**SGCalerías**

**(SSAD) SGCalerías Software Architecture Document**

**Versión 1.0**

**INTEGRANTES:**Victor Bustos   
Victor Otero **Identificación de Documento**

| **Identificación** | SGCalerías Software Architecture Document |
| --- | --- |
| **Proyecto** | Desarrollo sistema de gestión de comunidades SGCalerías |
| **Versión** | 1.0 |

| **Documento mantenido por** | Victor Otero y Victor Bustos |
| --- | --- |
| **Fecha de última revisión** | 14/09/2024 |
| **Fecha de próxima revisión** | Pendiente |

| **Documento aprobado por** | Víctor Otero |
| --- | --- |
| **Fecha de última aprobación** | 20/09/2024 |

**Historia de Revisiones**

| **Fecha** | **Versión** | **Descripción** | **Autor** |
| --- | --- | --- | --- |
| 14/09/2024 | 1.0 | Creación de documento | Equipo de desarrollo |
| 18/09/2024 | 1.1 | Modificación de documento | Equipo de desarrollo |

**Tabla de Contenidos**

[1](#_heading=h.gjdgxs) Introducción 3

1.1 Contexto del Problema 4

1.2 Propósito 4

1.3 Ámbito 4

1.4 Referencias 4

1.5 Resumen ejecutivo 4

1.6 Representación 4

2Metas y Restricciones de la Arquitectura 5

2.1 Metas de la arquitectura [5](#_heading=h.4d34og8)

2.2 Restricciones de la Arquitectura [5](#_heading=h.4d34og8)

2.3 Otros antecedentes y consideraciones [5](#_heading=h.4d34og8)

3Vista de Casos de Uso y Escenarios de Calidad 6

3[.1 Modelo de Casos de Uso](#_heading=h.35nkun2) 6

3[.2 Especificación de Casos de Uso Relevantes 15](#_heading=h.1ksv4uv)

3[.3 Especificación de los Escenarios de Calidad Relevantes](#_heading=h.44sinio)17

3[.4 Matriz de riesgos](#_heading=h.z337ya) 28

3[.5](#_heading=h.3j2qqm3) [Parte Estructural](#_heading=h.z337ya) 32

3.6 [Parte dinámica](#_heading=h.3j2qqm3) 33

4Vista de Procesos 34

5Vista de Implementación 35

5.1 vista de despliegue 35

5.2 vista física35

6Decisiones de diseño y selección de alternativas 36

6.1 Reutilización36

7Patrón de arquitectura 36

1. **Introducción**
   1. **Contexto del Problema**

El cliente necesita que nuestro equipo de desarrolladores diseñe e implemente un sistema que permita gestionar de manera integral los aspectos administrativos del condominio Calerias. El sistema debe garantizar la organización eficiente de información clave, como la de residentes, gastos comunes, convenios de pago, compras y ventas, asegurando además la transparencia y la accesibilidad de los datos.

* 1. **Propósito**

El propósito del proyecto es entregar al equipo de desarrollo una guía para crear un sistema robusto que permita al administrador del condominio gestionar los procesos administrativos y financieros de forma eficiente y organizada, mejorando así la experiencia tanto para los residentes como para los gestores del condominio.

* 1. **Ámbito**

El sistema abarca la gestión integral de condominios, incluyendo el registro y administración de residentes, gastos comunes, convenios de pago, ventas de productos del condominio y la generación de reportes financieros. Además, garantizará la automatización de procesos clave, como el envío de recibos, el pago de cuotas y la generación de estados de cuenta.

* 1. **Referencias**
* Diagrama de casos de uso
* Diagrama de clases
* Diagrama de entidad relación
* Escenarios de calidad
* Matriz de riesgos
  1. **Resumen ejecutivo**

**Resumen del proyecto problemática, solución, etc 5 ó 6 líneas.**

El condominio Calerias enfrenta desafíos administrativos significativos debido a la falta de un sistema de gestión eficiente. Este proyecto busca implementar una solución tecnológica que permita al administrador gestionar procesos clave como el manejo de gastos comunes, pagos de residentes y ventas de productos del condominio. Con esta herramienta, se busca optimizar las operaciones diarias, mejorar la transparencia con los residentes y facilitar una administración organizada y efectiva.

* 1. **Representación**

Se utilizará el enfoque de vistas 4+1, con la entrega de los siguientes elementos en esta etapa:

* **Vista lógica**:
  + Diagrama de clases
  + Diagrama de entidad-relación
* **Vista de comportamiento**:
  + Diagrama de casos de uso

Adicionalmente, se entregarán:

* Escenarios de calidad (1 por cada atributo y métrica de calidad seleccionada).
* Matriz de riesgos del proyecto.

1. **Metas y Restricciones de la Arquitectura**
   1. **Metas de la arquitectura**

El sistema cumplirá con las siguientes metas:

* **Seguridad**: Protegerá la información confidencial de los usuarios mediante contraseñas encriptadas y protocolos de seguridad actualizados.
* **Disponibilidad**: El sistema estará disponible 24/7 con alta tolerancia a fallos.
* **Usabilidad**: Será intuitivo y fácil de usar tanto para administradores como para residentes.
* **Portabilidad**: Compatible con diversos dispositivos y navegadores.
* **Eficiencia**: Tiempo de respuesta menor a 4 segundos en operaciones complejas.
* **Integridad de datos**: Respaldará y gestionará datos obsoletos para prevenir fallos sistémicos.
  1. **Restricciones de la Arquitectura**

**Tiempo**: No definido.

**Dinero**: No definido.

**Infraestructura**: Desarrollo web con HTML, Django y base de datos SQL Server.

**Conocimiento Previo**: Experiencia en desarrollo de aplicaciones web con énfasis en ventas y gestión de información.

* 1. **Otros antecedentes y consideraciones**

Otros aspectos a considerar como lenguajes, frameworks, base de datos, etc..

**Lenguajes:** HTML, CSS, JavaScript (con JQuery).

**Framework:** Django.

**Base de Datos:** SQL Server.

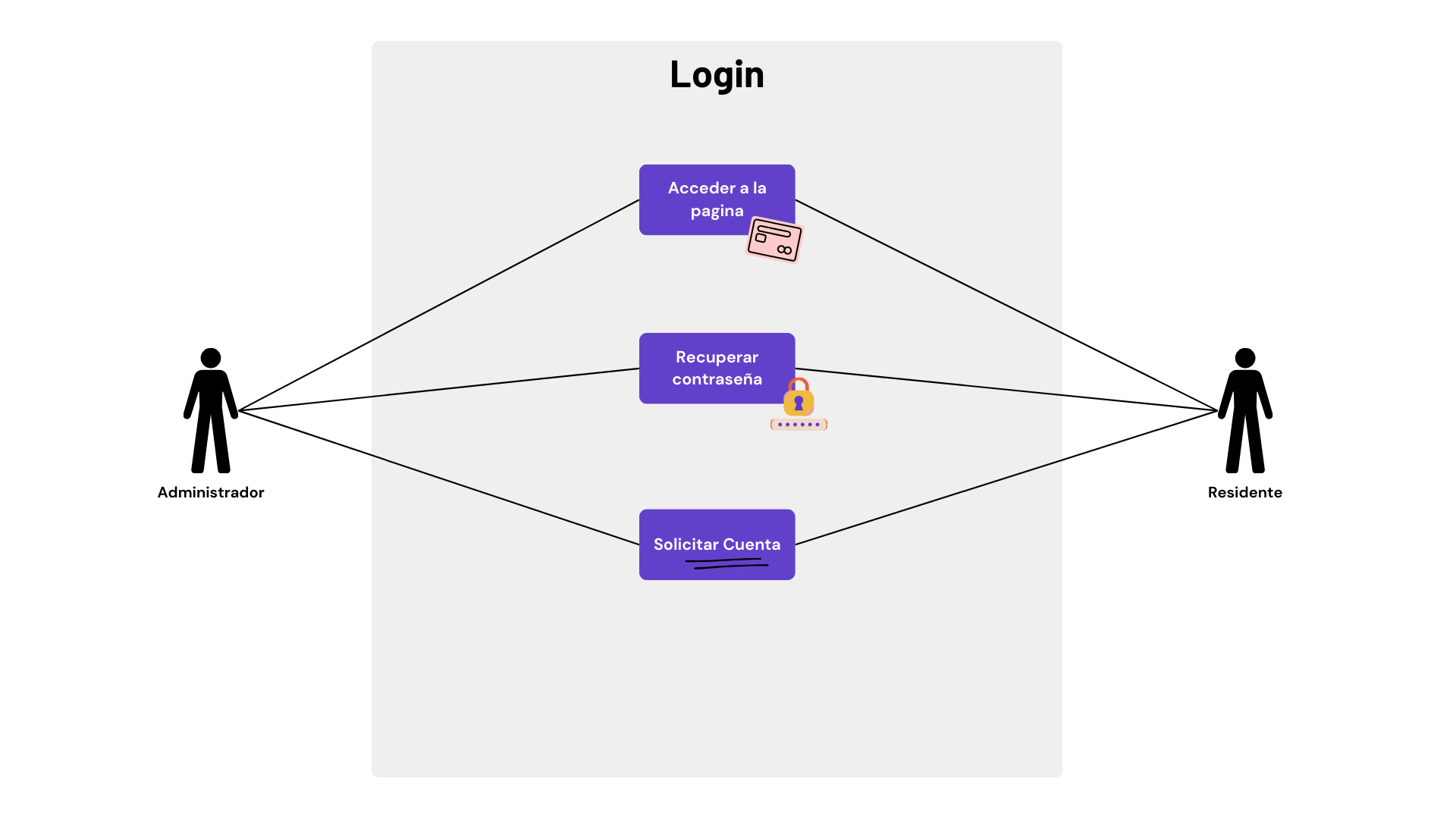
**Repositorio:** GitHub Desktop.

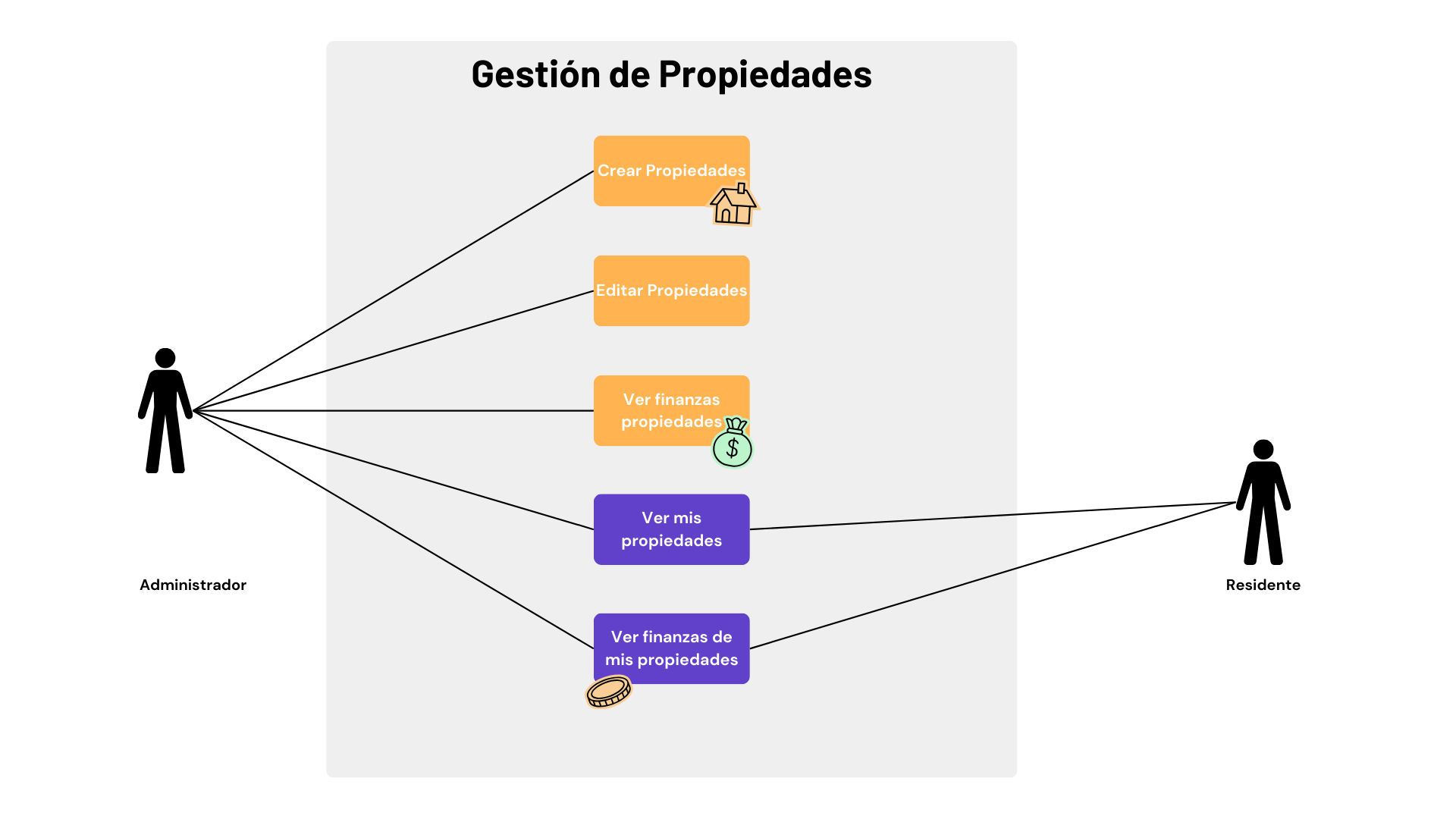
1. **Vista de Escenario (Casos de Uso) y Escenarios de Calidad**

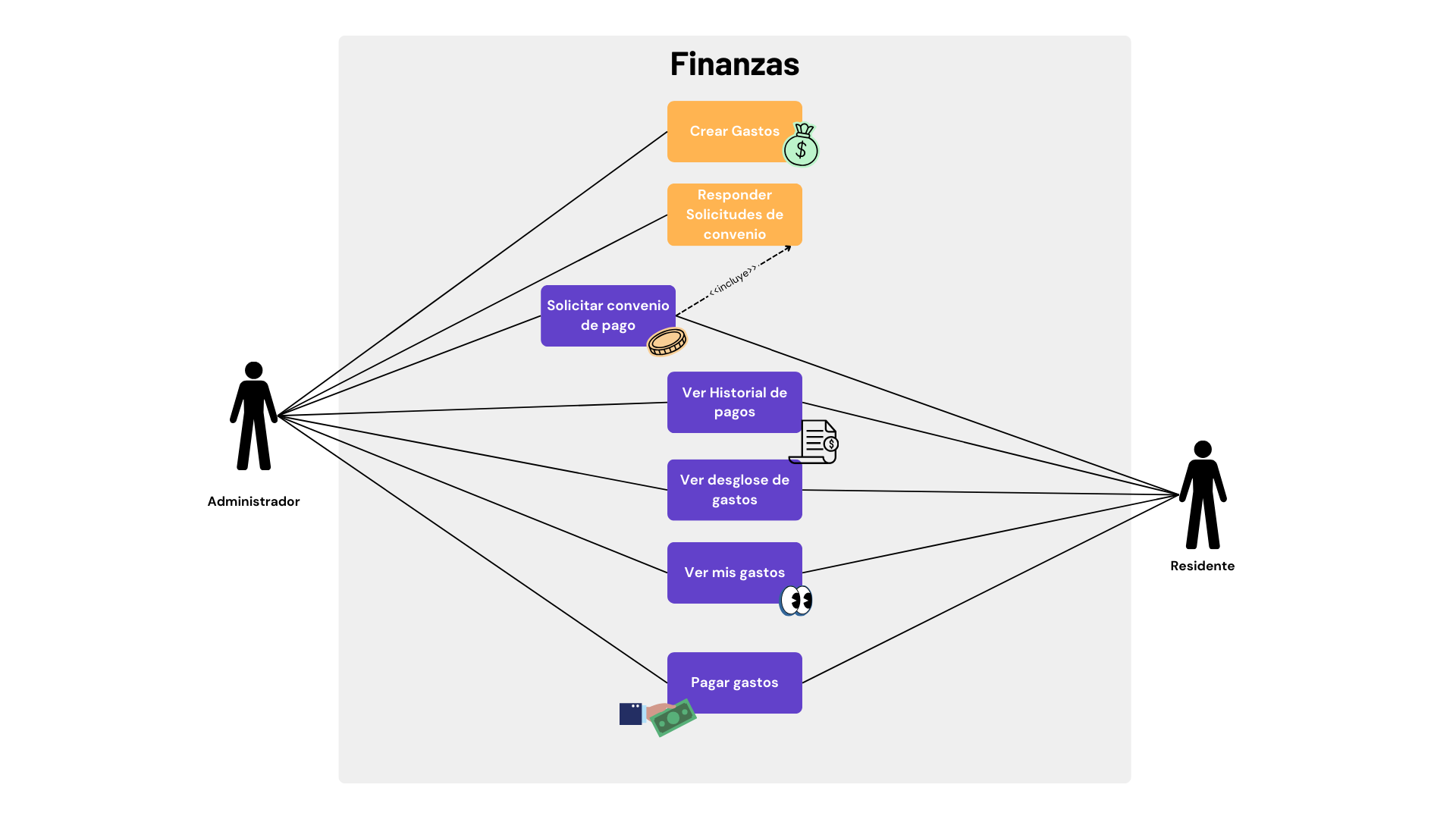
Esta sección describe en detalle el conjunto de escenarios funcionales y no funcionales que obtuvieron la mayor prioridad en el análisis. Para esto se presenta y describe el diagrama de casos de uso y los casos de uso prioritarios, así como los escenarios en que uno o más atributos de calidad se ven involucrados de manera significativa.

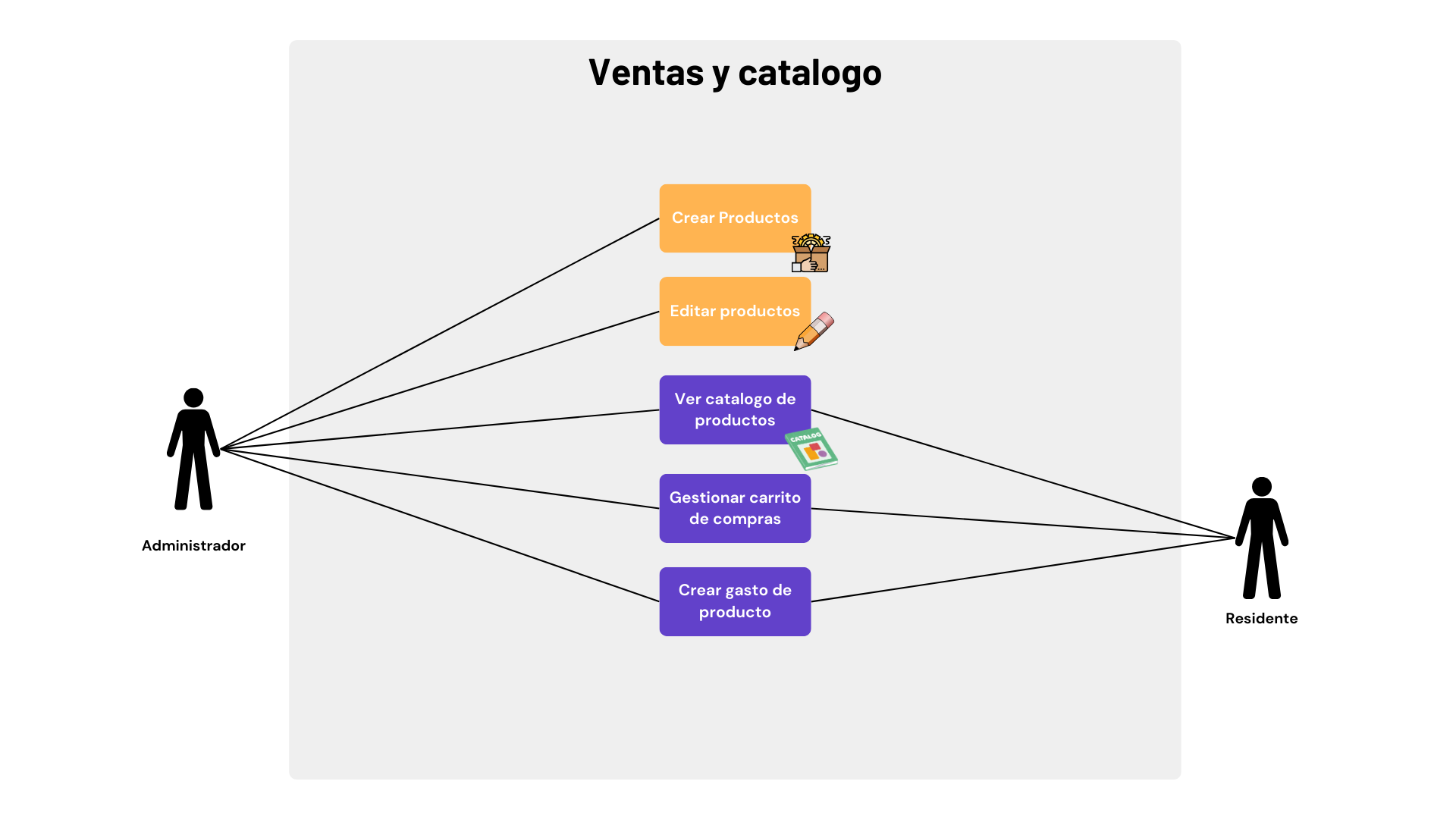
* 1. **Diagrama de Casos de Uso**

El modelo de casos de uso puede ser encontrado en el documento “Casos de Uso”.

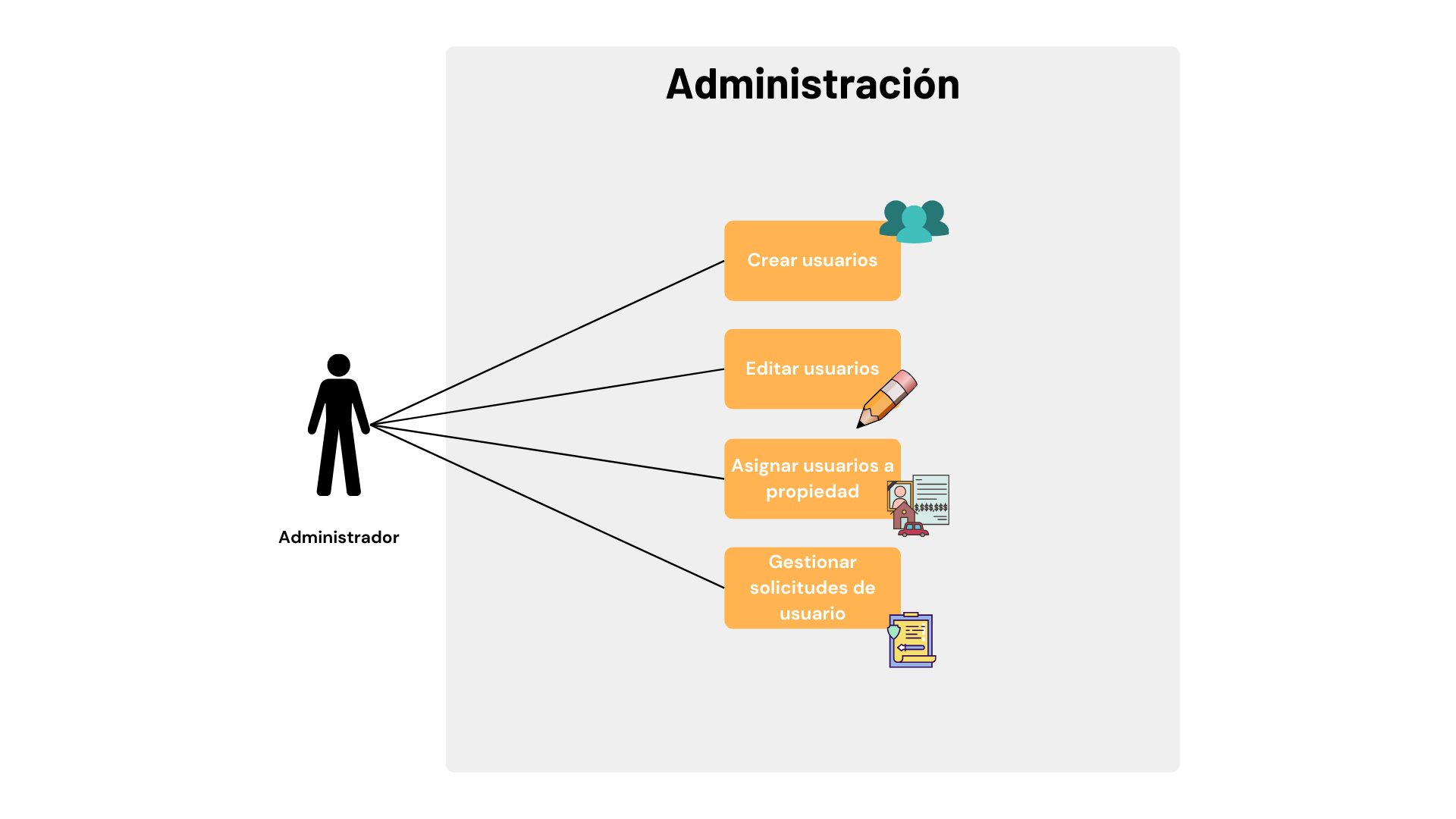


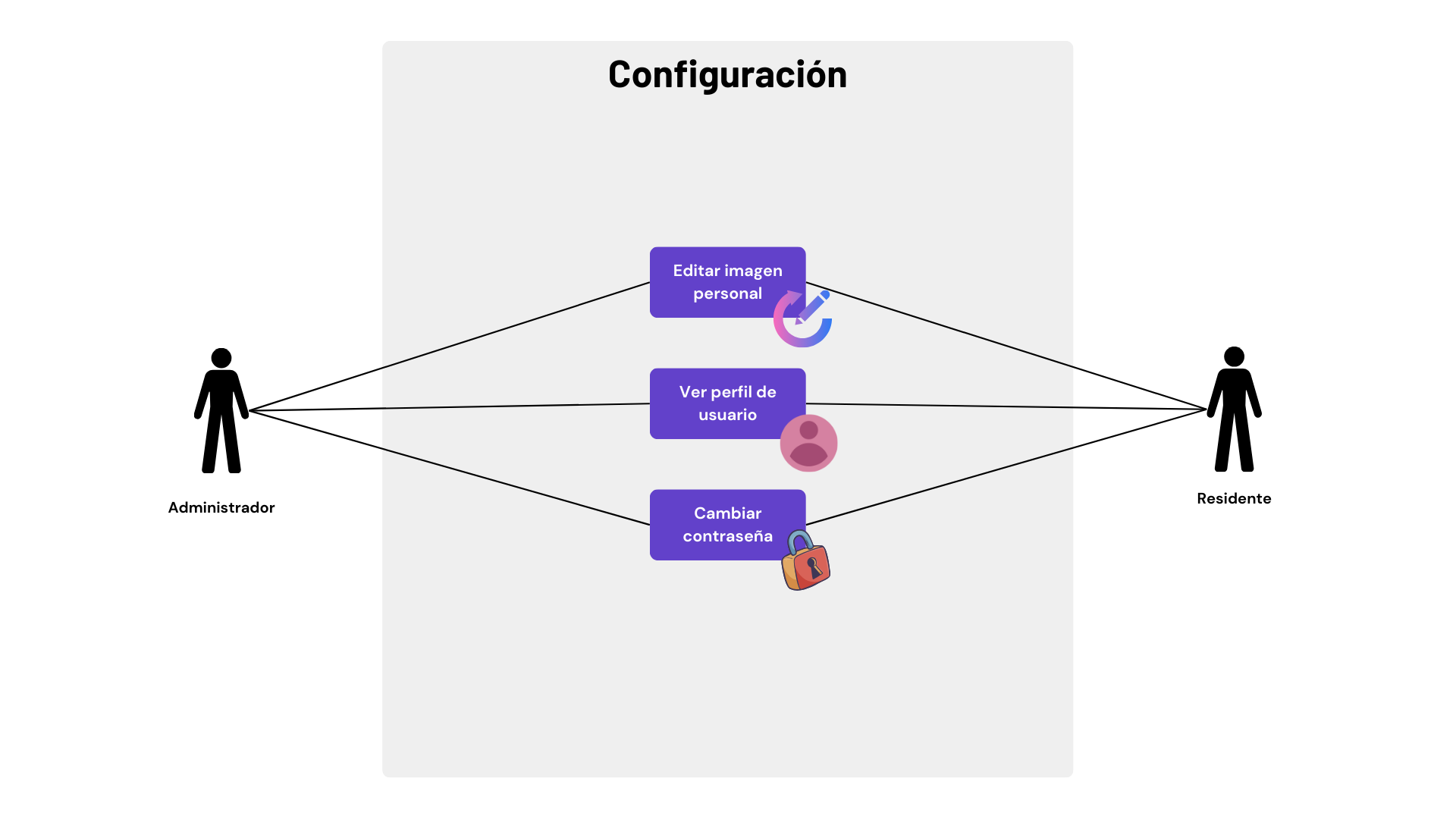












* 1. **Especificación de Casos de Uso Relevantes**

Los casos de uso considerados los más relevantes para el desarrollo de la arquitectura fueron determinados. Los criterios usados para dicha determinación fueron:

A continuación, se listan los casos de uso relevantes, los cuales pueden ser encontrados con su especificación detallada en el documento “Casos de Uso”.

* 1. **Especificación de los Escenarios de Calidad Relevantes**

Después de un análisis en conjunto con los stakeholders, los escenarios de calidad se expresan a continuación:

| Nombre del Requerimiento | Actores Relacionados | Descripción Corta del Requerimiento |
| --- | --- | --- |
| Registro de residentes | Administrador | Permitir al administrador registrar nuevos residentes en el sistema con sus datos personales y credenciales de acceso. |
| Iniciar sesión como residente | Residente | Permitir a los residentes iniciar sesión en la plataforma para acceder a sus datos y funcionalidades disponibles. |
| Modificar datos de residentes | Administrador | Facilitar al administrador la edición de datos personales de los residentes registrados en caso de cambios o actualizaciones. |
| Visualizar cuentas de residentes | Administrador | Permitir al administrador consultar un listado con los datos de todos los residentes registrados. |
| Registrar gastos comunes | Administrador | Registrar y clasificar gastos comunes asociados al condominio para su posterior prorrateo entre los residentes. |
| Visualizar y gestionar mis gastos | Residente | Ofrecer a los residentes acceso a un listado detallado de los gastos comunes asociados a sus propiedades. |
| Realizar pagos de gastos comunes | Residente | Permitir a los residentes efectuar pagos de gastos comunes a través de métodos digitales como Webpay o su billetera digital. |
| Generar convenios de pago | Administrador, Residente | Facilitar la solicitud, aprobación o rechazo de convenios de pago para deudas de gastos comunes. |
| Restringir compras por morosidad | Administrador | Impedir que los residentes con deudas puedan adquirir productos o servicios adicionales hasta que regularicen su situación. |
| Crear catálogo de productos | Administrador | Permitir al administrador registrar y actualizar productos disponibles para la venta en el catálogo del condominio. |
| Implementar carrito de compras | Residente | Ofrecer a los residentes la opción de agregar productos al carrito de compras y proceder al pago tras su revisión. |
| Generar reportes personalizados | Administrador | Permitir la creación de reportes en formatos como PDF o Excel, con información relevante de pagos, deudas y estadísticas del condominio. |
| Exportar base de datos a Excel | Administrador | Proveer una funcionalidad para exportar la base de datos del sistema a un archivo Excel para su análisis externo. |
| Enviar recibos de pago automáticamente | Residente | Generar y enviar recibos electrónicos a los residentes tras la realización de pagos exitosos en la plataforma. |

| Escenario de Calidad N°1 | |
| --- | --- |
| Atributo de Calidad Asociado (Característica):Compatibilidad  Sub Característica:Interoperabilidad | |
| Descripción: | |
|  | |
| Fuente del Estímulo: | Usuario |
| Estímulo: | Realizar la función de pagar de manera online |
| Artefacto: | Software |
| Ambiente: | Pago |
| Respuesta: | Webpay permitirá el pago de los servicios en línea de la página descontando el pago del cliente y depositandolo a la empresa |
| Medida de Respuesta: | Entre 2 y 6 segundos |

| Escenario de Calidad N°2 | |
| --- | --- |
| Atributo de Calidad Asociado (Característica): Portabilidad  Sub Característica: Adaptabilidad | |
| Descripción: El sistema debe tener compatibilidad con diferentes sistemas operativos del mercado para poder ejecutarse sin ningún problema. | |
|  | |
| Fuente del Estímulo: | Usuario |
| Estímulo: | Ingresar al sistema desde un dispositivo de Windows o Macintosh |
| Artefacto: | Software |
| Ambiente: | Ingreso |
| Respuesta: | El sistema esté disponible para su uso independiente de que sistema operativo esté usando. |
| Medida de Respuesta: | Inmediata |

| Escenario de Calidad | |
| --- | --- |
| Atributo de Calidad Asociado (Característica): Fiabilidad  Sub Característica: Disponibilidad | |
| Descripción: El sistema deberá atender 24x7 | |
|  | |
| Fuente del Estímulo: | Usuario |
| Estímulo: | El usuario ingresa al sistema |
| Artefacto: | Software |
| Ambiente: | Ingreso |
| Respuesta: | El sistema esté disponible para su uso y este tiene sus respectivos servicios funcionando. |
| Medida de Respuesta: | Inmediato |

| Escenario de Calidad N°4 | |
| --- | --- |
| Atributo de Calidad Asociado (Característica): Eficiencia de desempeño  Sub Característica: Comportamiento Temporal | |
| Descripción: El sistema debe tener un tiempo de respuesta corto para una determinada tarea, debe tener una baja utilización de recursos computacionales y un tiempo corto de transmisión de datos. | |
|  | |
| Fuente del Estímulo: | Usuario |
| Estímulo: | Ejecutar determinadas tareas. |
| Artefacto: | Software |
| Ambiente: | Interacción con el sistema |
| Respuesta: | Bajo tiempo de respuesta al ejecutar las tareas del sistema |
| Medida de Respuesta: | Menos de 4 segundos |

| Escenario de Calidad N°5 | |
| --- | --- |
| Atributo de Calidad Asociado (Característica): Usabilidad  Subcaracterística: Estetica de la interfaz de usuario | |
| Descripción: Integrar los colores de la empresa en el software. | |
|  | |
| Fuente del Estímulo: | Usuario |
| Estímulo: | Entrar a cualquier función al sistema |
| Artefacto: | Software |
| Ambiente: | Normal |
| Respuesta: | El sistema deberá estar constituido con los colores representativos de la empresa |
| Medida de Respuesta: | Inmediato |

| Escenario de Calidad N°6 | |
| --- | --- |
| Atributo de Calidad Asociado (Característica): Seguridad  Sub Característica: No repudio | |
| Descripción:El sistema debe tener la capacidad de respaldar las acciones o eventos que han sido descartados, de manera que los datos relacionados entre sí no fallen al deshacerse de algún dato. | |
|  | |
| Fuente del Estímulo: | Usuario |
| Estímulo: | Acciones en el sistema |
| Artefacto: | Base de datos |
| Ambiente: | Normal |
| Respuesta: | El sistema generar un respaldo/historial de las acciones  descartadas |
| Medida de Respuesta: | inmediata |

| Escenario de Calidad N°7 | |
| --- | --- |
| Atributo de Calidad Asociado (Característica): Usabilidad  Sub Característica: Capacidad para ser Usado | |
| Descripción: La interfaz debe ser llamativa e intuitiva con el usuario , para mejorar su nivel interactivo. | |
|  | |
| Fuente del Estímulo: | Usuario |
| Estímulo: | Interactuar con el Sistema |
| Artefacto: | Software |
| Ambiente: | Normal |
| Respuesta: | El sistema facilita el uso al usuario mediante el posicionamiento intuitivo de los elementos. |
| Medida de Respuesta: | Inmediata |

| Escenario de Calidad N°8 | |
| --- | --- |
| Atributo de Calidad Asociado (Característica): Seguridad  Sub Característica:Confidencialidad | |
| Descripción :El sistema debe brindar protección contra el acceso de datos e información no autorizados, ya sea accidental o deliberadamente. | |
|  | |
| Fuente del Estímulo: | Intruso |
| Estímulo: | Intentar ingresar al sistema |
| Artefacto: | Software |
| Ambiente: | Normal |
| Respuesta: | El sistema debe reconocer al intruso y bloquear su acceso al sistema, protegiendo la integridad de los datos. |
| Medida de Respuesta: | Inmediato al no validar los datos ingresados. |

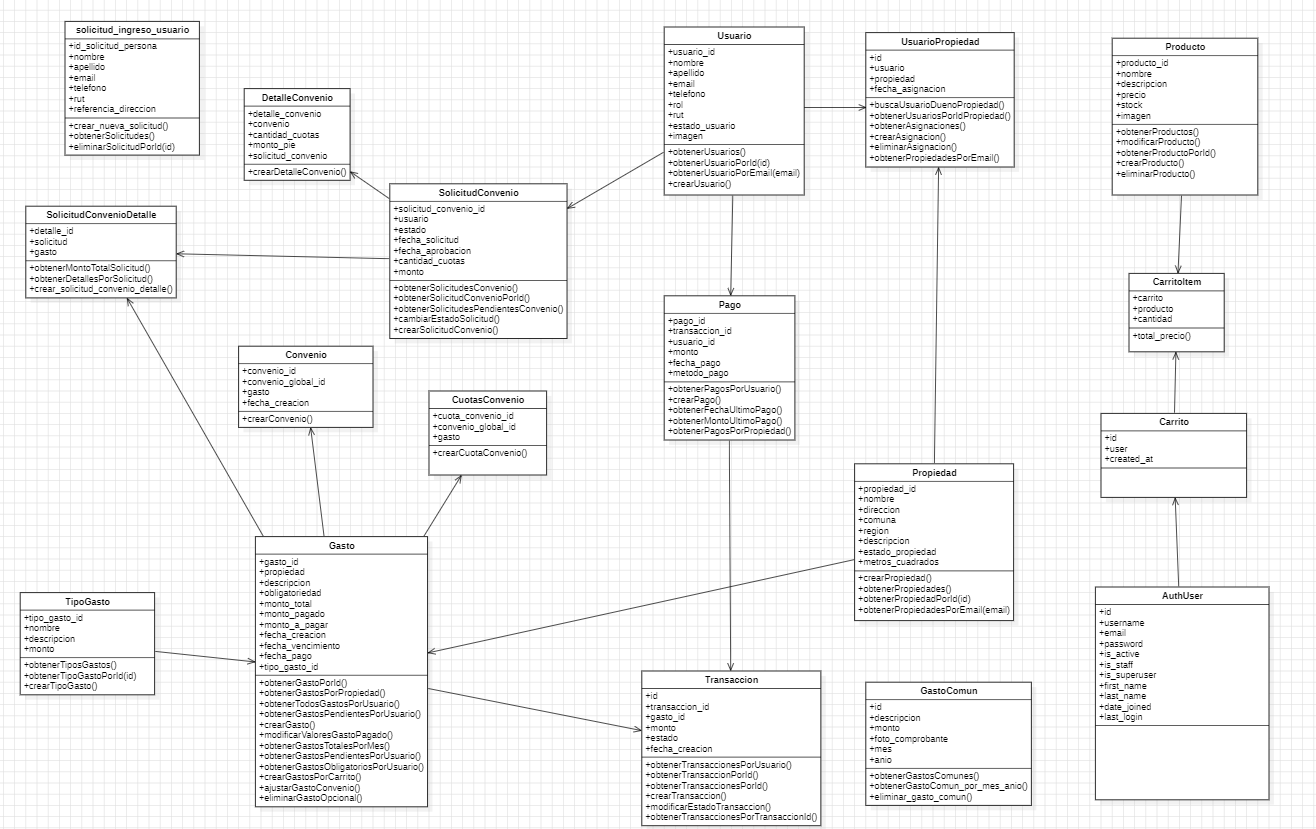
* 1. **Matriz de riesgos**

| **Riesgo** | **Categoría** | **Probabilidad** | **Impacto** | **Acción de Mitigación** | **Responsable** |
| --- | --- | --- | --- | --- | --- |
| Problemas en la configuración del entorno | Técnica | Alta | 2 | Verificar configuraciones en un ambiente de prueba antes de empezar el desarrollo | Equipo de desarrollo |
| Retrasos en la entrega de funcionalidades | Cronograma | Media | 3 | Reasignación de tareas o ajuste en el cronograma para permitir tiempos adicionales si fuera necesario | Equipo de desarrollo |
| Baja en la comunicación del equipo | Organizacional | Media | 2 | Establecer reuniones diarias para asegurar el intercambio de información, utilizando herramientas de gestión de tareas | Equipo de desarrollo |
| Fallos en la seguridad del sistema | Técnica | Baja | 3 | Realizar pruebas de seguridad en cada iteración para evitar brechas | Equipo de desarrollo |
| Cambios en los requisitos del cliente | Requerimientos | Media | 2 | Documentar y gestionar los cambios en el alcance, evaluar impacto en tiempo y esfuerzo | Equipo de desarrollo |
| Dificultades en la integración con Webpay | Técnica | Media | 2 | Establecer pruebas tempranas de integración con Webpay, asegurar una comunicación fluida con el equipo de soporte de Webpay | Equipo de desarrollo |
| Sobrecarga en el servidor durante horas pico | Técnica | Baja | 3 | Realizar pruebas de carga, optimizar el rendimiento del sistema para soportar altas demandas | Equipo de desarrollo |
| Desacuerdo entre miembros del equipo | Organizacional | Baja | 2 | Implementar reuniones de mediación, promover comunicación efectiva y un ambiente de trabajo positivo para resolver conflictos | Equipo de desarrollo |
| Feedback insuficiente de los usuarios | Usuarios | Media | 2 | Programar pruebas de usuario en etapas tempranas, analizar sus comentarios e implementar los ajustes necesarios | Equipo de desarrollo |
| Problemas en la migración de datos | Técnica | Baja | 3 | Crear un plan de respaldo de datos, realizar pruebas piloto para asegurar la integridad en la migración de datos | Equipo de desarrollo |

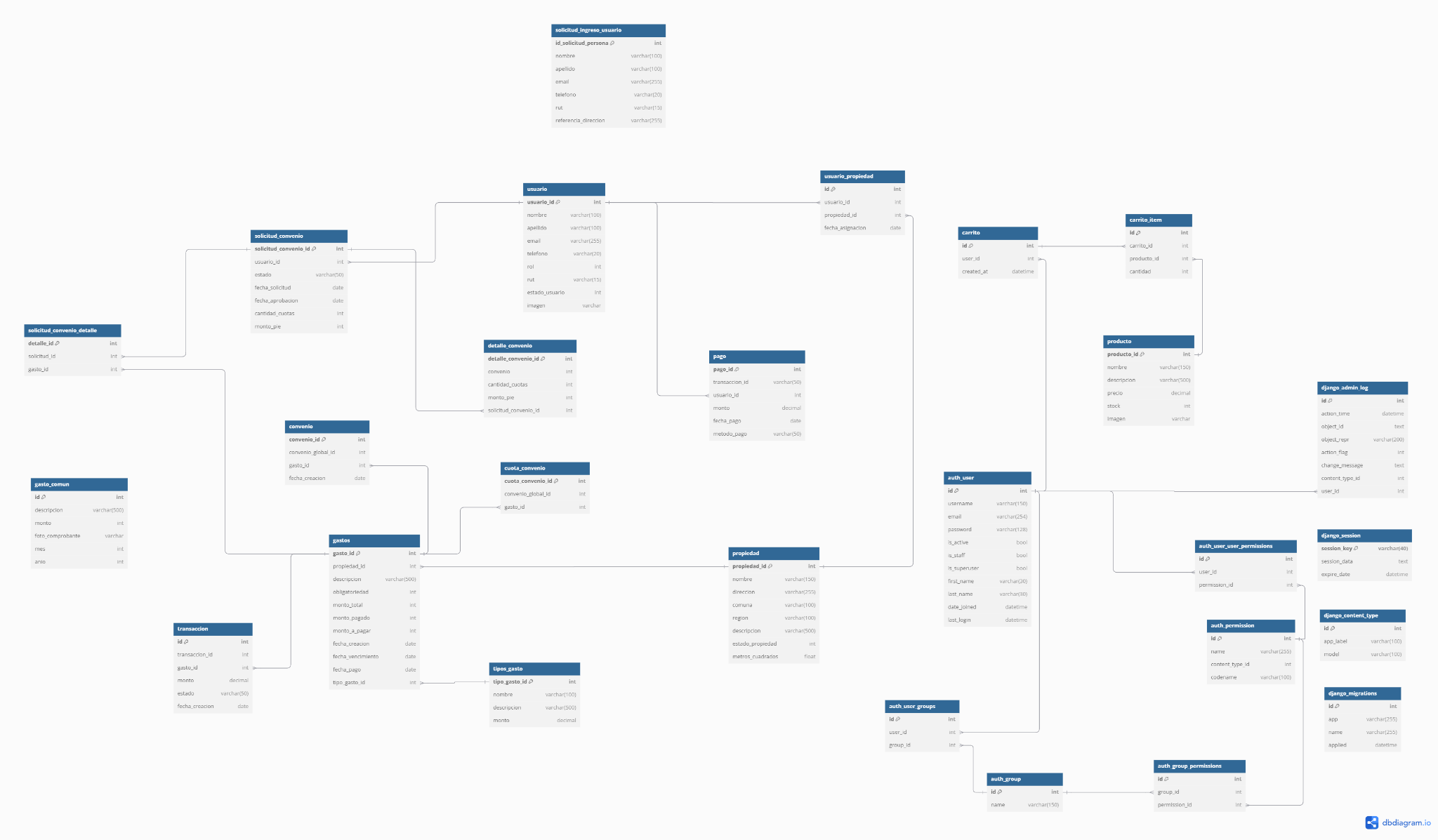
**Vista Lógica**

A continuación, se presenta una vista lógica de la aplicación expresado en dos diagramas, uno de ellos que muestra la parte estructural o estática de la aplicación (módulos), y otra vista que representa la parte dinámica (componentes y conectores).

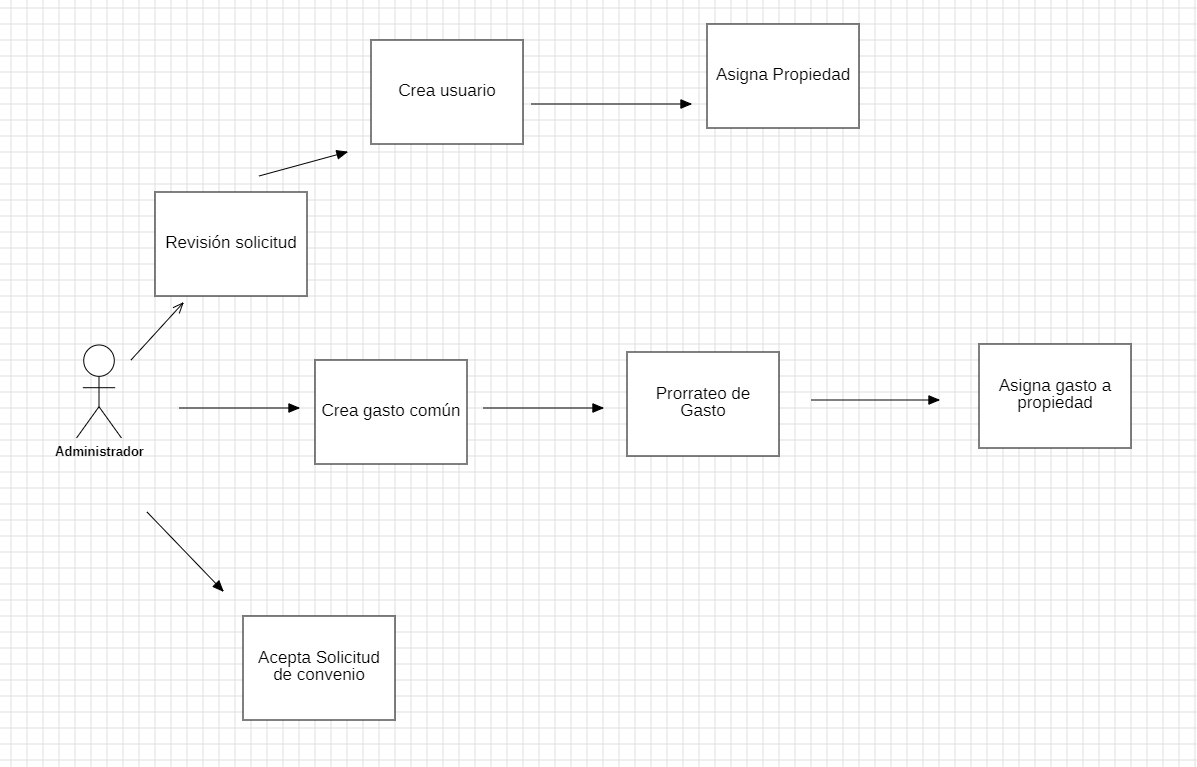
**DIAGRAMA DE CLASES**

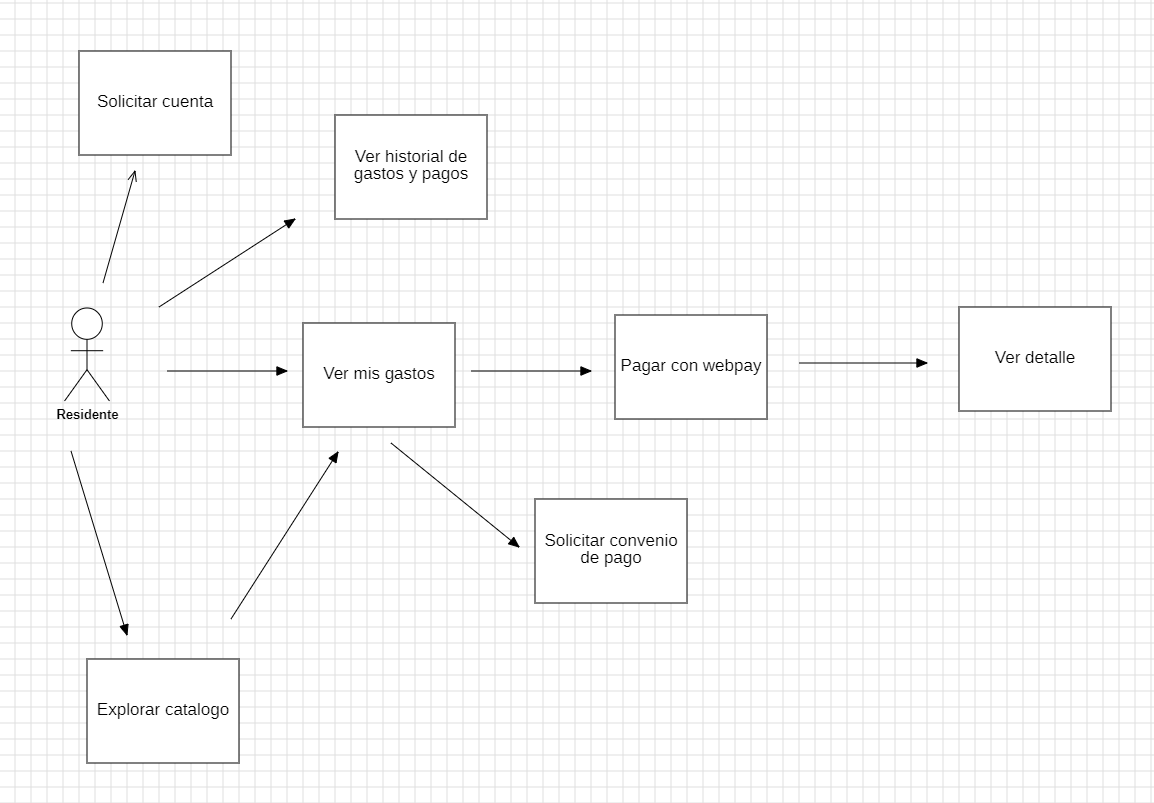
****

**DIAGRAMA DE ENTIDAD RELACIÓN**



**DIAGRAMA DE COMUNICACIÓN**

****



1. **Vista de Implementación**

[Diagrama de topología](#bookmark=id.f37en4om8uh4)

**5.1 Vista de Despliegue**

En esta vista se despliegan los nodos que participan con el sistema. Los nodos principales son los nodos Servidor de Integración. Características a continuación:

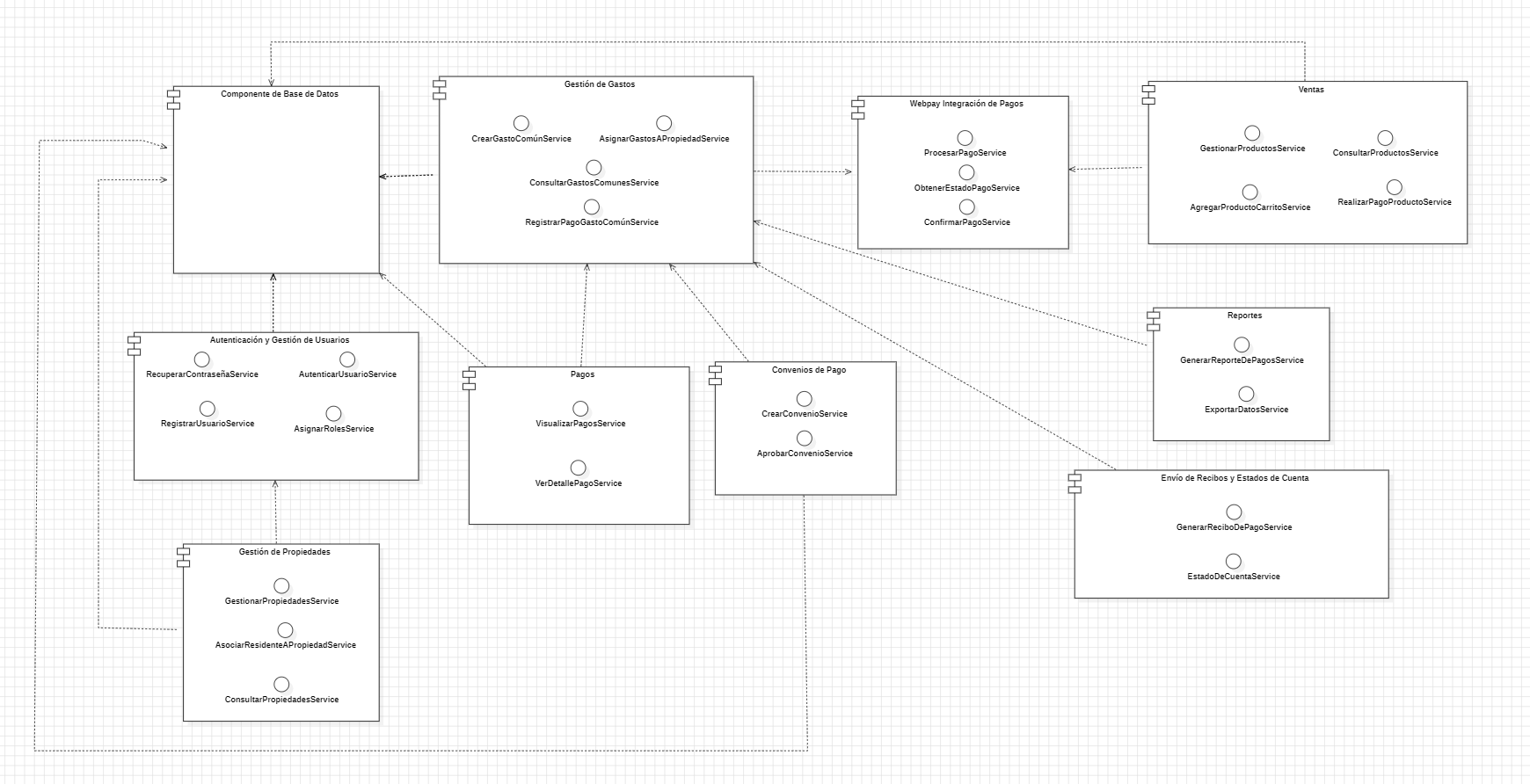
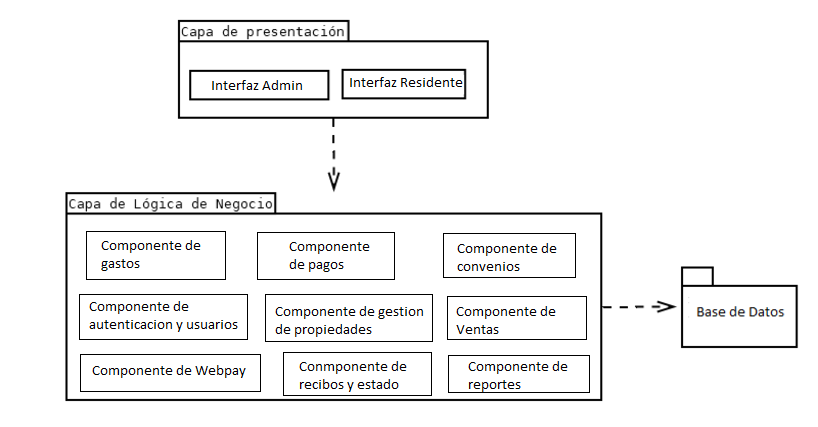
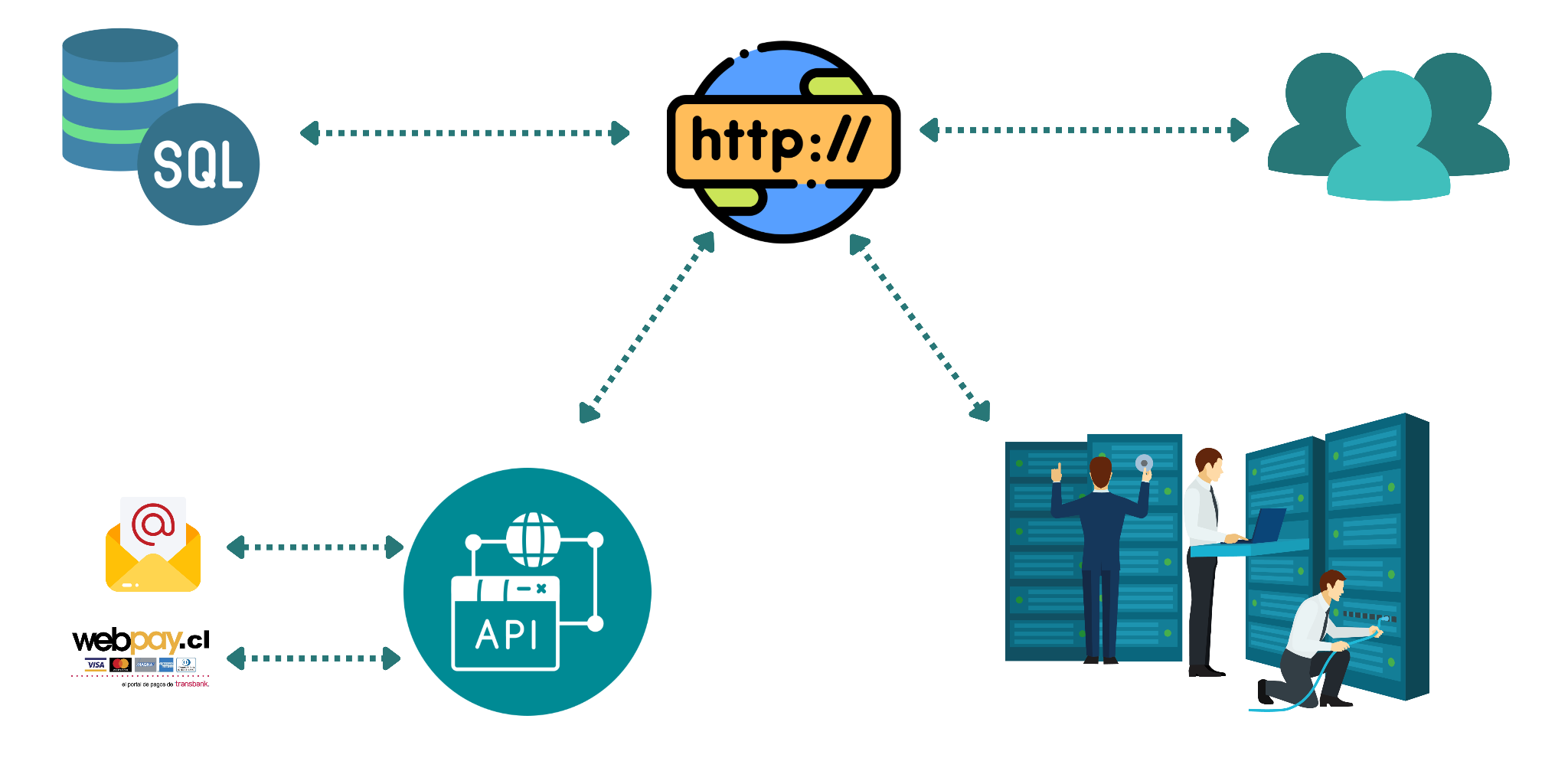
Diagrama de componentes

Diagrama de paquetes

**5.2** **Vista Física**

Diagrama de topología



**Vista de escenarios**

Los diagramas de casos de usos se pueden encontrar en el punto 3.1

1. **Decisiones de Diseño y Selección de Alternativas**

**Indicar acá cuáles son sus principales decisiones, como lenguaje, base de datos, conocimientos del equipo, tecnología, infraestructura, enlaces de telecomunicaciones, patrones de arquitectura a seguir**

Para el desarrollo del programa se utilizará el framework Django bajo su patrón de arquitectura MTV (Modelo Vista Template) con los lenguajes HTML, CSS y JS / JQUERY. Para la base de datos se utilizará Oracle Cloud y se mantendrá una conexión entre los trabajadores con github desktop.

* 1. **Análisis de Reutilización**

**¿Qué es lo que podemos reutilizar? programas, diagramas, arquitectura en general, etc, de qué proyecto se podrá reutilizar. Eventualmente si no hay otro proyecto, puede que no haya que reutilizar**

El proyecto se desarrollará desde cero al ser tan diferente de lo desarrollado anteriormente por los programadores.

***7.Patrón de arquitectura***

