

Revisiones

Fecha	Versión	Descripción	Autor
<16/10/17>	<1.0>	Version inicial	Grupo Tecno

Contenido:

1.	Introducción	2
1.1	Definiciones, Siglas y Abreviaturas	2
1.2	Referencias	2
2.	Representación Arquitectónica	3
3.	Objetivos de la Arquitectura y Restricciones	3
4.	Vista de Casos de Uso	3
4.1	Realizaciones de los Casos de Uso	7
5.	Vista Lógica	11
5.1	Visión General	14
5.2	Paquetes de Diseño Significativos	14
6.	Vista de Procesos	14
7.	Vista de Despliegue	15
8.	Vista de Implementación	16
8.1	Visión General	17
8.2	Capas	17
9.	Vista de Datos (opcional)	17
10.	Tamaños y Performance	17

1. Introducción

A través del siguiente documento se pretende mostrar una visión de la arquitectura del Sistema diseñado para la empresa Walt Disney Company que gestionara la venta de productos, registro de clientes y gestión de los parques y atracciones con el uso del “Fast Pass”, para el cual utilizaremos distintas vistas del software a fin de mostrar cómo realizamos nuestra solución, brindando una visión global y comprensible del diseño general del sistema.

1.1 Definiciones, Siglas y Abreviaturas

LEL DEL SISTEMA

Termino	Definición
MagicBand	Pulsera plastica que permite el ingreso al parque y atracciones, hacer uso de las reservas efectuadas con anterioridad vía web o mobile, y realizar transacciones de compras de alimentos o souvenirs.
FastPass	Reserva de turnos de atracciones o restaurants para evitar pérdidas de tiempo

[This subsection provides the definitions of all terms, acronyms, and abbreviations required to properly interpret the **Software Architecture Document**. This information may be provided by reference to the project’s Glossary.]

1.2 Referencias

OBJETIVO LIMITE Y ALCANCE O DOCUMENTOS QUE REFERENCIAN AL PROYECTO
Para comprender mejor ciertas definiciones e ideas planteadas en nuestra solución brindada recomendamos leer los siguientes documentos:

- Documento de especificación de objetivos, límite y alcance del Sistema Disney Fast Pass.
- Documento de presentación y muestra de las pantallas interactivas del Sistema Disney Fast Pass.
- Documento de especificación de diagramas UML del Sistema Disney Fast Pass.

2. Representación Arquitectónica

No aplica

[This section describes what software architecture is for the current system, and how it is represented. Of the **Use-Case**, **Logical**, **Process**, **Deployment**, and **Implementation Views**, it enumerates the views that are necessary, and for each view, explains what types of model elements it contains.]

3. Objetivos de la Arquitectura y Restricciones

No aplica

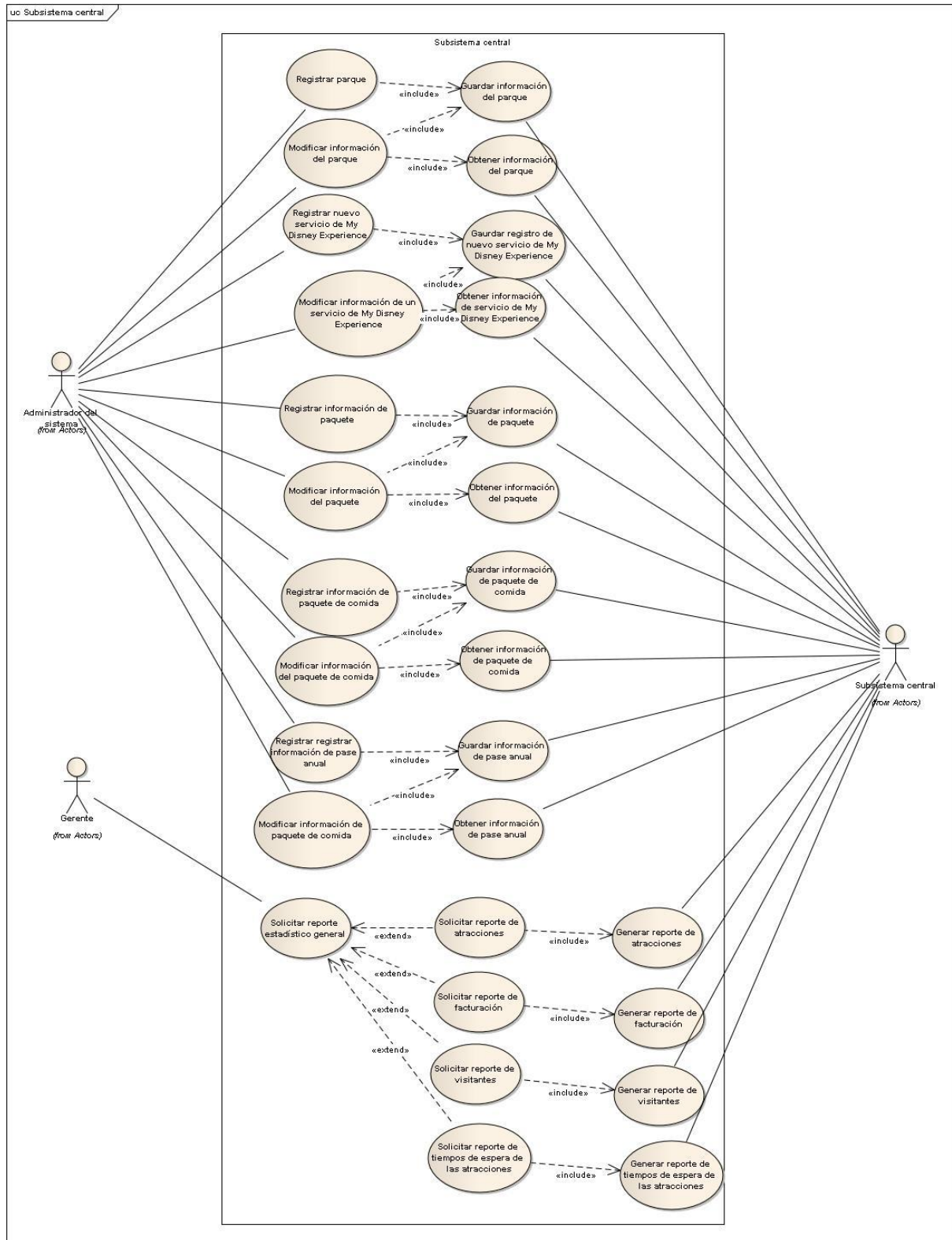
[This section describes the software requirements and objectives that have some significant impact on the architecture; for example, safety, security, privacy, use of an off-the-shelf product, portability, distribution, and reuse. It also captures the special constraints that may apply: design and implementation strategy, development tools, team structure, schedule, legacy code, and so on.]

4. Vista de Casos de Uso

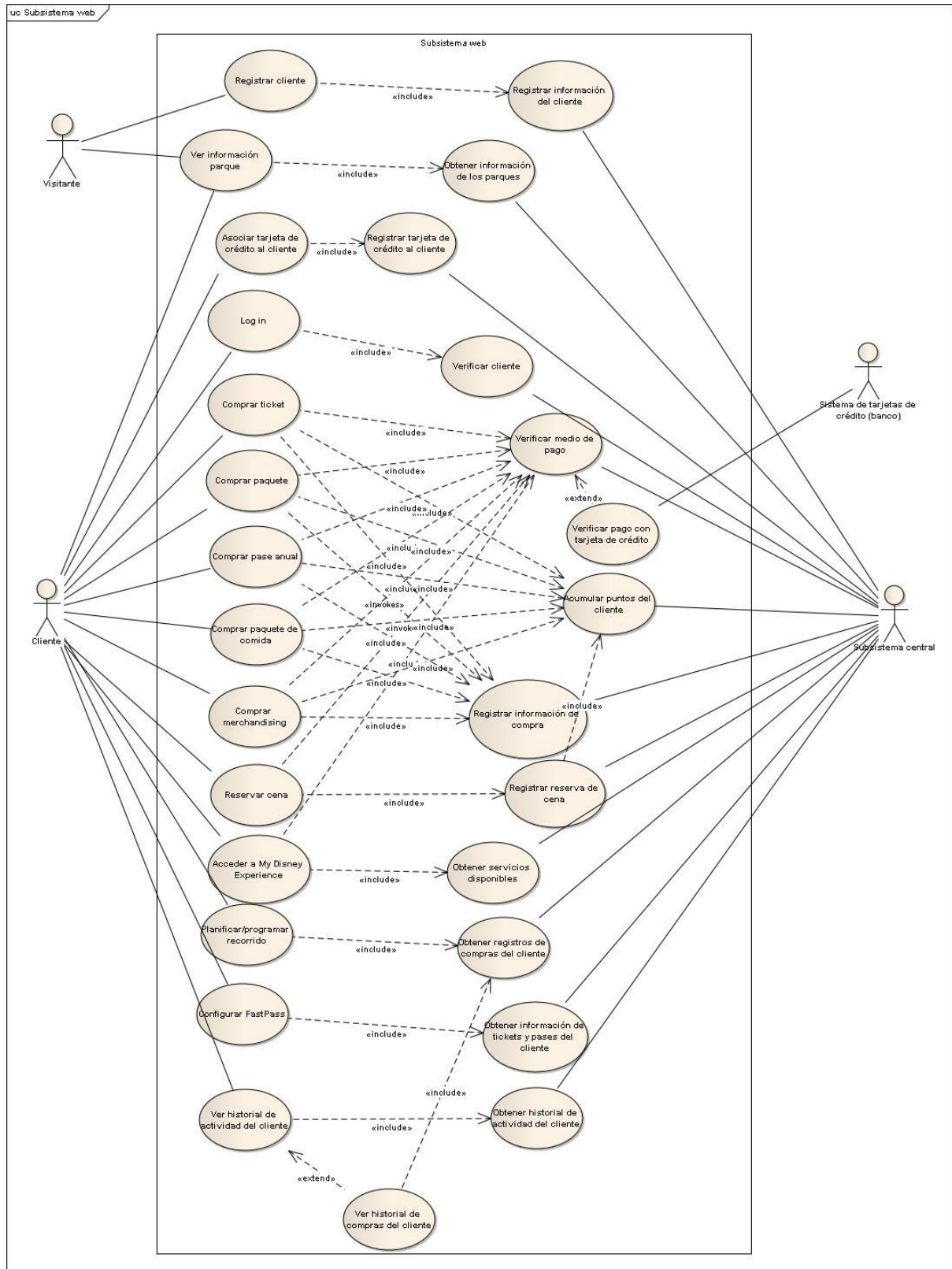
La Vista de Casos de Uso presenta un subconjunto del Modelo de Casos de Uso donde se representan cada una de las funcionalidades del producto.

Como puede verse en el mismo, se representan cuatro roles, donde se describen las acciones básicas que puede realizar una persona que sea Visitante, considerado aquel que no posee una cuenta registrada y aquella persona que es Cliente, donde sus posibles acciones abarcan mayores funcionalidades.

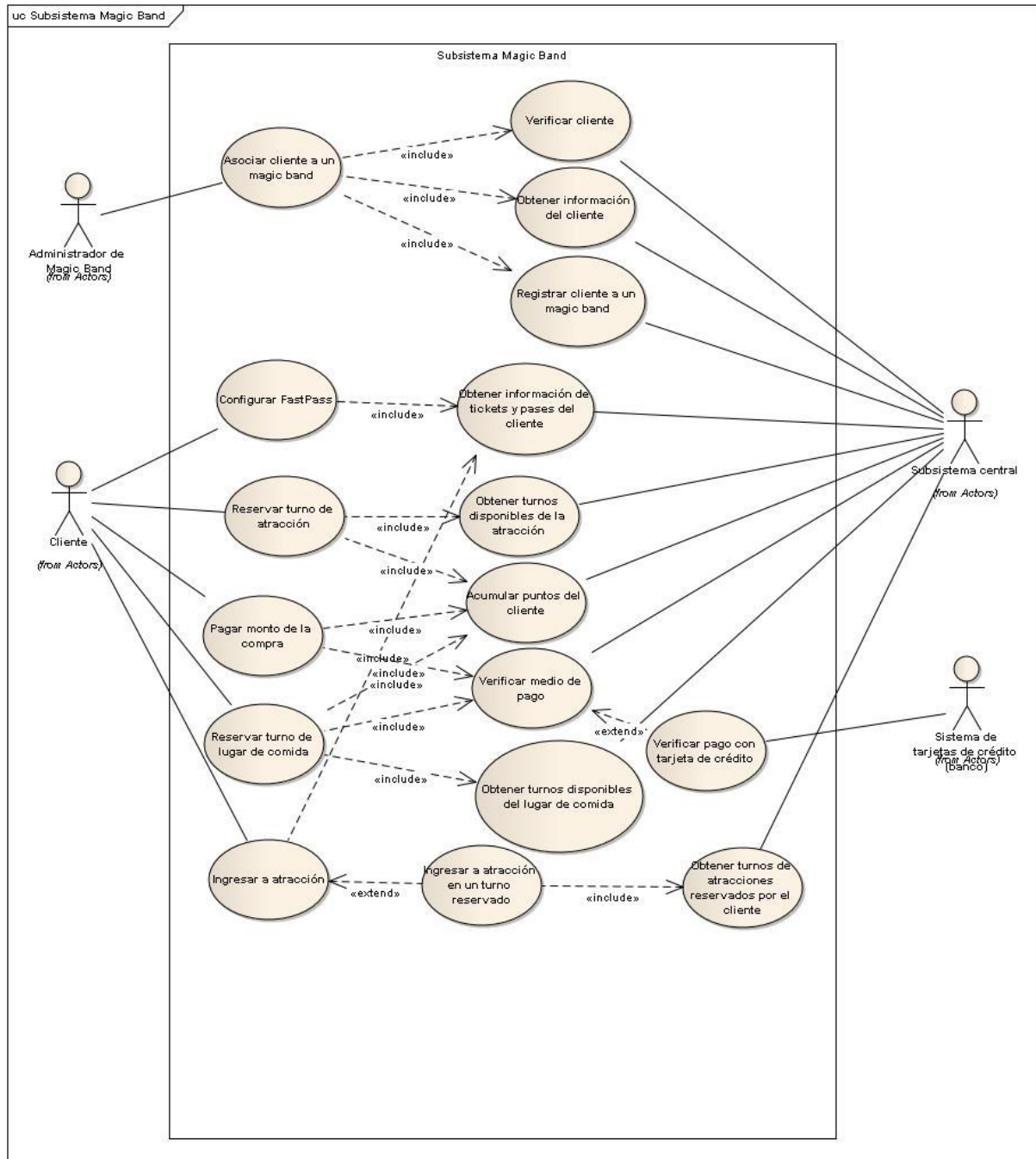
Subsistema Central



Subsistema Web



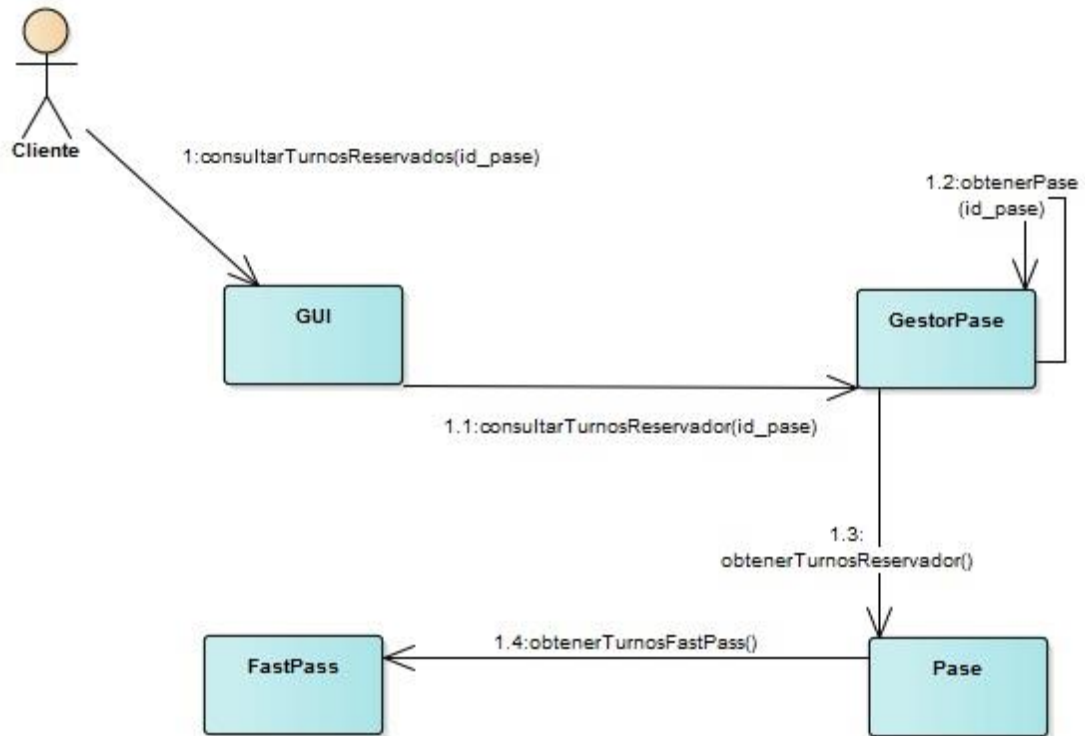
Subsistema MagicBand



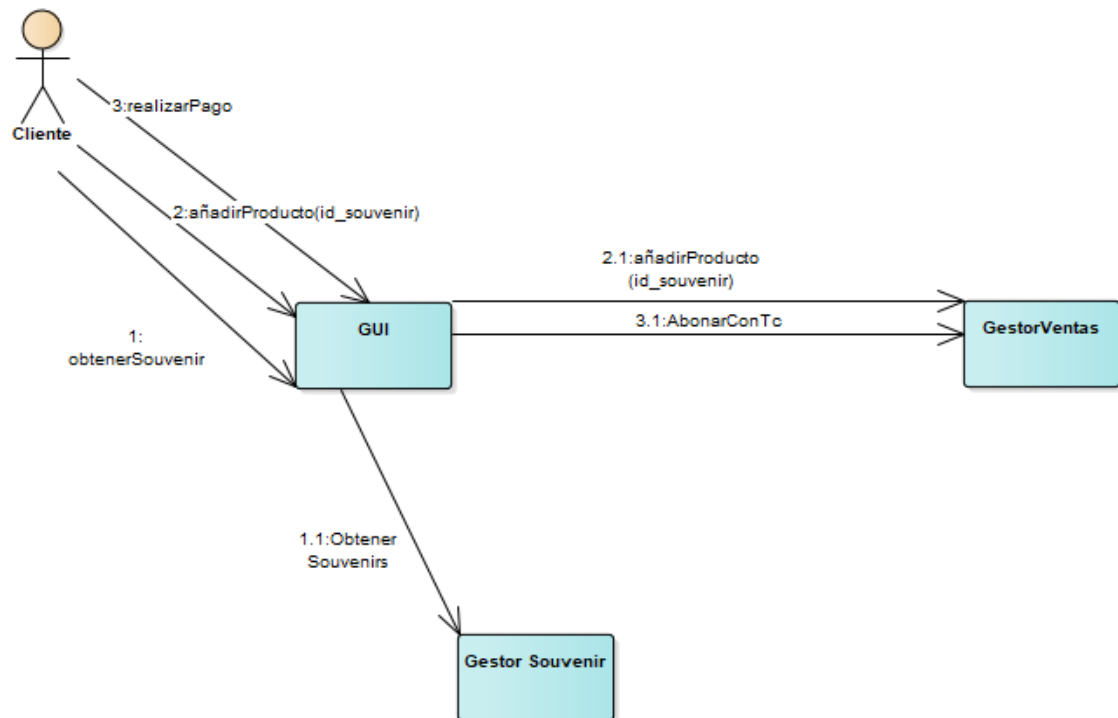
4.1 Realizaciones de los Casos de Uso

Se detalla a continuación el procedimiento de resolución del caso de uso más significativo para el entorno de nuestro sistema, donde un cliente realiza una consulta de los turnos reservados, reservar un turno y se le asigna una MagicBand para el uso del fast pass. Para los escenarios mencionados son modelados mediante diagramas de secuencia y de colaboración.

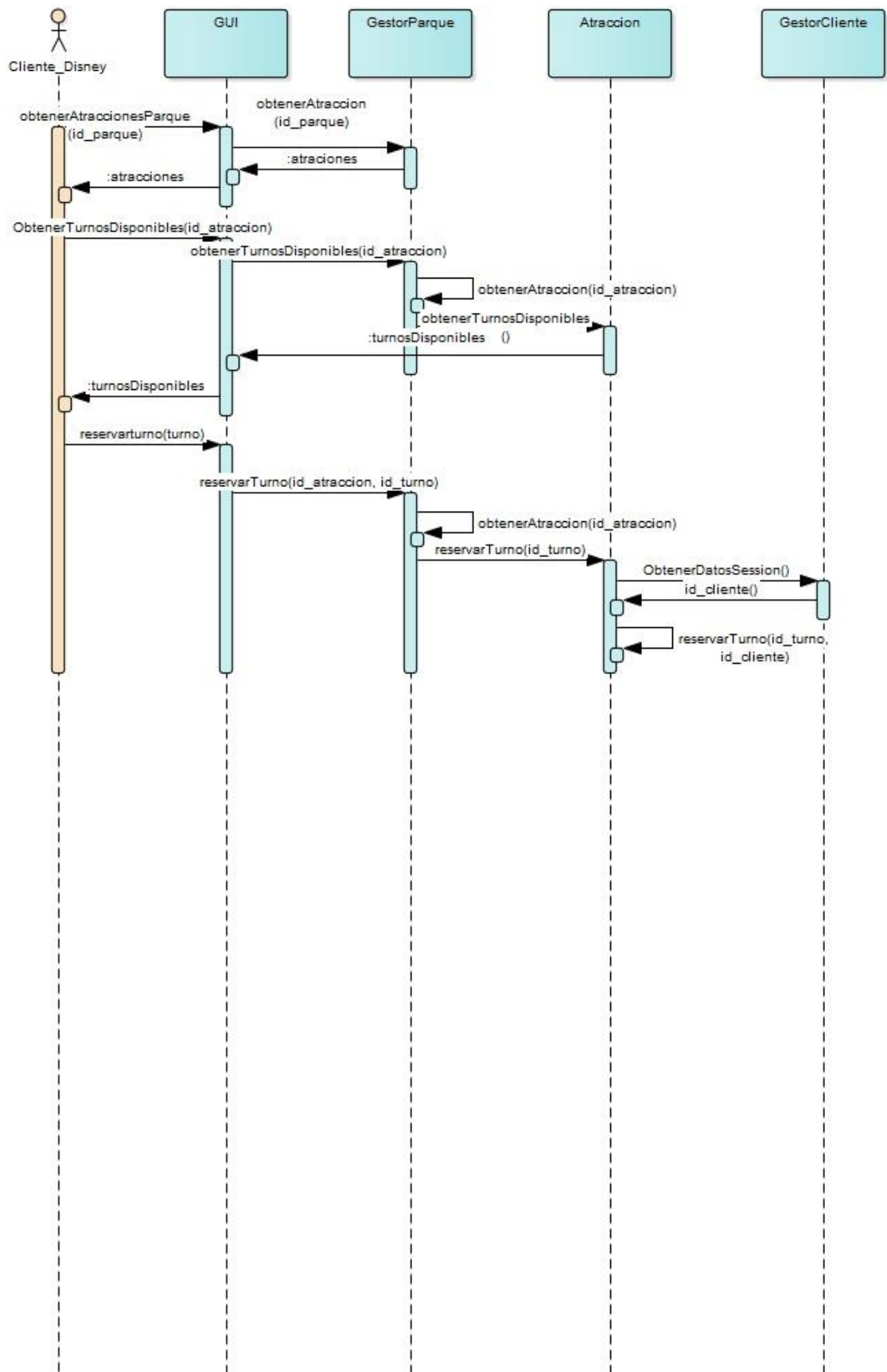
sd ConsultarTurnosReservados

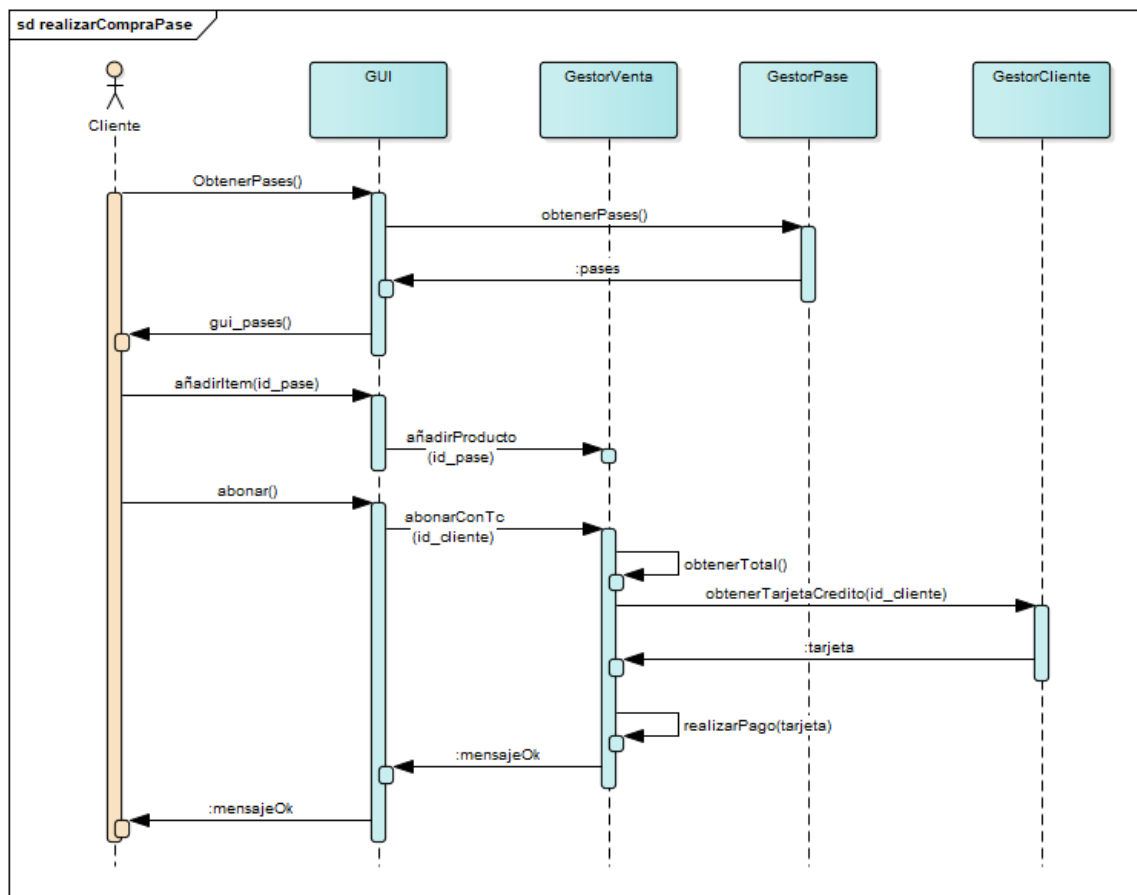
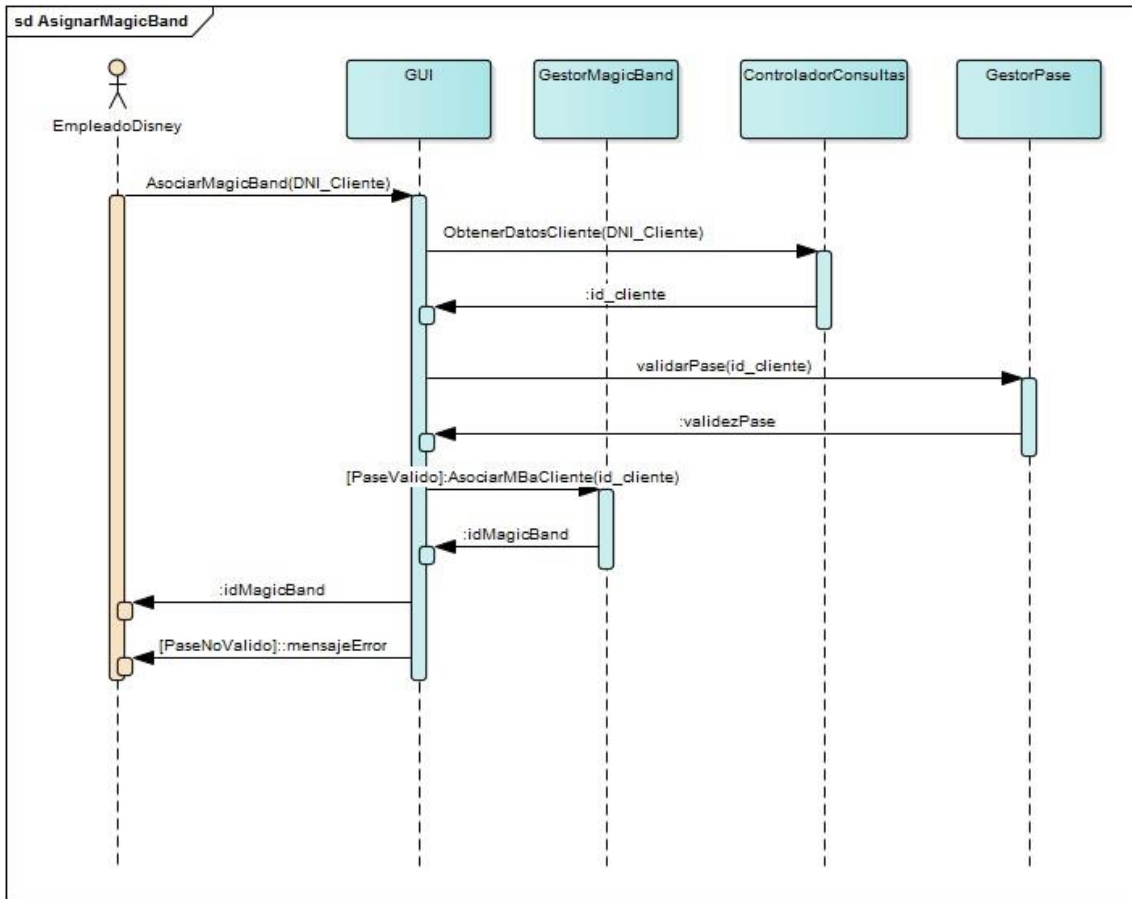


sd realizarCompraDe Souvenir



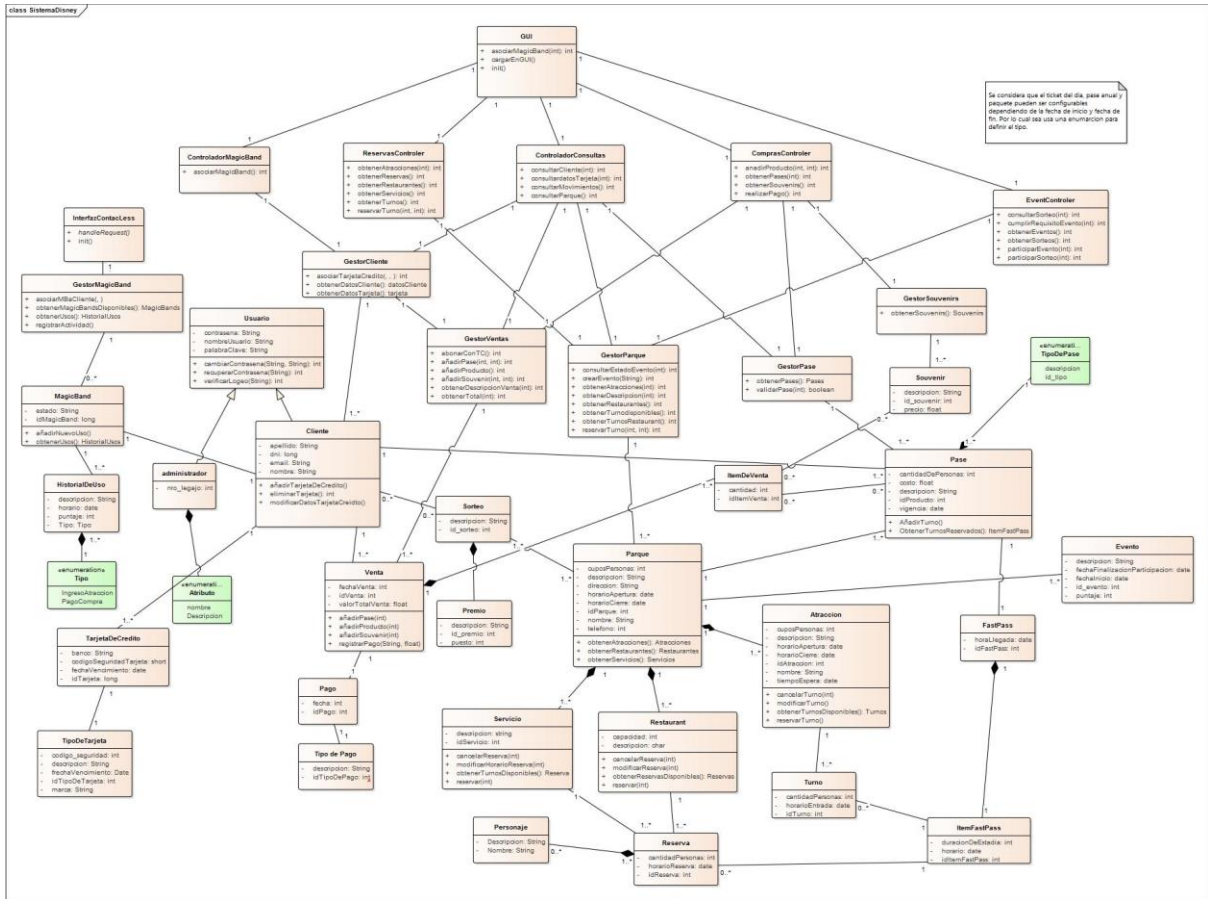
sd ReservarTurno



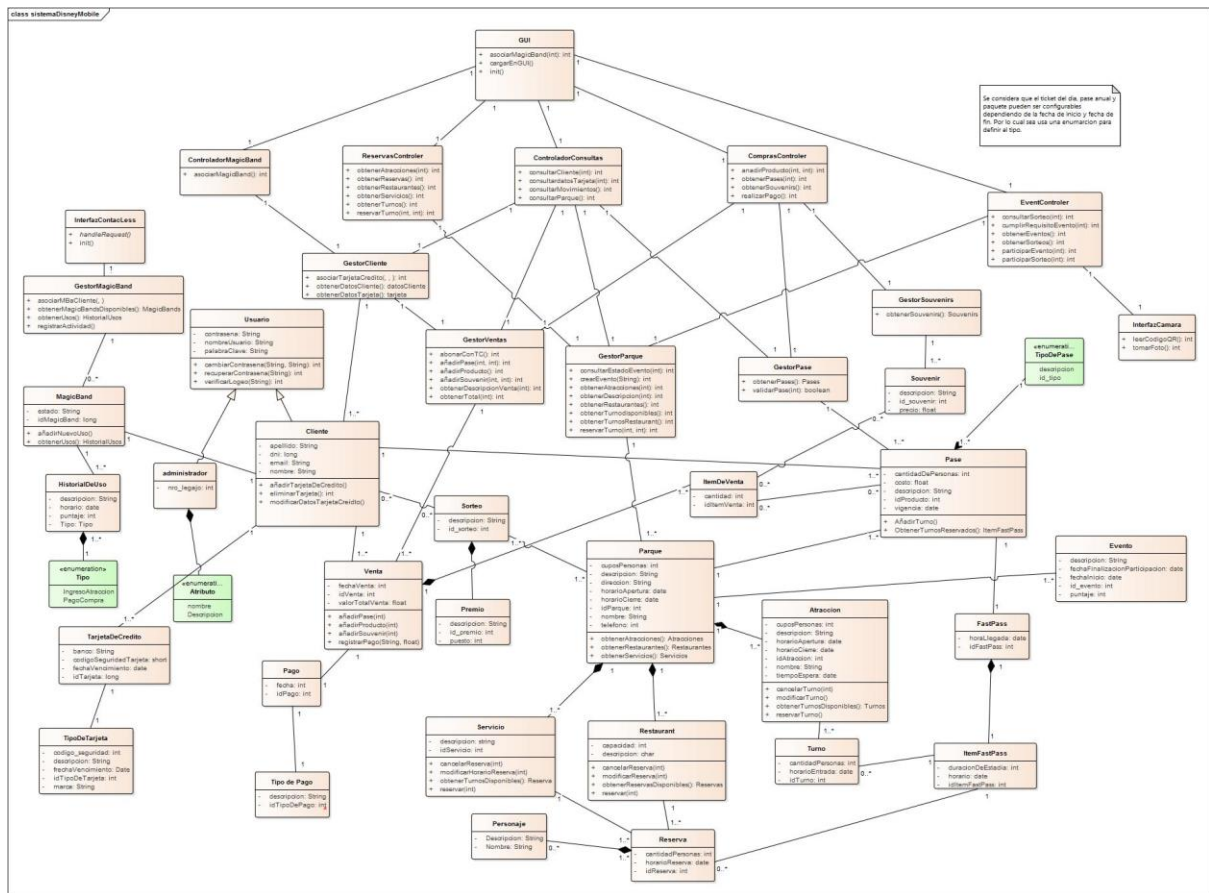


5. Vista Lógica

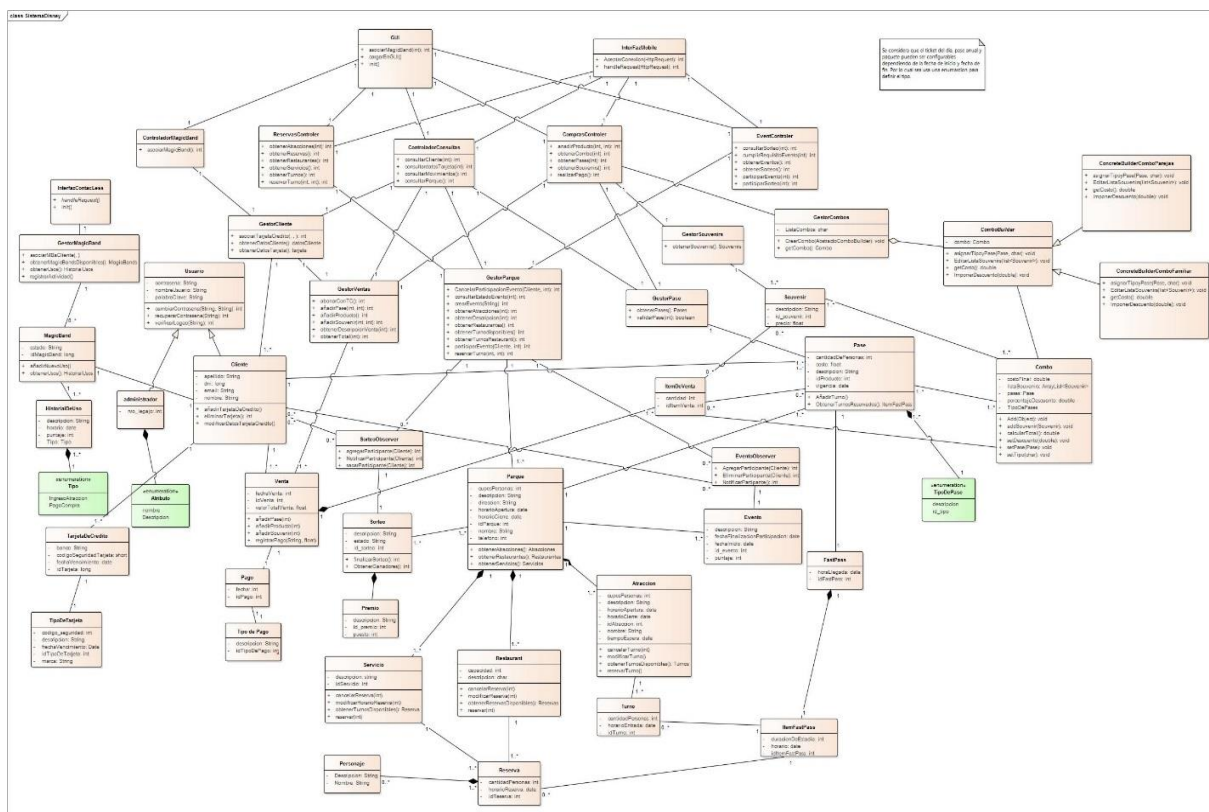
Sistema Web sin Patrones



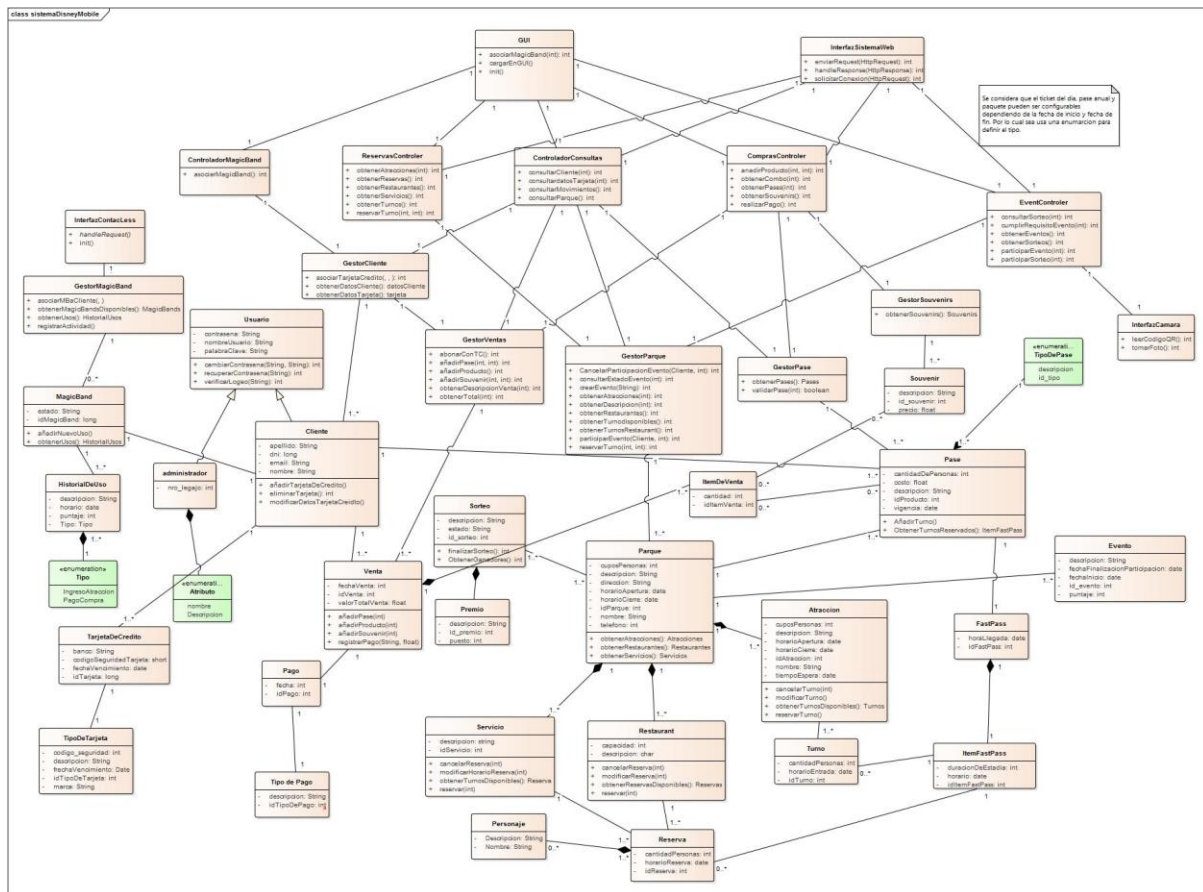
Sistema Mobile sin Patroni



Sistema Web con Patrones



Sistema Mobile con Patrones



Justificación

Observer

Utilizamos el patrón observer ya que queremos resolver el problema de cómo notificar al usuario cuando finaliza un evento o sorteo. Mediante el patrón observer un usuario se puede suscribir a un sorteo o evento y cuando el sorteo o evento cambia su estado la clase observador, que contiene a los clientes que están suscritos, le enviará una notificación del cambio de estado diciendo como salió en ese evento o sorteo

Builder

Utilizamos el patrón builder ya que queremos resolver el problema de cómo crear diferentes combos con descuentos los cuales tienen una cantidad de pases, tipo y souvenirs especificados. Mediante el patrón builder se puede crear varias clases concretas las cuales modifican a una instancia de la clase combo, por ejemplo, una clase concreta **ComboFamiliar** puede setear la instancia que se crea del combo con una cantidad de pases específicos, ejemplo de familia tipo 4 personas = 4 pases, con un listado de souvenirs propios para cada individuo y un descuento por comprar el combo de un x porcentaje. Esto

le da flexibilidad a la hora de crear promociones o vender pases para fechas poco concurridas, como por ejemplo, si existen fechas que segun la estadística baja la cantidad de gente que asiste al parque se crea un combo para dicho periodo de tiempo con el nombre “ComboPromoXFechas”.

5.1 Visión General

No aplica

[This subsection describes the overall decomposition of the design model in terms of its package hierarchy and layers.]

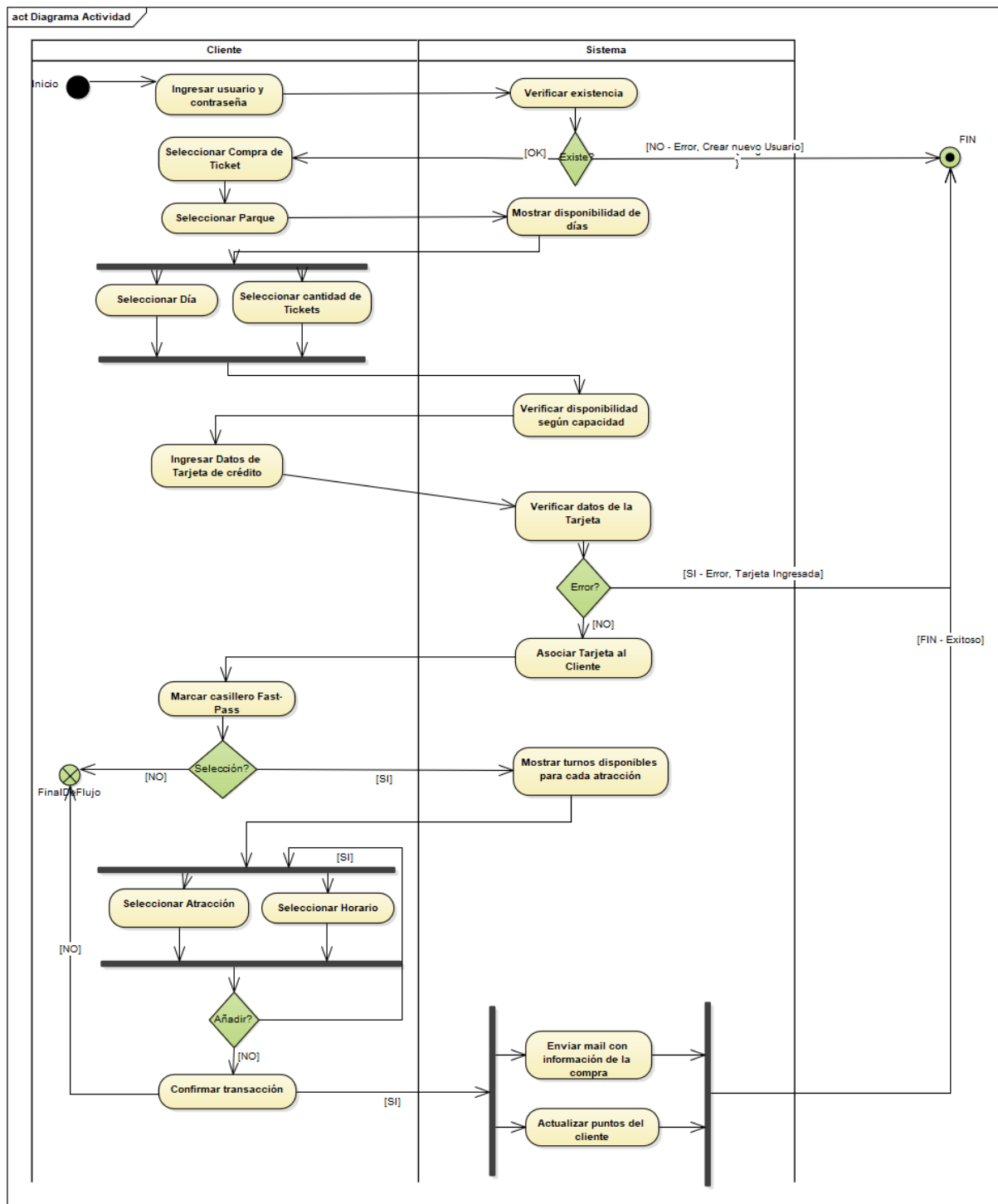
5.2 Paquetes de Diseño Significativos

[For each significant package, include a subsection with its name, its brief description, and a diagram with all significant classes and packages contained within the package.

For each significant class in the package, include its name, brief description, and, optionally, a description of some of its major responsibilities, operations, and attributes.]

6. Vista de Procesos

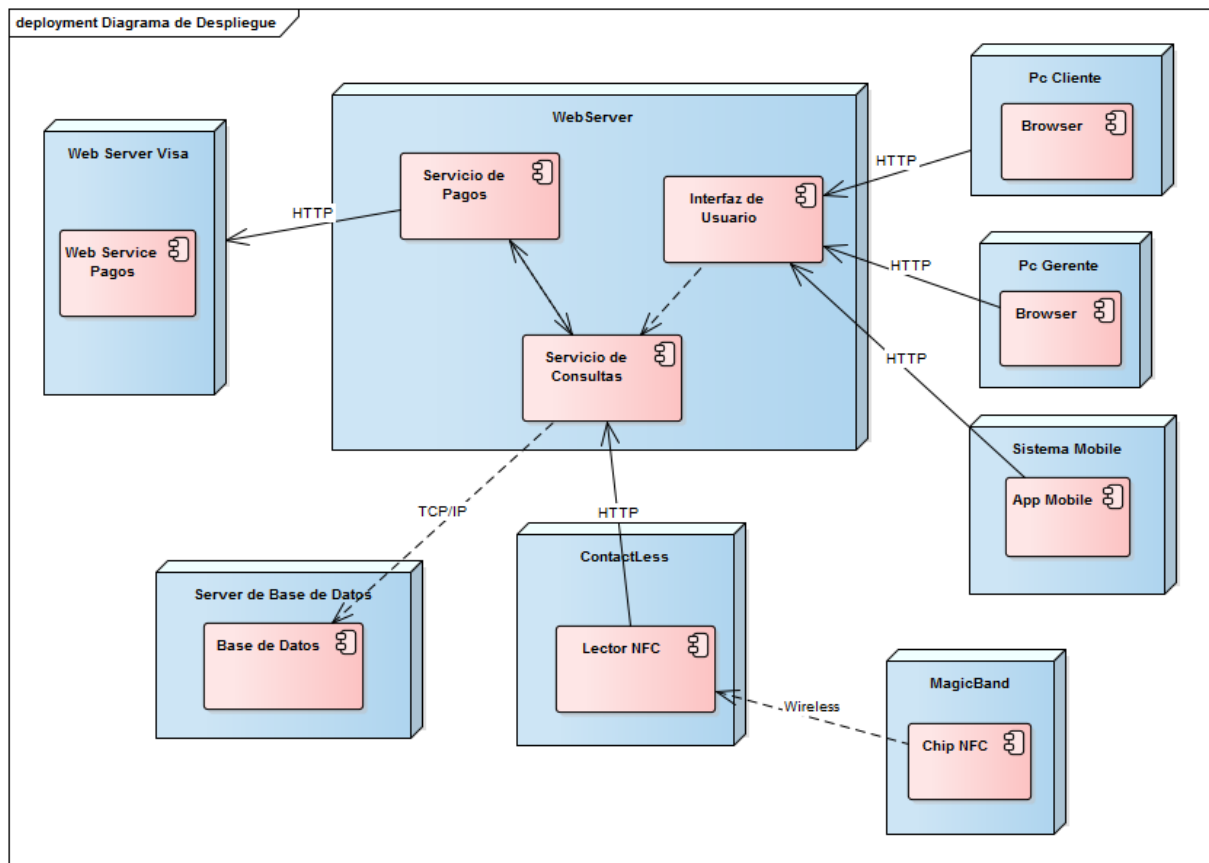
Se muestra cómo será el procedimiento donde un cliente realiza la compra de un ticket y cómo esta situación es resuelta, por un lado se muestra los pasos y las distintas actividades que pueden suceder en el transcurso al momento de realizar uno de los casos de uso más importantes que sería el de “Comprar Ticket” mediante el Diagrama de Actividad, Diagrama de Actividad “Comprar Ticket”



7. Vista de Despliegue

Se muestra una visión del sistema una vez ya afianzado en su continua funcionalidad, separando por un lado aquella aplicación de dispositivo Mobile que se conectan al sistema, así como también computadoras particulares que realicen conexiones con los diferentes roles y la conexión de las magicband.

Diagrama de despliegue



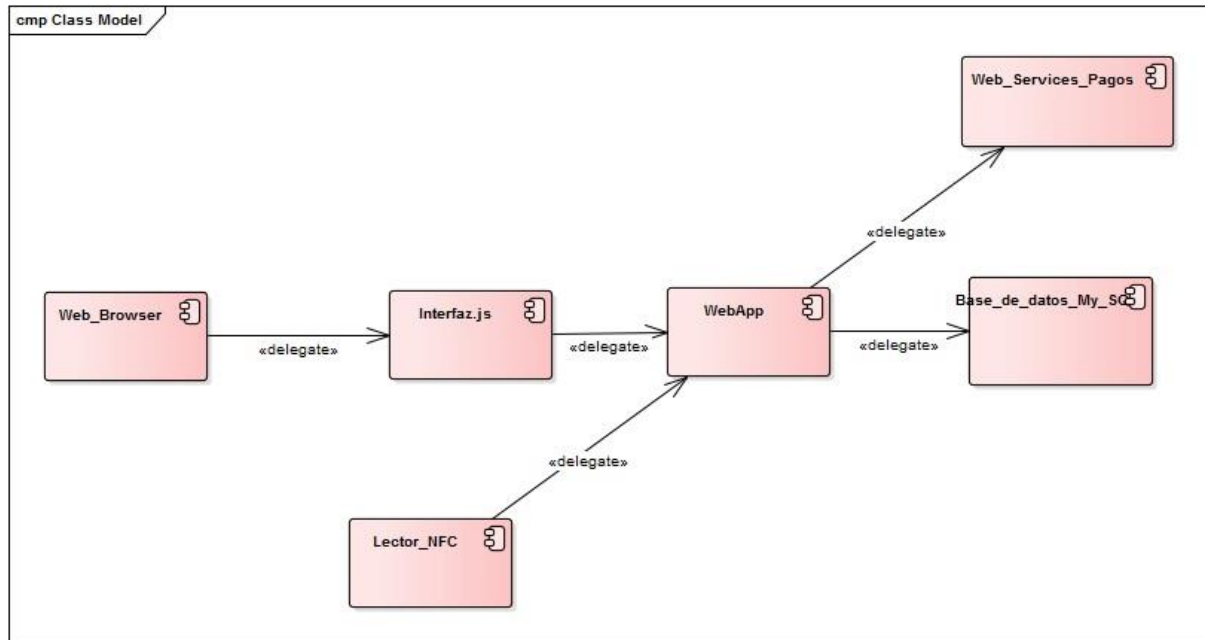
[This section describes one or more physical network (hardware) configurations on which the software is deployed and run. It is a view of the Deployment Model. At a minimum for each configuration it should indicate the physical nodes (computers, CPUs) that execute the software and their interconnections (bus, LAN, point-to-point, and so on.) Also include a mapping of the processes of the **Process View** onto the physical nodes.]

8. Vista de Implementación

Se muestra la visión del sistema desde el punto de vista de los componentes que integran al mismo, cómo se conectan y se interrelacionan para poder permitir su correcta funcionalidad.

Se muestra a continuación el diagrama de componentes correspondiente.

Diagrama de componentes:



8.1 Visión General

No aplica

[This subsection names and defines the various layers and their contents, the rules that govern the inclusion to a given layer, and the boundaries between layers. Include a component diagram that shows the relations between layers.]

8.2 Capas

No aplica

[For each layer, include a subsection with its name, an enumeration of the subsystems located in the layer, and a component diagram.]

9. Vista de Datos (opcional)

No aplica

[A description of the persistent data storage perspective of the system. This section is optional if there is little or no persistent data, or the translation between the Design Model and the Data Model is trivial.]

10. Tamaños y Performance

- El sistema debe soportar un mínimo de 320000 usuarios conectados simultáneamente. (ya que esta es la sumatoria de la estimación máxima de usuarios permitidos dentro de cada parque)
- El tiempo de respuesta de la consulta que provee el dispositivo móvil no debe ser mayor de 3 segundos.

- El tiempo para realizar una transacción o una reserva para una atracción no debe ser mayor a 7 segundos.
- La navegación por medio del Browser debe ser intuitiva permitiendo que en 3 clicks o menos, teniendo como punto de partida la página de inicio, el cliente pueda realizar la compra de cualquier producto o realizar una reserva.

11. Calidad

Para un mejor aprovechamiento de la arquitectura de software se dan los siguientes requerimientos de calidad:

1. Usabilidad
 - a. El sistema permitirá un manejo intuitivo por parte de los usuarios.
 - b. El sistema permitirá el acceso concurrente de varios usuarios, de modo que puedan realizar transacciones simultáneas.
2. Eficiencia
 - a. El programa no demorara más de 3 segundos en realizar la búsqueda de cualquier parque y/o atracción para la reserva de turno o compra de tickets.
3. Seguridad
 - a. El sistema permitirá el acceso a funcionalidades dependiendo del perfil del usuario que ingresa al sistema, validando su ingreso a través de un login brindando el usuario y la clave correspondiente.
 - b. Usar criptografía y protocolos de alta seguridad Secure Sockets Layer (SSL)/Transport Layer Security (TLS) e Internet Protocol Security (IPSEC) para salvaguardar los datos confidenciales de los tarjetahabientes durante su transmisión.
4. Confiabilidad
 - a. El sistema tendrá en cuenta que la información ingresada en él sea válida, para lo cual mostrará mensajes que expliquen al usuario acerca de los errores que éste pudiera cometer y de aquellos que pueda cometer el mismo sistema.
5. Mantenimiento
 - a. El sistema será flexible, facilitando su mantenimiento futuro.