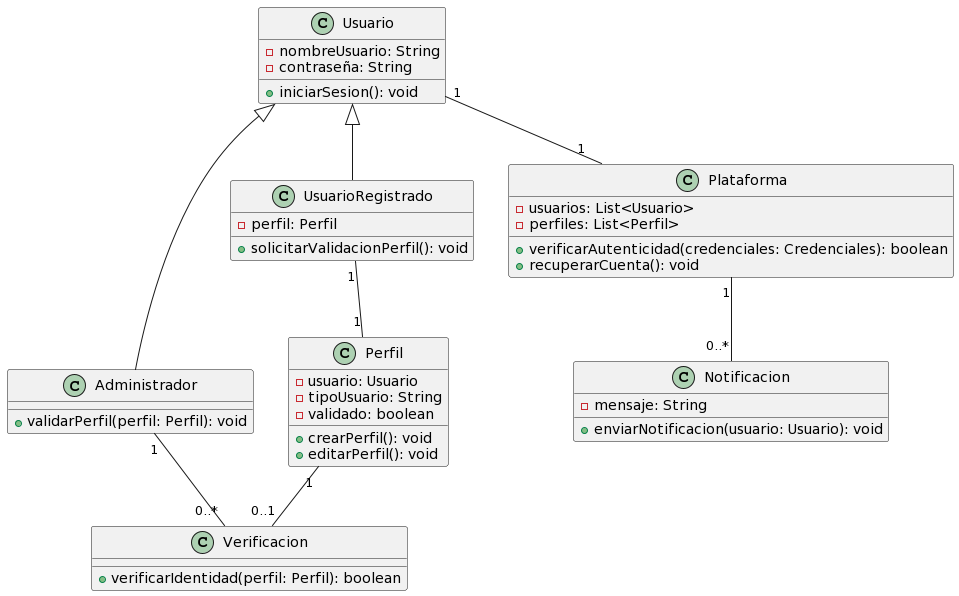
1. **Vista Lógica**: Esta vista describe los módulos lógicos principales del sistema, sus responsabilidades y dependencias. Se centra en la estructura y organización del sistema desde el punto de vista de los diferentes stakeholders, como usuarios finales, desarrolladores o directores de proyecto. Esta vista proporciona una visión conceptual del sistema.
2. **Vista de Despliegue**: Esta vista muestra cómo los componentes de hardware (nodos) del sistema se relacionan entre sí en el entorno de producción. Cada nodo representa un dispositivo de cómputo específico, como un servidor o una estación de trabajo. Esta vista ayuda a comprender la distribución física de los componentes del sistema y su interconexión
3. **Vista de Procesos**: En esta vista se describen los aspectos de concurrencia y sincronización del diseño. Se representan los procesos que existen en el sistema y cómo se comunican entre sí. Esta vista permite comprender cómo se gestionan las tareas concurrentes y cómo se coordinan para lograr un funcionamiento adecuado.
4. **Vista Física**: Esta vista describe el sistema desde la perspectiva del Ingeniero en Sistemas. Se centra en la topología de los componentes físicos del sistema, como servidores, redes y dispositivos de almacenamiento. Esta vista proporciona información sobre la infraestructura física necesaria para soportar la ejecución del sistema.
5. **Vista de Escenarios**: Esta vista describe secuencias de interacciones entre objetos y procesos, utilizando un conjunto de casos de uso. Los escenarios representan situaciones del mundo real en las que el sistema interactúa con sus usuarios o con otros sistemas. Esta vista permite comprender el comportamiento dinámico del sistema y validar su adecuación a los requisitos del usuario.

# **Vista Casos de Uso**

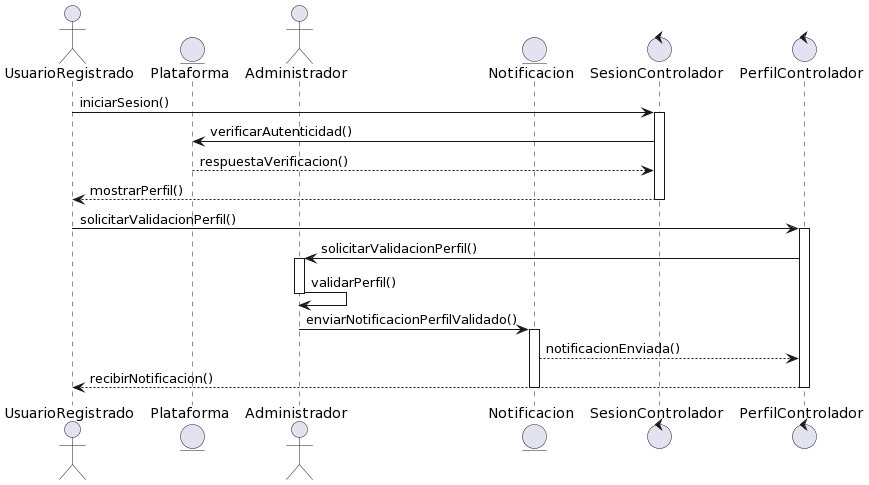
## **Caso de Uso: Validar Perfil**

### **Escenario #1: Validar Perfil (Exitoso)**

#### Diagramas de Clase



#### Diagrama de Secuencia



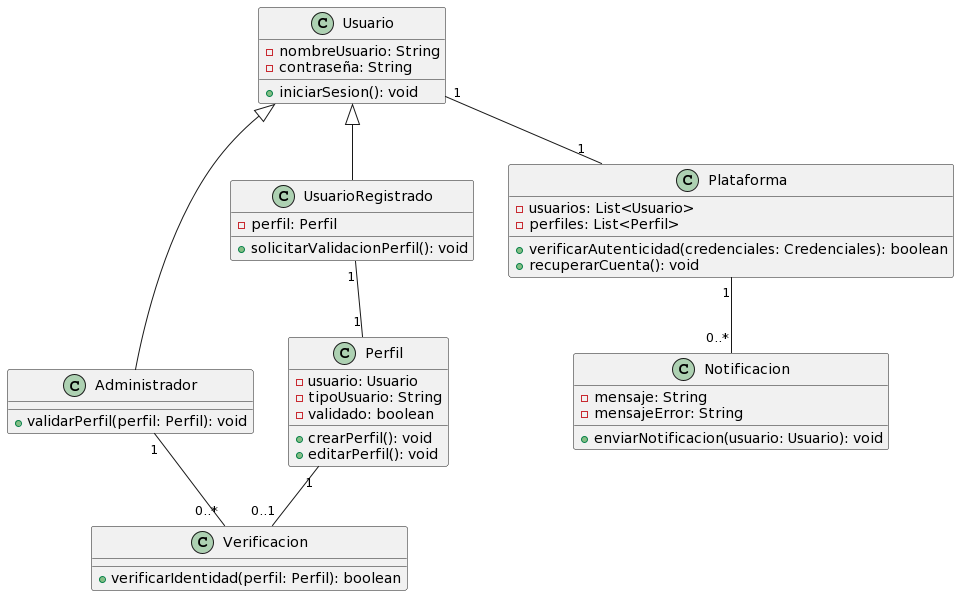
#### Diagrama de Colaboración

A diagram of a diagram

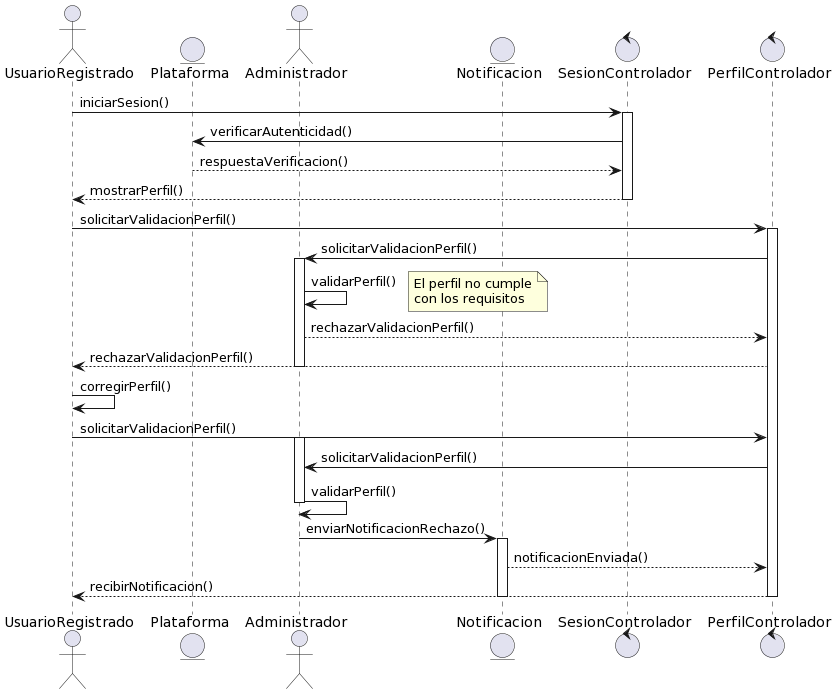
Description automatically generated

### **Escenario #2: Validar Perfil (Fallido)**

#### Diagrama de Clase



#### Diagrama de Secuencia



#### Diagrama de Colaboración

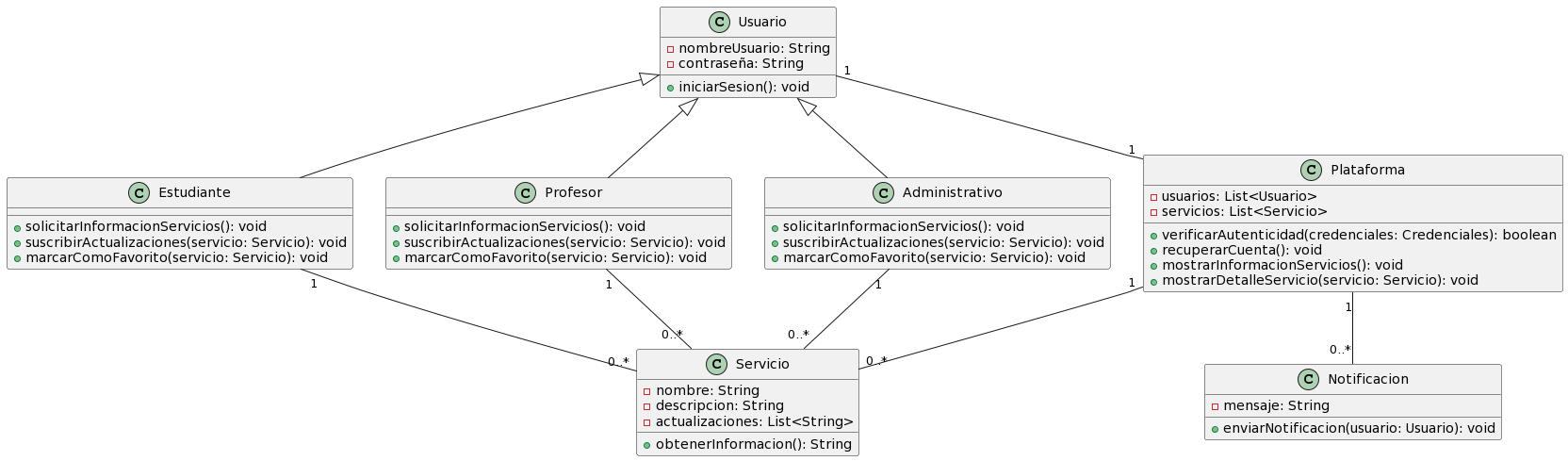
A diagram of a diagram

Description automatically generated

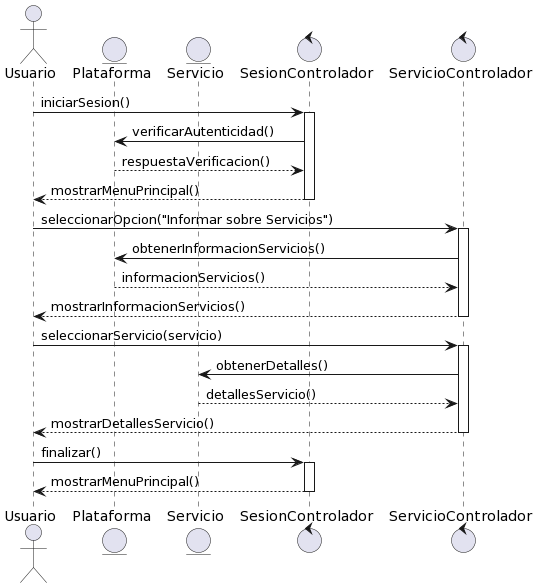
## **Caso de Uso: Informar sobre Servicios**

### **Escenario #1: Informar sobre Servicios (Exitoso)**

#### Diagrama de Clases



#### Diagrama de Secuencia



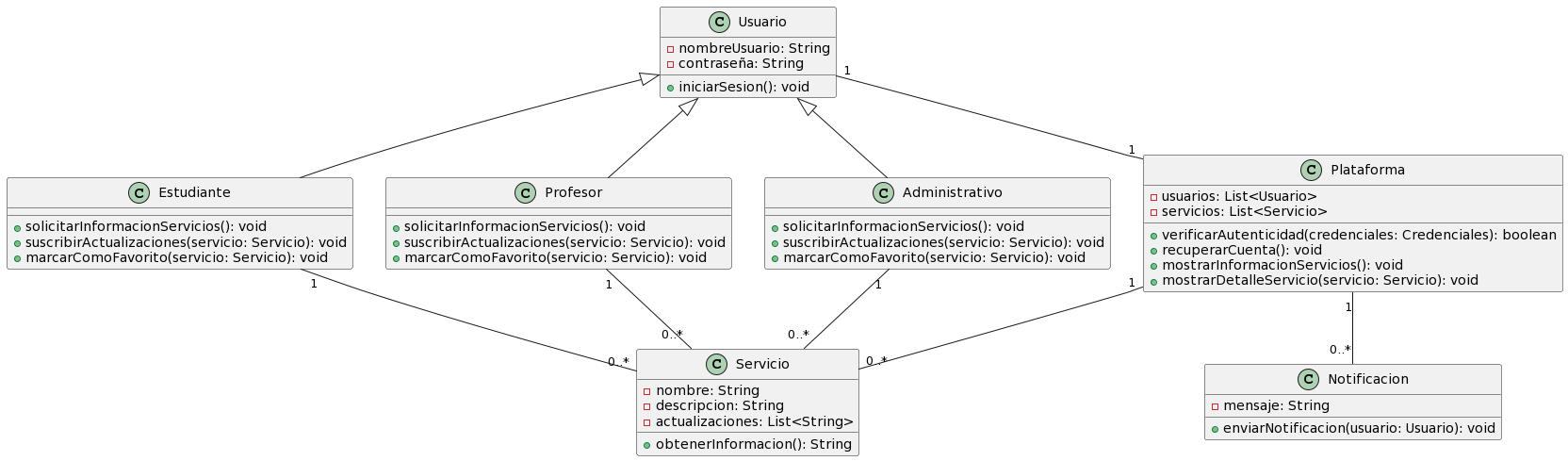
#### Diagrama de Colaboración

A diagram of a diagram

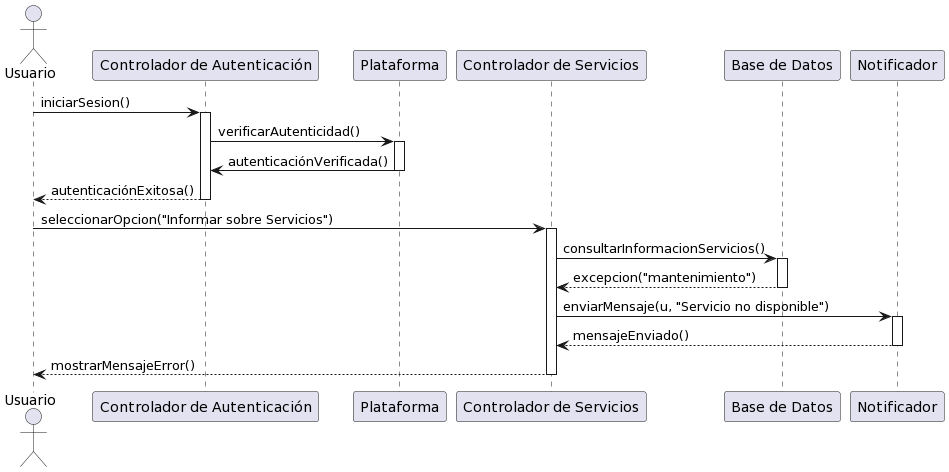
Description automatically generated

### **Escenario #2: Informar sobre Servicios (Fallido)**

#### Diagrama de Clases



#### Diagrama de Secuencia



#### Diagrama de Colaboración

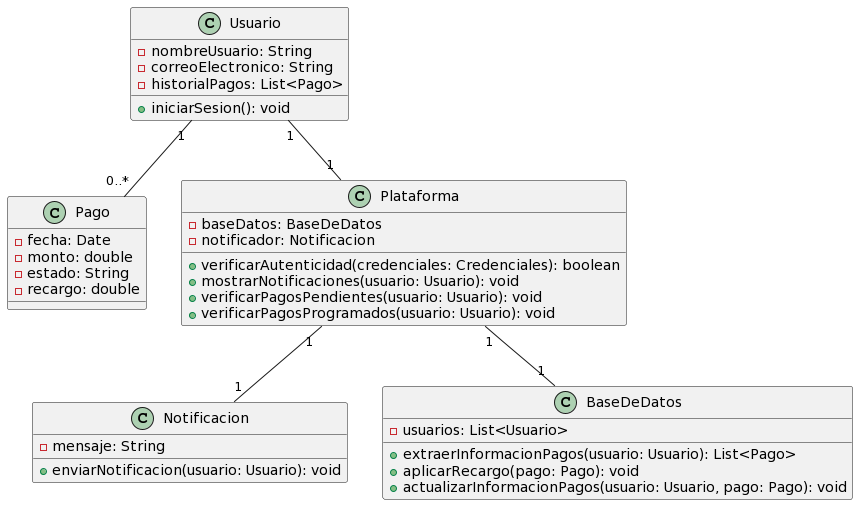
A diagram of a diagram

Description automatically generated

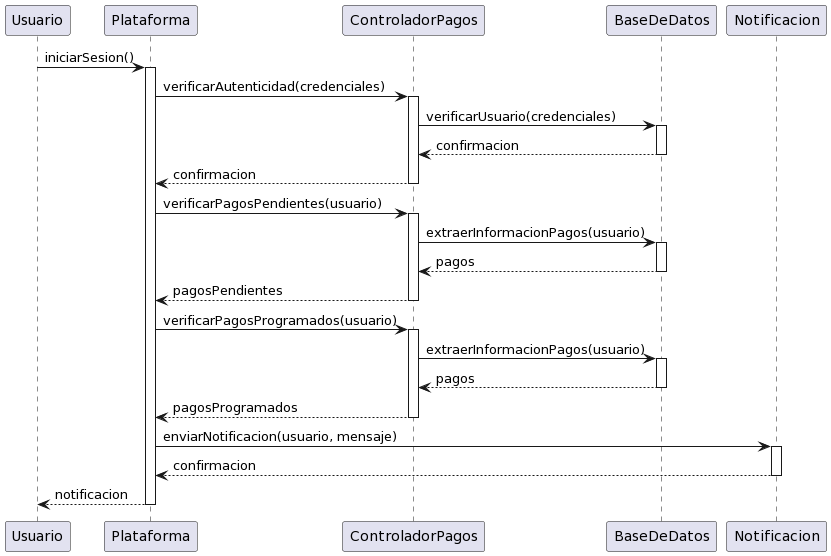
## **Caso de Uso: Notificación de Pagos Realizados y Pendientes**

### **Escenario #1: Notificación de Pagos Realizados y Pendientes (Exitoso)**

#### Diagramas de Clase



#### Diagrama de Secuencia



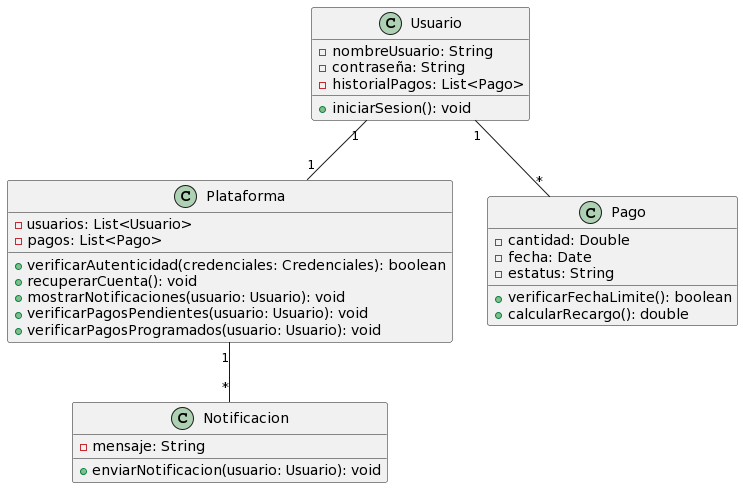
#### Diagrama de Colaboración

A diagram of a diagram

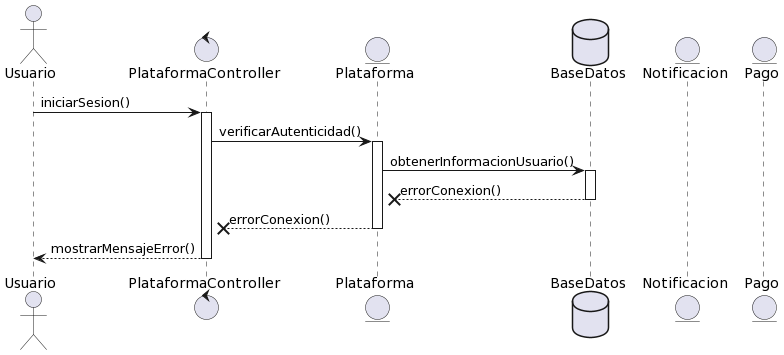
Description automatically generated

### **Escenario #1: Notificación de Pagos Realizados y Pendientes (Fallido)**

#### Diagramas de Clase



#### Diagrama de Secuencia

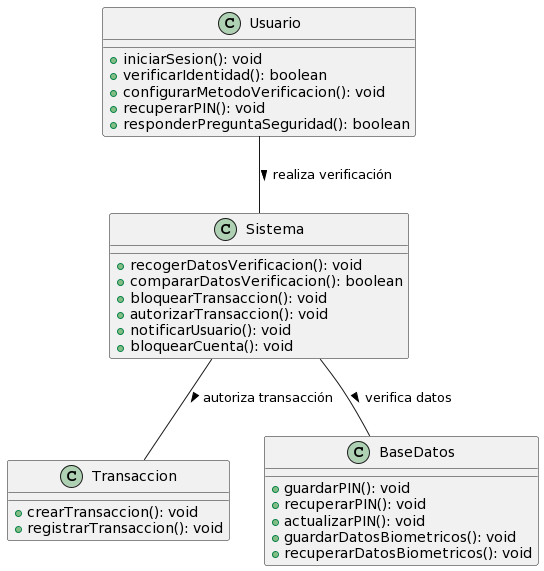


#### A diagram of a diagram Description automatically generatedDiagrama de Colaboración

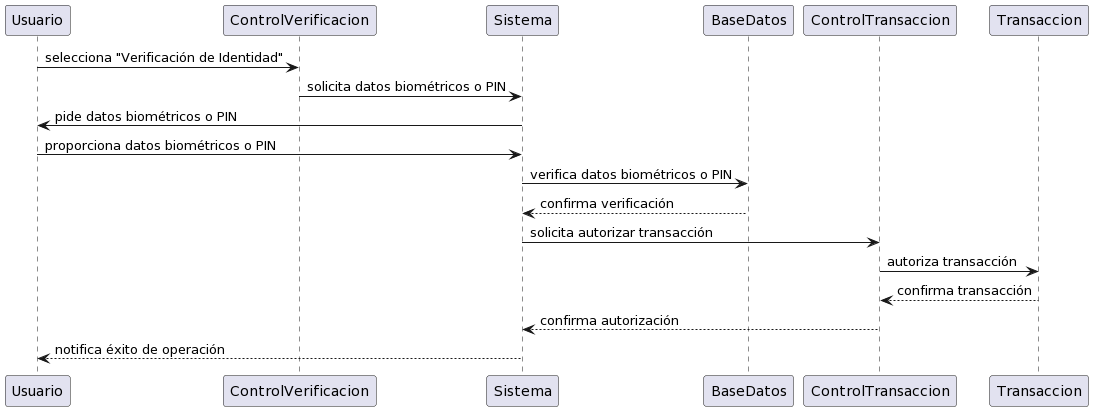
## **Caso de Uso: Verificación de Identidad**

### **Escenario #1: Verificación de Identidad (Exitoso)**

#### Diagramas de Clase



#### Diagrama de Secuencia



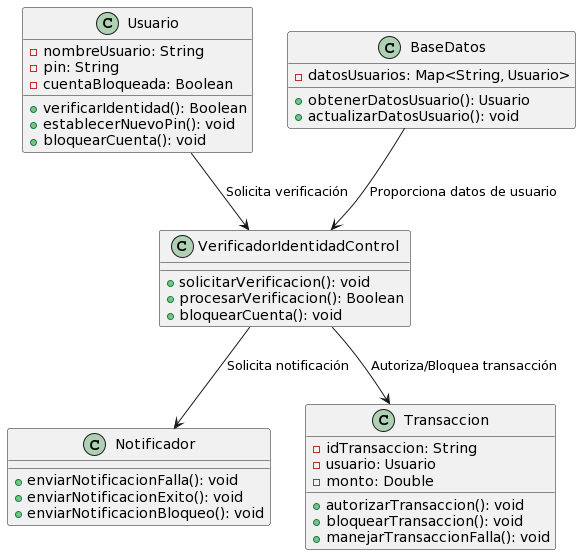
#### Diagrama de Colaboración

A diagram of a diagram

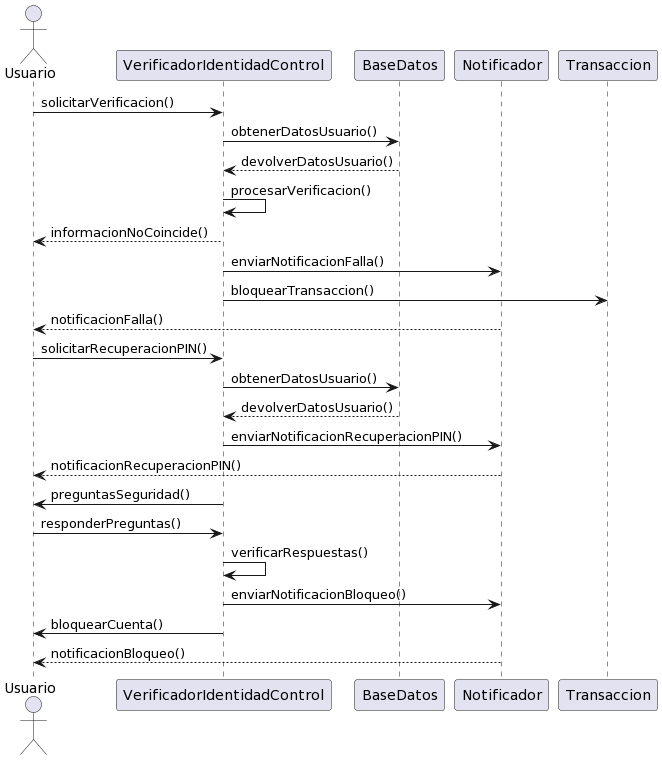
Description automatically generated

### **Escenario #2: Verificación de Identidad (Fallido)**

#### Diagramas de Clase



#### Diagrama de Secuencia



#### Diagrama de Colaboración

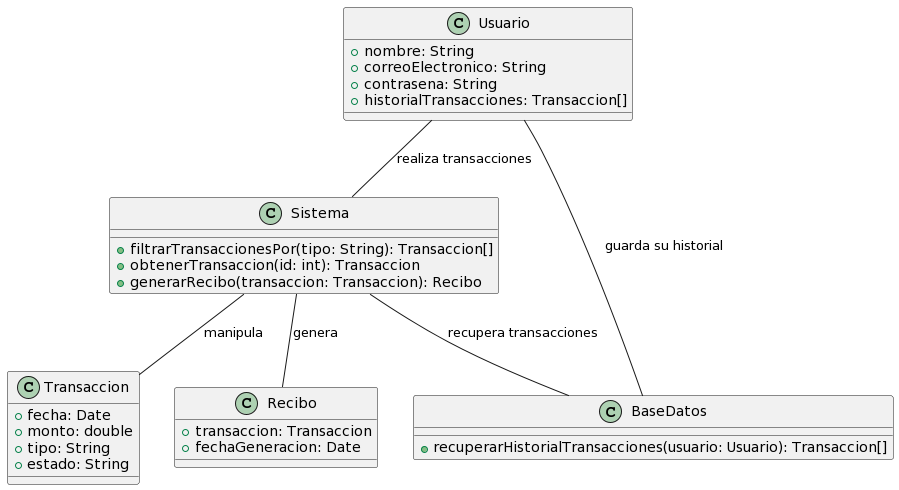
A diagram of a diagram

Description automatically generated

## **Caso de Uso: Mostrar Historial de Transacciones**

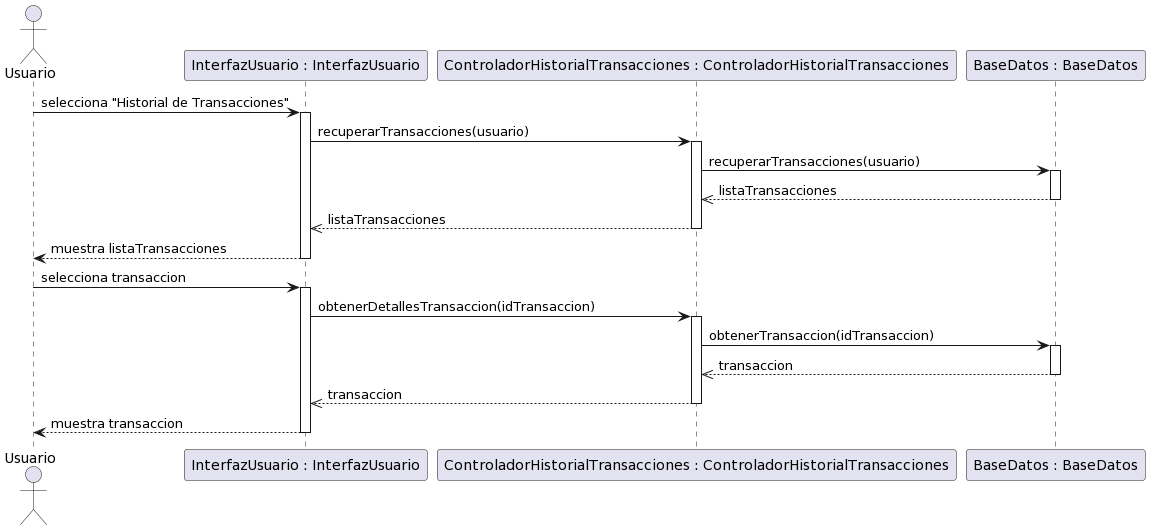
### **Escenario #1: Mostrar Historial de Transacciones (Exitoso)**

#### Diagramas de Clase



#### Diagrama de Secuencia

**Diagrama de Clases**



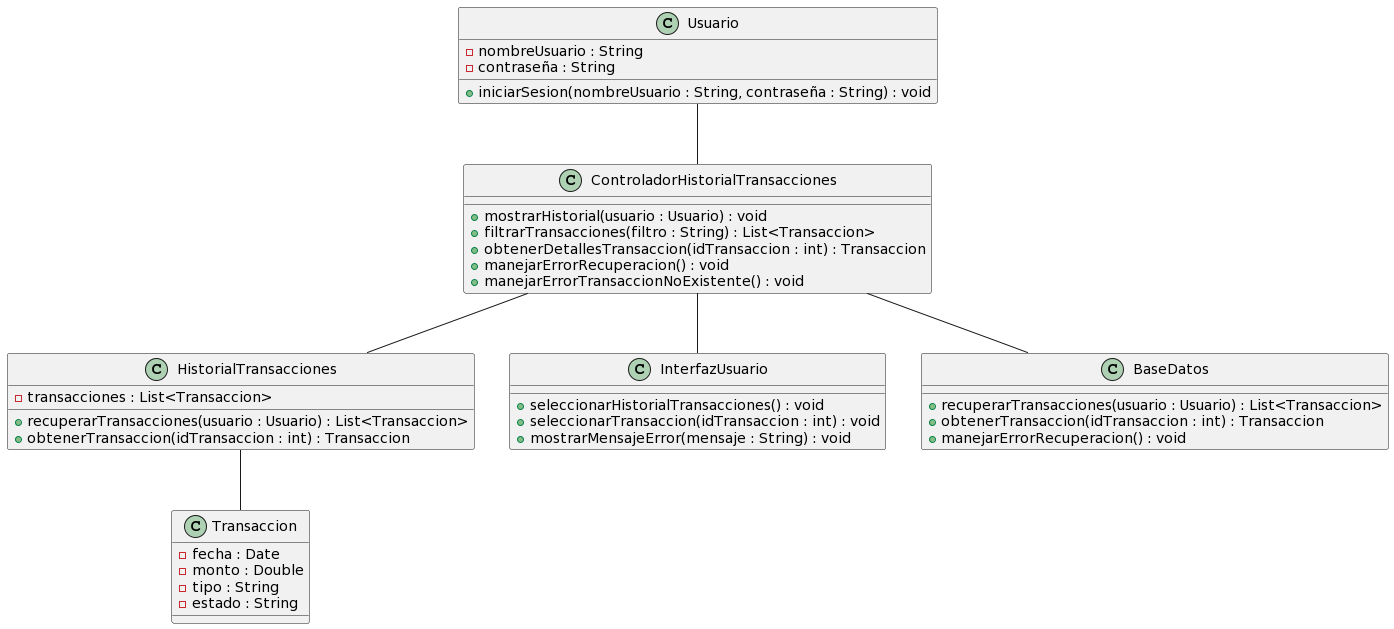
#### Diagrama de Colaboración

A diagram of a diagram

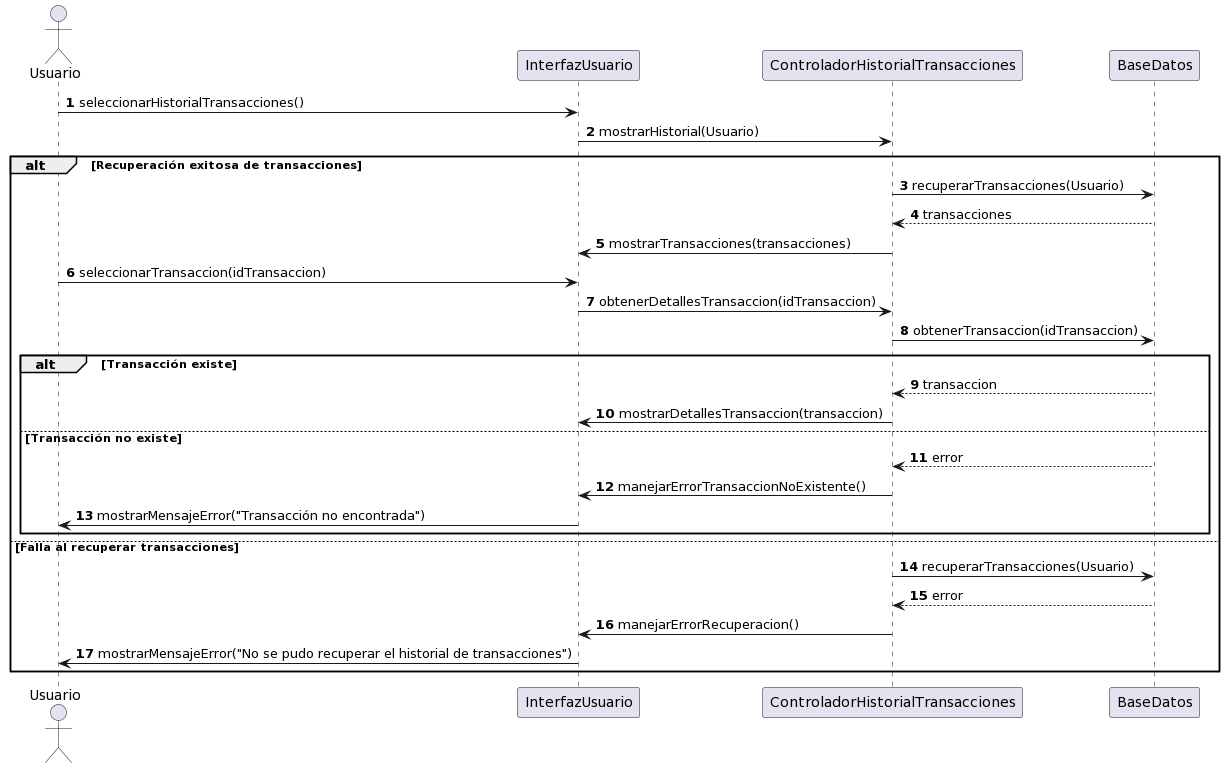
Description automatically generated

### **Escenario #2: Mostrar Historial de Transacciones (Fallido)**

#### Diagramas de Clase



#### Diagrama de Secuencia



#### Diagrama de Colaboración

A diagram of a transaction

Description automatically generated

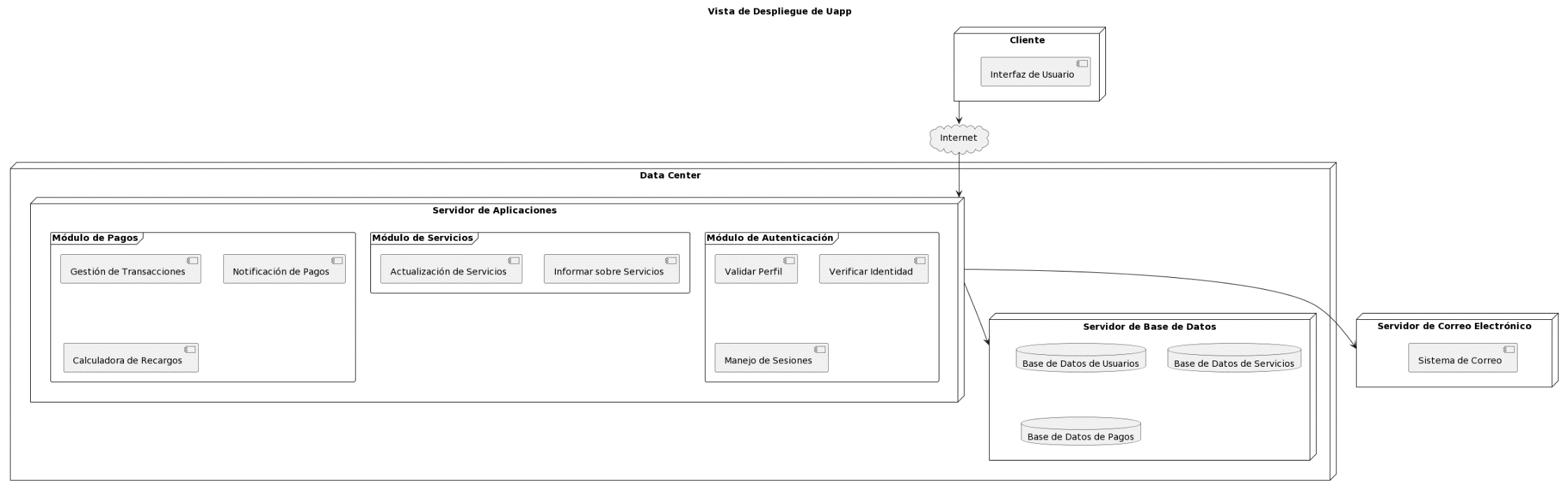
# **Vistas de la Aplicación**

## **Vista Lógica**

A diagram of a service

Description automatically generated

## **Vista de Despliegue**

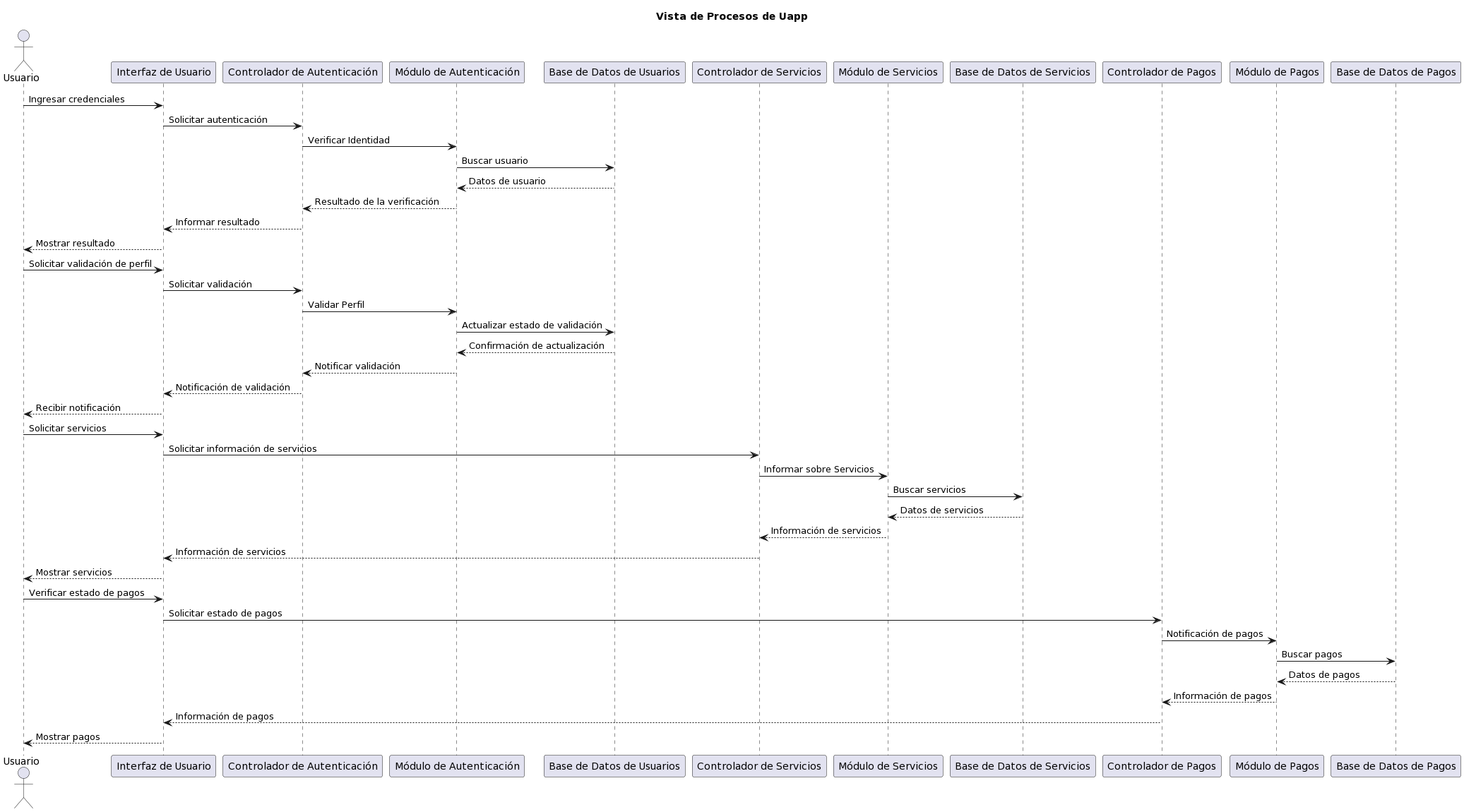
****

## **Vista Puesta en Marcha**

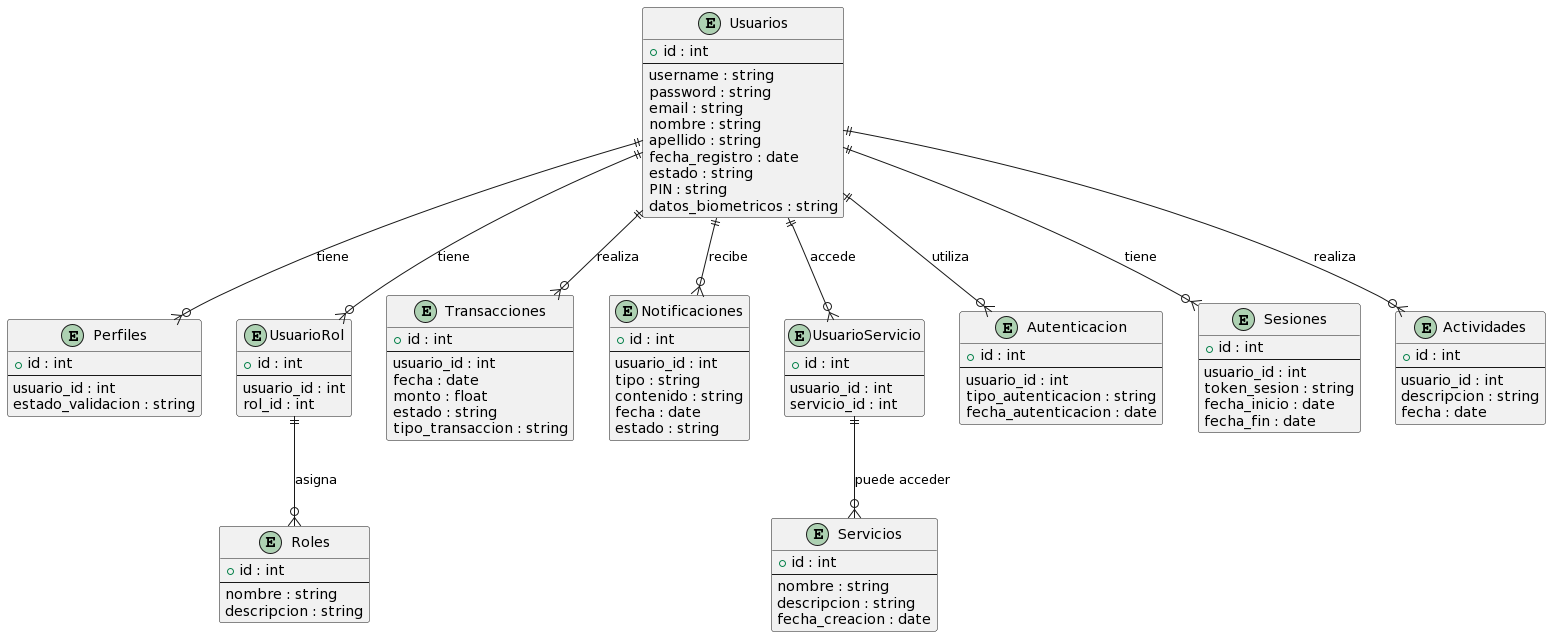
**A diagram of a diagram

Description automatically generated**

## **Vista de Procesos**

****

## **Vista de Datos**



# **Casos de Prueba**

Un Caso de Prueba en la Ingeniería de Software es un conjunto de condiciones o variables bajo las cuales un analista determinará si un sistema satisface los requisitos o funciona correctamente.

Es una especificación de las entradas, condiciones de ejecución, y los resultados esperados, que sirven para determinar si un sistema cumple con ciertos requisitos o no.

Los casos de prueba permiten al equipo de desarrollo de software validar y verificar si el software desarrollado cumple con los requisitos funcionales y no funcionales establecidos durante la fase de diseño.

Un Caso de Prueba bien construido no sólo examina la funcionalidad esperada, sino que también incluye condiciones límite, valores inválidos o erróneos, y otros escenarios basados en los requisitos del sistema.

Los casos de prueba se utilizan en el proceso de pruebas para identificar cualquier falla o defecto en el sistema y asegurar que se corrige antes de la entrega final del software.

Cada Caso de Prueba típicamente consta de una descripción de la condición o capacidad a probar, los pasos necesarios para llegar a la funcionalidad que se está probando, las entradas necesarias, y los resultados esperados.

Es común que los Casos de Prueba se agrupen en una suite de pruebas que abarque una funcionalidad completa o componente del sistema.

A continuación, tomaremos cada uno de los Casos de Uso y procederemos a diseñar Casos de Prueba para ellos.

## **Caso de Uso:** Validar Perfil

|  |  |  |  |
| --- | --- | --- | --- |
| **ID de Prueba** | **Descripción** | **Pasos** | **Resultados Esperados** |
| **T01** | Verificar el inicio de sesión y validación de perfil exitosos | 1. El usuario inicia sesión en la plataforma 2. El sistema verifica la autenticidad de las credenciales 3. El sistema muestra el perfil del usuario 4. El sistema revisa y valida el perfil del usuario | El sistema muestra una notificación de que el perfil ha sido validado |
| **T02** | Verificar la notificación de validación de perfil fallida | 1. El usuario inicia sesión en la plataforma 2. El sistema verifica la autenticidad de las credenciales 3. El sistema muestra el perfil del usuario 4. El sistema intenta validar el perfil del usuario | El sistema muestra una notificación de que la validación del perfil ha fallado |
| **T03** | Verificar la corrección del perfil y la revalidación | 1. El usuario inicia sesión en la plataforma 2. El sistema verifica la autenticidad de las credenciales 3. El sistema muestra el perfil del usuario 4. El sistema revisa y valida el perfil del usuario | El sistema muestra una notificación de que el perfil ha sido validado |
| **T04** | Verificar la funcionalidad "Recuperar Contraseña" durante el proceso de validación | 1. El usuario intenta iniciar sesión en la plataforma 2. El usuario selecciona la opción "Recuperar Contraseña" 3. El sistema verifica la identidad del usuario y restablece la contraseña | El sistema permite al usuario restablecer su contraseña y continuar con el proceso de validación |
| **T05** | Verificar la alternativa de verificación a través del correo electrónico durante el proceso de validación | 1. El usuario inicia sesión en la plataforma 2. El sistema verifica la autenticidad de las credenciales 3. El sistema muestra el perfil del usuario 4. El sistema ofrece la alternativa de verificación a través del correo electrónico | El sistema verifica la identidad del usuario a través del correo electrónico y continua con el proceso de validación |

## **Caso de Uso:** Informar sobre Servicios

|  |  |  |  |
| --- | --- | --- | --- |
| **ID de Prueba** | **Descripción** | **Pasos** | **Resultados Esperados** |
| **T06** | Verificar la autenticación del usuario | El usuario intenta iniciar sesión con sus credenciales correctas | El sistema permite al usuario acceder a la plataforma |
| **T07** | Verificar la obtención y presentación de la información de los servicios | 1. El usuario navega hasta la opción “Informar sobre Servicios” 2. El sistema realiza una consulta a la base de datos | El sistema presenta una lista estructurada con la información de los servicios al usuario |
| **T08** | Verificar el filtro y búsqueda de servicios | El usuario intenta filtrar los servicios por una categoría específica | El sistema muestra solo los servicios que corresponden a la categoría seleccionada |
| **T09** | Verificar la presentación detallada de un servicio | El sistema realiza una nueva consulta a la base de datos para obtener los detalles del servicio | El sistema muestra una descripción detallada del servicio al usuario |
| **T10** | Verificar la funcionalidad de "Suscribirse a las actualizaciones del servicio" | El usuario selecciona la opción "Suscribirse a las actualizaciones del servicio" | El sistema añade el servicio a la lista de suscripciones del usuario y le notifica que recibirá futuras actualizaciones sobre este servicio |

## **Caso de Uso:** Notificación de Pagos Realizados y Pendientes

|  |  |  |  |
| --- | --- | --- | --- |
| **ID de Prueba** | **Descripción** | **Pasos** | **Resultados Esperados** |
| **T11** | Verificar la autenticación del usuario | El usuario intenta iniciar sesión con sus credenciales correctas | El sistema permite al usuario acceder a la plataforma |
| **T12** | Verificar la recopilación de información de pagos | El sistema accede al historial de pagos del usuario | El sistema recopila correctamente la información de pagos del usuario |
| **T13** | Verificar la generación y envío de notificaciones | 1. El sistema genera una notificación con la información de los pagos realizados y pendientes 2. El sistema envía la notificación al usuario por correo electrónico | El sistema envía con éxito la notificación al usuario |
| **T14** | Verificar la aplicación de recargos | 1. El sistema verifica la fecha actual y la fecha límite del pago 2. El sistema aplica un recargo del 25% al monto pendiente | El sistema aplica correctamente el recargo y actualiza la notificación con esta información |
| **T15** | Verificar el funcionamiento de los pagos programados | 1. El sistema verifica el estado del pago programado 2. El sistema actualiza la información de los pagos y envía una notificación al usuario | El sistema maneja correctamente los pagos programados y notifica al usuario sobre el estado del pago |

## **Caso de Uso:** Verificación de Identidad

|  |  |  |  |
| --- | --- | --- | --- |
| **ID de Prueba** | **Descripción** | **Pasos** | **Resultados Esperados** |
| **T16** | Verificar la autenticación del usuario | 1. El usuario selecciona la opción "Verificación de Identidad" 2. El usuario proporciona datos biométricos o introduce su PIN | El sistema autoriza la transacción y notifica al usuario |
| **T17** | Verificar el manejo de PIN incorrectos | 1. El usuario selecciona la opción "Verificación de Identidad" 2. El usuario introduce un PIN incorrecto | El sistema bloquea la transacción y envía una notificación al usuario |
| **T18** | Verificar la recuperación del PIN | 1. El usuario selecciona la opción "Verificación de Identidad" 2. El usuario selecciona "Recuperar PIN" 3. El usuario responde las preguntas de seguridad correctamente | El sistema permite al usuario restablecer su PIN |
| **T19** | Verificar el manejo de respuestas incorrectas a las preguntas de seguridad | 1. El usuario selecciona la opción "Verificación de Identidad" 2. El usuario selecciona "Recuperar PIN" 3. El usuario responde incorrectamente a las preguntas de seguridad | El sistema no permite al usuario restablecer su PIN |
| **T20** | Verificar la autenticación de dos factores | 1. El usuario selecciona la opción "Verificación de Identidad" 2. El usuario introduce su PIN y responde correctamente a una pregunta de seguridad | El sistema autoriza la transacción y notifica al usuario |

## **Caso de Uso:** Mostrar Historial de Transacciones del Usuario

|  |  |  |  |
| --- | --- | --- | --- |
| **ID de Prueba** | **Descripción** | **Pasos** | **Resultados Esperados** |
| **T21** | Verificar la visualización del historial de transacciones | El usuario selecciona la opción "Historial de Transacciones" | El sistema muestra todas las transacciones realizadas por el usuario |
| **T22** | Verificar el filtrado de transacciones | 1. El usuario selecciona la opción "Historial de Transacciones" 2. El usuario aplica un filtro de búsqueda | El sistema muestra las transacciones que coinciden con el filtro |
| **T23** | Verificar la visualización de detalles de una transacción específica | 1. El usuario selecciona la opción "Historial de Transacciones" 2. El usuario selecciona una transacción específica | El sistema muestra los detalles de la transacción seleccionada |
| **T24** | Verificar la descarga del recibo de una transacción exitosa | 1. El usuario selecciona la opción "Historial de Transacciones" 2. El usuario selecciona una transacción exitosa 3. El usuario selecciona la opción para descargar el recibo | El sistema genera y descarga el recibo de la transacción |
| **T25** | Verificar la manipulación de transacciones pendientes o fallidas | 1. El usuario selecciona la opción "Historial de Transacciones" 2. El usuario selecciona una transacción pendiente o fallida 3. El usuario toma una acción para resolver la transacción | El sistema actualiza la transacción según la acción tomada por el usuario |

**UNIVERSIDAD TECNOLÓGICA DE PANAMA**

**FACULTAD DE INGENIERIA DE SISTEMAS COMPUTACIONALES**

**COEVALUACIÓN**

**Nombre:** Johel Batista

**Escala numérica: Muy Significativo: 5 Significativo: 4 Aceptable: 3 Inexistente: 2**

**Instrucciones**: Conforme a la participación en el trabajo en grupo, marque con una (X) la casilla que considere le corresponde. Valor 45 puntos.

Equivale a 5% del trabajo escrito, para que tenga validez, esta evaluación debe ser avalada por todos los integrantes del grupo, con su firma.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ASPECTO** | **Muy**  **Significativo** | **Significativo** | **Aceptable** | **Inexistente** |
| Su aporte al equipo fue | **X** |  |  |  |
| Su respuesta a las responsabilidades asignadas y aceptadas en el grupo fue | **X** |  |  |  |
| Su aporte a la creación de un clima de tolerancia y respeto ante la presencia de pensamientos u opiniones diferentes fue | **X** |  |  |  |
| Si el participante no hubiera actuada en equipo, su ausencia se habría considerado como | **X** |  |  |  |
| Su contribución a la solución del problema en el grupo fue | **X** |  |  |  |
| Los planteamientos personales (originales) al desarrollo del trabajo fueron | **X** |  |  |  |
| Las iniciativas asumidas por su persona fueron | **X** |  |  |  |
| Su asistencia a las reuniones de trabajo fue | **X** |  |  |  |
| Su contribución a la generación de un clima de armonía fue | **X** |  |  |  |



**UNIVERSIDAD TECNOLÓGICA DE PANAMA**

**FACULTAD DE INGENIERIA DE SISTEMAS COMPUTACIONALES**

**COEVALUACIÓN**

**Nombre:** Sebastián Maldonado

**Escala numérica: Muy Significativo: 5 Significativo: 4 Aceptable: 3 Inexistente: 2**

**Instrucciones**: Conforme a la participación en el trabajo en grupo, marque con una (X) la casilla que considere le corresponde. Valor 45 puntos.

Equivale a 5% del trabajo escrito, para que tenga validez, esta evaluación debe ser avalada por todos los integrantes del grupo, con su firma.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ASPECTO** | **Muy**  **Significativo** | **Significativo** | **Aceptable** | **Inexistente** |
| Su aporte al equipo fue | **X** |  |  |  |
| Su respuesta a las responsabilidades asignadas y aceptadas en el grupo fue | **X** |  |  |  |
| Su aporte a la creación de un clima de tolerancia y respeto ante la presencia de pensamientos u opiniones diferentes fue | **X** |  |  |  |
| Si el participante no hubiera actuada en equipo, su ausencia se habría considerado como | **X** |  |  |  |
| Su contribución a la solución del problema en el grupo fue | **X** |  |  |  |
| Los planteamientos personales (originales) al desarrollo del trabajo fueron | **X** |  |  |  |
| Las iniciativas asumidas por su persona fueron | **X** |  |  |  |
| Su asistencia a las reuniones de trabajo fue | **X** |  |  |  |
| Su contribución a la generación de un clima de armonía fue | **X** |  |  |  |



**UNIVERSIDAD TECNOLÓGICA DE PANAMA**

**FACULTAD DE INGENIERIA DE SISTEMAS COMPUTACIONALES**

**COEVALUACIÓN**

**Nombre:** Alessandro Vergara

**Escala numérica: Muy Significativo: 5 Significativo: 4 Aceptable: 3 Inexistente: 2**

**Instrucciones**: Conforme a la participación en el trabajo en grupo, marque con una (X) la casilla que considere le corresponde. Valor 45 puntos.

Equivale a 5% del trabajo escrito, para que tenga validez, esta evaluación debe ser avalada por todos los integrantes del grupo, con su firma.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ASPECTO** | **Muy**  **Significativo** | **Significativo** | **Aceptable** | **Inexistente** |
| Su aporte al equipo fue | **X** |  |  |  |
| Su respuesta a las responsabilidades asignadas y aceptadas en el grupo fue | **X** |  |  |  |
| Su aporte a la creación de un clima de tolerancia y respeto ante la presencia de pensamientos u opiniones diferentes fue | **X** |  |  |  |
| Si el participante no hubiera actuada en equipo, su ausencia se habría considerado como | **X** |  |  |  |
| Su contribución a la solución del problema en el grupo fue | **X** |  |  |  |
| Los planteamientos personales (originales) al desarrollo del trabajo fueron | **X** |  |  |  |
| Las iniciativas asumidas por su persona fueron | **X** |  |  |  |
| Su asistencia a las reuniones de trabajo fue | **X** |  |  |  |
| Su contribución a la generación de un clima de armonía fue | **X** |  |  |  |



**UNIVERSIDAD TECNOLÓGICA DE PANAMA**

**FACULTAD DE INGENIERIA DE SISTEMAS COMPUTACIONALES**

**COEVALUACIÓN**

**Nombre:** Mariana Zabala

**Escala numérica: Muy Significativo: 5 Significativo: 4 Aceptable: 3 Inexistente: 2**

**Instrucciones**: Conforme a la participación en el trabajo en grupo, marque con una (X) la casilla que considere le corresponde. Valor 45 puntos.

Equivale a 5% del trabajo escrito, para que tenga validez, esta evaluación debe ser avalada por todos los integrantes del grupo, con su firma.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ASPECTO** | **Muy**  **Significativo** | **Significativo** | **Aceptable** | **Inexistente** |
| Su aporte al equipo fue | **X** |  |  |  |
| Su respuesta a las responsabilidades asignadas y aceptadas en el grupo fue | **X** |  |  |  |
| Su aporte a la creación de un clima de tolerancia y respeto ante la presencia de pensamientos u opiniones diferentes fue | **X** |  |  |  |
| Si el participante no hubiera actuada en equipo, su ausencia se habría considerado como | **X** |  |  |  |
| Su contribución a la solución del problema en el grupo fue | **X** |  |  |  |
| Los planteamientos personales (originales) al desarrollo del trabajo fueron | **X** |  |  |  |
| Las iniciativas asumidas por su persona fueron | **X** |  |  |  |
| Su asistencia a las reuniones de trabajo fue | **X** |  |  |  |
| Su contribución a la generación de un clima de armonía fue | **X** |  |  |  |



# **Lecciones Aprendidas**

En el transcurso de este curso, transformamos la improvisada creación de software en un proceso estandarizado que se centra en las necesidades del usuario y proporciona información esencial para el equipo de desarrollo. Esto nos ha permitido establecer criterios objetivos para evaluar la calidad del software y mejorar su proceso de creación, lo que resulta en un software con menos errores y, en última instancia, reduce los costos asociados con los errores una vez que el software está terminado.

* Johel Batista

Mi conclusión es que aplicar una arquitectura de software en un proyecto real nos ayuda a establecer una base sólida para su diseño. Las especificaciones de casos de uso reflejan detalladamente las necesidades de los clientes o usuarios del sistema. Los casos de prueba son un conjunto de pruebas esenciales para validar y asegurar el correcto funcionamiento del sistema. Los diagramas de secuencia y colaboración son herramientas útiles de modelado en UML para cada flujo de ejecución de sus correspondientes escenarios.

* Alessandro Vergara

El aprendizaje más valioso de este proyecto final fue la capacidad de reforzar y ampliar mis conocimientos previos de ingeniería de software. A lo largo del semestre, comprendí de mejor manera cómo crear escenarios para los casos de uso que creamos, entendiendo su utilidad de una manera más clara. También profundicé en los patrones que podemos usar para representar nuestro software y la importancia de cada uno. Además, aprendí sobre los diagramas de actividad y la importancia y utilidad de las vistas 4+1.

* Sebastián Maldonado

Para finalizar, la arquitectura de software permite la estructuración, construcción y visualización de los detalles requeridos para desarrollar un sistema de software en base a una necesidad específica. Nos ayuda tanto a los desarrolladores como a los usuarios sin conocimiento técnico a entender el funcionamiento del software. A través de la arquitectura de software, podemos visualizar cómo las necesidades del usuario son satisfechas y, si se cumplen correctamente, podemos entregar un producto final de alta calidad.

* Mariana Zabala

En general, la arquitectura de software es una parte esencial del desarrollo de software, estableciendo las reglas, características y políticas que se deben aplicar para cumplir con las necesidades del cliente. Este trabajo nos ha permitido aplicar todo lo aprendido en el curso de ingeniería de software, desde los requerimientos y casos de uso, hasta los escenarios, diagramas de actividad, vistas y casos de prueba.

# **Referencias Bibliográficas**

1. Bello, E. (2021). Stakeholders: quiénes son, por qué son importantes y cómo gestionarlos. *Thinking for Innovation*. https://www.iebschool.com/blog/stakeholders-quienes-son-digital-business/
2. *Cómo escribir casos de prueba: una plantilla de muestra con ejemplos*. (2020, enero 1). Ebooks Online. https://ebooksonline.es/como-escribir-casos-de-prueba-una-plantilla-de-muestra-con-ejemplos/
3. *Diseño - Modelo 4+1 - documento 1 (software architecture document) — read the docs template 1.0 documentation*. (s/f). Readthedocs.Io. Recuperado el 15 de julio de 2022, de https://proyecto-semestral.readthedocs.io/en/latest/6%20-%20Design.html
4. *Planos Arquitectónicos: El Modelo de 4+1 Vistas de la Arquitectura del Software*. (s/f). Docplayer.Es. Recuperado el 15 de julio de 2022, de https://docplayer.es/3857190-Planos-arquitectonicos-el-modelo-de-4-1-vistas-de-la-arquitectura-del-software.html
5. Turrado, J., & Turrado, J. (s/f). *Introducción rápida a LINQ con C#: manejar información en memoria nunca fue tan sencillo*. campusMVP.es. Recuperado el 15 de julio de2022, de https://www.campusmvp.es/recursos/post/introduccion-rapida-a-linq-con-c-sharp.aspx
6. (S/f). Edu.uy. Recuperado el 15 de julio de 2022, de https://www.fing.edu.uy/inco/grupos/mina/pGrado/simulador/docs/SAD.pdf

# **Anexos**

## **Glosario de Términos**

1. **RUP (Rational Unified Process):** Proceso de desarrollo de software iterativo e incremental creado por Rational Software Corporation (ahora parte de IBM). Proporciona un marco de trabajo para el desarrollo de software basado en mejores prácticas.
2. **Requisitos**: Especificaciones de lo que el sistema debe hacer o lograr. Los requisitos definen las funcionalidades, características y restricciones del software.
3. **Caso de Uso**: Descripción de una interacción entre el sistema y los actores externos. Representa un escenario que describe cómo el sistema se utiliza para lograr un objetivo específico.
4. **Arquitectura**: Estructura y organización fundamental del sistema. Define los componentes principales, las relaciones entre ellos y los principios de diseño que guían el desarrollo del software.
5. **Diagrama de Clases**: Representación visual de las clases, atributos y relaciones entre clases en un sistema. Muestra la estructura estática del software.
6. **Diagrama de Secuencia**: Representación visual de la interacción entre objetos a lo largo del tiempo. Muestra cómo los objetos se comunican y colaboran para lograr un resultado específico.
7. **Caso de Prueba**: Documento que describe una secuencia de acciones y condiciones que se utilizarán para verificar si una función específica del software funciona correctamente.
8. **Prototipo**: Versión preliminar y parcial del sistema o de una función específica. Se utiliza para validar y refinar los requisitos, el diseño y la funcionalidad antes de desarrollar la versión final.
9. **Administración de Configuración**: Proceso de gestión y control de los elementos de configuración del software, incluidos el código fuente, la documentación y los recursos asociados. Se asegura de que los cambios se realicen de manera controlada y rastreable.