**ANÁLISIS PROPERTY MANAGAMENT SYSTEM**

**Documento de Diseño**

**GRUPO 1**

Juan Camilo Ibáñez Nariño - 201924835

Johan Alexis Bautista Quinayas - 202214789

Pablo Andrés Restrepo Rodríguez - 202011172

**Universidad de los Andes**

**Diseño y programación orientado a objetos**

**Rubén Manrique**

**15 de marzo de 2023**

1. **Contexto y Alcance:**

Siguiendo con la línea del análisis, se identificaron ciertos usuarios (administrador, recepcionista y empleado) para los cuales la aplicación, y por ende el diseño para llegar a dicha implementación, debe servir. El problema, entonces, se divide en las interacciones entre usuario y aplicación y los resultados a los que se quiere llegar.

En la aplicación Property Management System se requiere almacenar y actualizar la información completa de un hotel. Esto incluye la información sobre las habitaciones, tarifas, huéspedes y servicios que pueda llegar a incluir dicho hotel. El sistema, entre otras funcionalidades, también debe:

* permitir el registro o cancelación de reserva
* retornar pagos a nombre de uno o varios huéspedes por habitación en forma de facturas
* avisar (y en algunos casos corregir) sobre posibles errores en la carga de datos
* visualizar inventarios mediante una interfaz gráfica
* bloquear o desbloquear habitaciones según un rango de fechas

Dados los requerimientos generales anteriormente presentados y las restricciones enumeradas en el análisis, este documento se encargará de desglosar el problema en subcategorías más pequeñas hasta llegar a una asignación específica de responsabilidades dentro del programa.

1. **Objetivos:**

* Permitir una visualización clara del problema.
* Reducir la complejidad del diseño al implementar estructuras claras y un desglosamiento a fondo del problema.
* Asignar responsabilidades, colaboraciones y notas a componentes dentro del funcionamiento de la aplicación.
* Complementar y/o corregir el análisis del problema construyendo diagramas de clase y de secuencia.
* Agilizar el proceso de implementación (a futuro).

1. **Solución de Diseño:**

**Nivel 1. Componentes candidatos y estereotipos**

1. Al reconocer la necesidad de almacenar, permitir el acceso de la extensa cantidad de datos que se necesita para que el sistema opere correctamente, es evidente que se necesita un componente que se encargue de tal responsabilidad. Dicho componente, haciendo énfasis en el estereotipo Information holder, se denomina “Base de datos”. Adicionalmente, también se presenta como un estructurador, puesto que no tendrá una lógica compleja y tendrá un rol para organizar los archivos y permitir que se acceda a ellos
2. Siguiendo con la necesidad de modificar y constante mantenimiento de la información para un correcto desarrollo del sistema, se evidencia la necesidad de un controlador de la información, dejando de lado el almacenamiento de la base de datos, este se enfoca en el rol del administrador para la carga de archivos y el correcto procesamiento de los mismos.

**Descomposición de requerimientos funcionales e identificación de responsabilidades:**

De acuerdo a los requerimientos funcionales encontrados en el análisis se realiza un desglosamiento de los mismos en responsabilidades.

|  |  |
| --- | --- |
| **Requerimiento funcional** | **Responsabilidades asociadas** |
| Registrar pago | Buscar el huésped |
| Buscar el consumo |
| Eliminar el consumo |

|  |  |
| --- | --- |
| **Requerimiento funcional** | **Responsabilidades asociadas** |
| Hacer una reserva | Verificar habitaciones disponibles |
| Crear un objeto reservación |
| Agregar reservación a la lista de reservas |
| Bloquear las habitaciones asociadas |

|  |  |
| --- | --- |
| **Requerimiento funcional** | **Responsabilidades asociadas** |
| Registrar consumo de un huésped | Buscar el huésped |
| Crear el consumo |
| Asociar el consumo al huésped |

|  |  |
| --- | --- |
| **Requerimiento funcional** | **Responsabilidades asociadas** |
| Cancelar una reserva | Buscar la reservación |
| Verificar fecha de anticipación para cancelar |
| Eliminar la reservación |

|  |  |
| --- | --- |
| **Requerimiento funcional** | **Responsabilidades asociadas** |
| Generar facturas | Buscar el huésped |
| Acceder a las facturas asociadas al huésped |

|  |  |  |  |
| --- | --- | --- | --- |
| **Requerimiento funcional** | **Desglosamiento** | **Desglosamiento** | **Responsabilidades asociadas** |
| Cargar y/o modificar inventario de habitaciones | Cargar inventario de habitaciones | Cargar archivo de habitaciones | Subir archivo de habitaciones |
| Procesar archivo de habitaciones |
| Actualizar inventario habitaciones | Eliminar habitación |
| Crear habitación |
| Modificar inventario de habitaciones | Modificar información de la habitación | Buscar habitación |
| Modificar capacidad |
| Cambiar tipo habitación |
| Cambiar balcón |
| Cambiar vista |
| Cambiar Cocina |
| Cambiar camas |

|  |  |  |  |
| --- | --- | --- | --- |
| **Requerimiento funcional** | **Desglosamiento** | **Desglosamiento** | **Responsabilidades asociadas** |
| Cargar y/o modificar tarifas por tipo de cuarto | Cargar tarifas por tipo de cuarto | Cargar archivo de tarifas por tipo de cuarto | Subir archivo de tarifas por tipo de cuarto |
| Procesar archivo de tarifas por tipo de cuarto |
| Modificar tarifas por tipo de cuarto | Actualizar tarifas por tipo de cuarto | Modificar tarifas por tipo de cuarto |

|  |  |  |  |
| --- | --- | --- | --- |
| **Requerimiento funcional** | **Desglosamiento** | **Desglosamiento** | **Responsabilidades asociadas** |
| Cargar y/o modificar información de servicios | Cargar información de servicios | Cargar archivo de tarifas por tipo de cuarto | Subir archivo de información de servicios |
| Procesar archivo de información de servicios |
| Modificar información de servicios | Modificar información de servicios | Cambiar precio servicio |
| Cambiar nombre del servicio |

|  |  |
| --- | --- |
| **Requerimiento funcional** | **Responsabilidades asociadas** |
| Ver información del inventario de habitaciones | Desplegar total de habitaciones |
| Ver disponibilidad de habitación |
| Ver huésped actual |

|  |  |
| --- | --- |
| **Requerimiento funcional** | **Responsabilidades asociadas** |
| Registro (check in) | Instanciar objeto huésped |
| Instanciar grupo de huéspedes |
| Asociar grupo de huéspedes al huésped principal |
| Asociar huésped a las habitaciones correspondientes |

**Asignación de responsabilidades:**

|  |  |  |
| --- | --- | --- |
| **#** | **Responsabilidad** | **Componente** |
| **1** | Buscar el huésped | Base de datos |
| **2** | Buscar un consumo | Huésped |
| **3** | Eliminar el consumo | Huésped |
| **4** | Verificar habitaciones disponibles | Base de datos |
| **5** | Crear un objeto reservación | Reservación |
| **6** | Agregar reservación a la lista de reservas | Base de datos |
| **7** | Bloquear la habitación | Habitación |
| **8** | Crear objeto consumo | Consumo |
| **9** | Asociar el consumo al huésped | Huésped |
| **10** | Buscar la reservación | Base de datos |
| **11** | Verificar fecha de anticipación para cancelar | Reservación |
| **12** | Eliminar la reservación | Base de datos |
| **13** | Acceder a las facturas asociadas al huésped | Huésped |
| **14** | Subir archivo de habitaciones | Administrador |
| **15** | Procesar archivo de habitaciones | Base de datos |
| **16** | Eliminar habitación | Base de datos |
| **17** | Crear habitación | Habitación |
| **18** | Buscar habitación | Base de datos |
| **19** | Modificar capacidad | Habitación |
| **20** | Cambiar tipo habitación | Habitación |
| **21** | Cambiar balcón | Habitación |
| **22** | Cambiar vista | Habitación |
| **23** | Cambiar Cocina | Habitación |
| **24** | Cambiar camas | Habitación |
| **25** | Subir archivo de tarifas por tipo de cuarto | Administrador |
| **26** | Procesar archivo de tarifas por tipo de cuarto | Base de datos |
| **27** | Modificar tarifas por tipo de cuarto | Habitación |
| **28** | Subir archivo de información de servicios | Administrador |
| **29** | Procesar archivo de información de servicios | Base de datos |
| **30** | Cambiar precio servicio | Servicio |
| **31** | Cambiar nombre del servicio | Servicio |
| **32** | Desplegar total de habitaciones | Base de datos |
| **33** | Ver disponibilidad de habitación | Habitación |
| **34** | Ver huésped actual | Habitación |
| **35** | Instanciar objeto huésped | Huésped |
| **36** | Instanciar grupo de huéspedes | Grupo de huéspedes |
| **37** | Asociar grupo de huéspedes al huésped principal | Huésped |
| **38** | Asociar huésped a las habitaciones correspondientes | Habitación |
| Otras responsabilidades encontradas | | |
| **39** | CalcularCapacidad | Habitación |
| **40** | Modificar menús de restaurante | Administrador |

**Colaboraciones:**

Ahora bien, a partir de las responsabilidades definidas y asignadas a los diferentes componentes, se implementan colaboraciones como un conjunto de responsabilidades de diferentes componentes con el fin de cumplir con requerimientos funcionales.

**Hacer una reserva.**

Implica que la recepción desde la clase recepcionista, después de indicar por medio de la interfaz el llamado al método, se continúa con el cumplimiento de las siguientes responsabilidades:

**Responsabilidades asociadas:**

1. Le indica a la base de datos verificar las habitaciones disponibles
2. Le indica a la clase reserva, implementar constructor para instanciar un objeto reservación
3. Le indica a la base de datos agregar la reservación a la lista de reservas
4. Finalmente le indica a la clase habitación, en sus instancias correspondientes bloquear las habitaciones asociadas a la reservación.

**Cargar y/o modificar inventario de habitaciones**

Implica que, desde el controlador de la información, desde la clase administrador, después de indicar por medio de la interfaz el llamado al método, se continúa con el cumplimiento de las siguientes responsabilidades:

**Responsabilidades asociadas:**

1. Se llama a subir archivo de habitaciones por parte del administrador
2. Se procesa y desglosa el archivo de habitaciones desde la base de datos
3. Se eliminan las habitaciones a sobrescribir desde la clase hotel
4. Se le indica a la clase habitación hacer uso de constructor para instanciar habitaciones

Seguidamente se enumeran las responsabilidades opcionales según el rol de administrador disponga:

1. Buscar habitación
2. Modificar capacidad
3. Cambiar tipo habitación
4. Cambiar balcón
5. Cambiar vista
6. Cambiar Cocina
7. Cambiar camas

**Ver información del inventario de habitaciones:**

Después de que recepcionista indique el llamado al método aplican las siguientes responsabilidades de los diferentes componentes, en este caso, base de datos y habitación.

**Responsabilidades asociadas:**

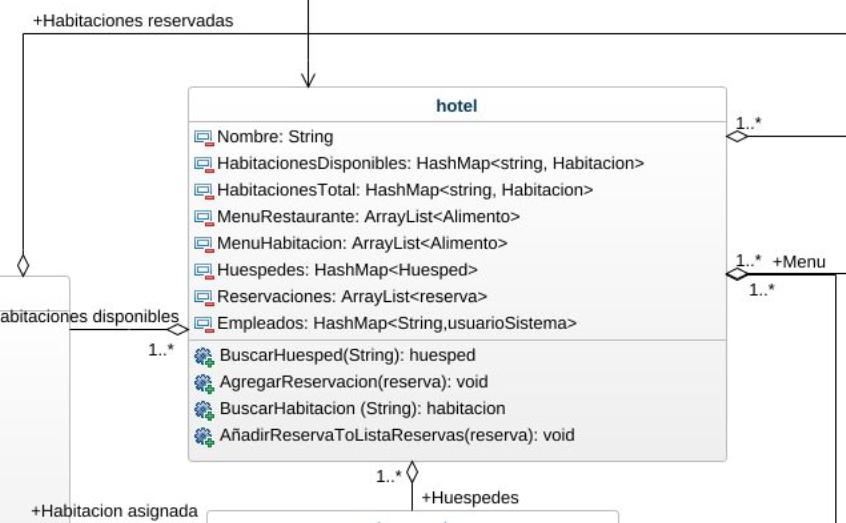
1. Inicialmente se le indica a la base de datos desplegar total de habitaciones
2. Se le indica a la habitación ver disponibilidad de habitación
3. Finalmente, se le indica a la habitación permitir ver huésped actual

**Nivel 2.**

**Base de datos.**

En esta sección vale la pena recordar las responsabilidades asociadas a este componente:

* Buscar un huésped
* Agregar reservación a las reservaciones del hotel
* Buscar una habitación del inventario

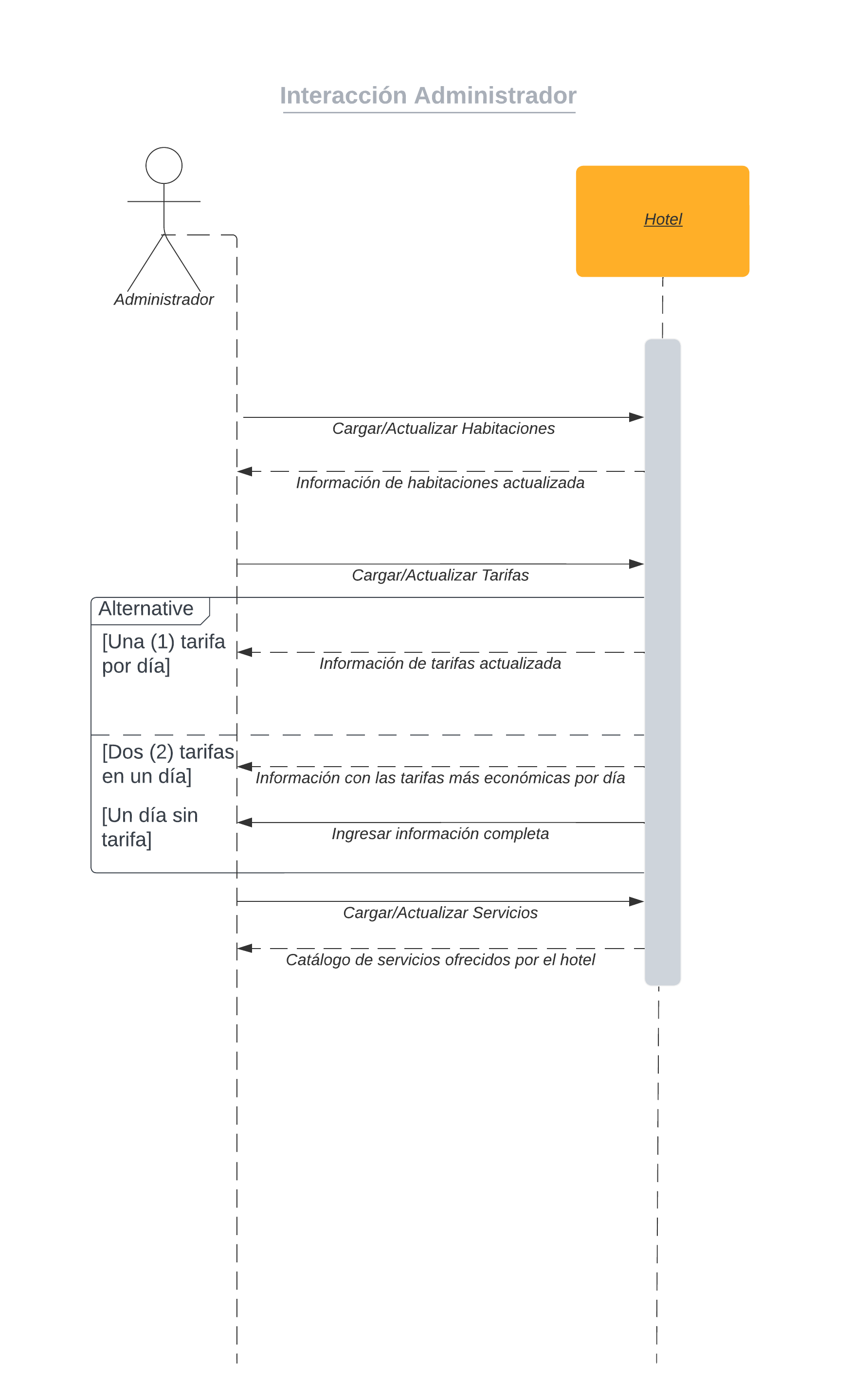


**Controlador de la información**

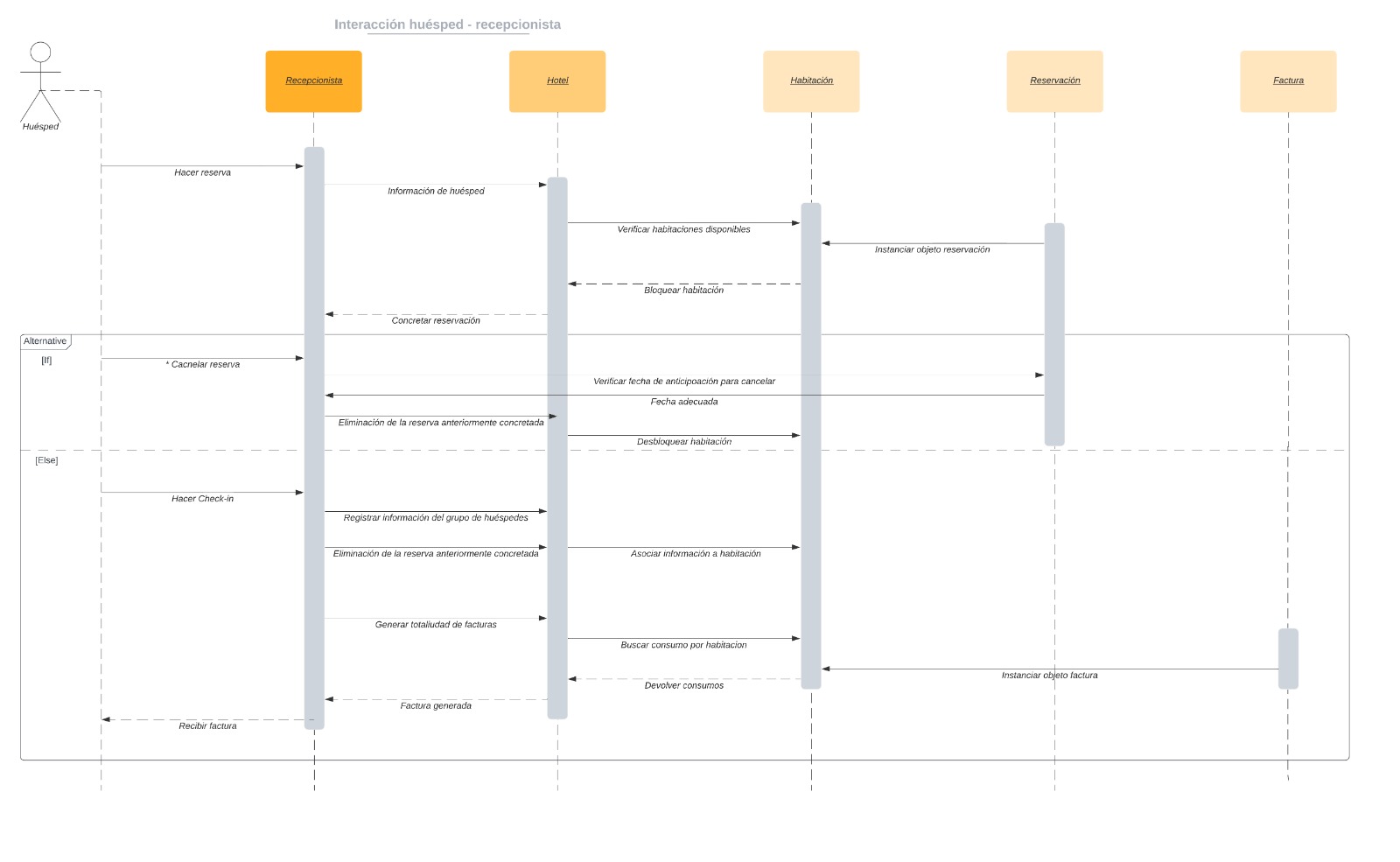
En esta sección vale la pena recordar las responsabilidades asociadas a este componente:

* Subir archivo de habitaciones
* Procesar archivo de habitaciones
* Subir archivo de tarifas por tipo de cuarto
* Procesar archivo de tarifas por tipo de cuarto
* Subir archivo de tarifas por tipo de cuarto
* Procesar archivo de tarifas por tipo de cuarto

Diagrama de Secuencia:



A continuación, interacción desde Huésped:



1. Diagrama de Clases Completo:

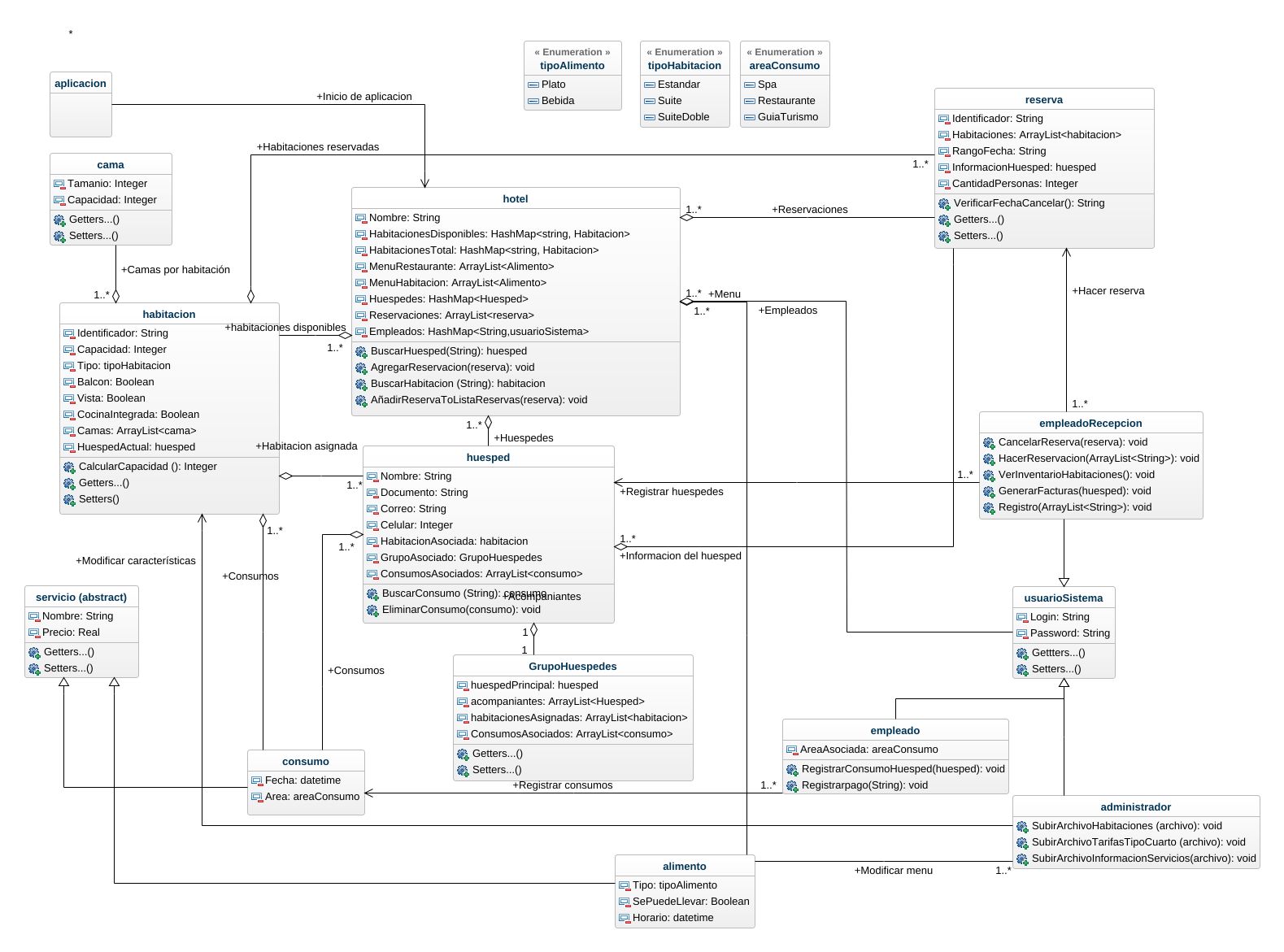
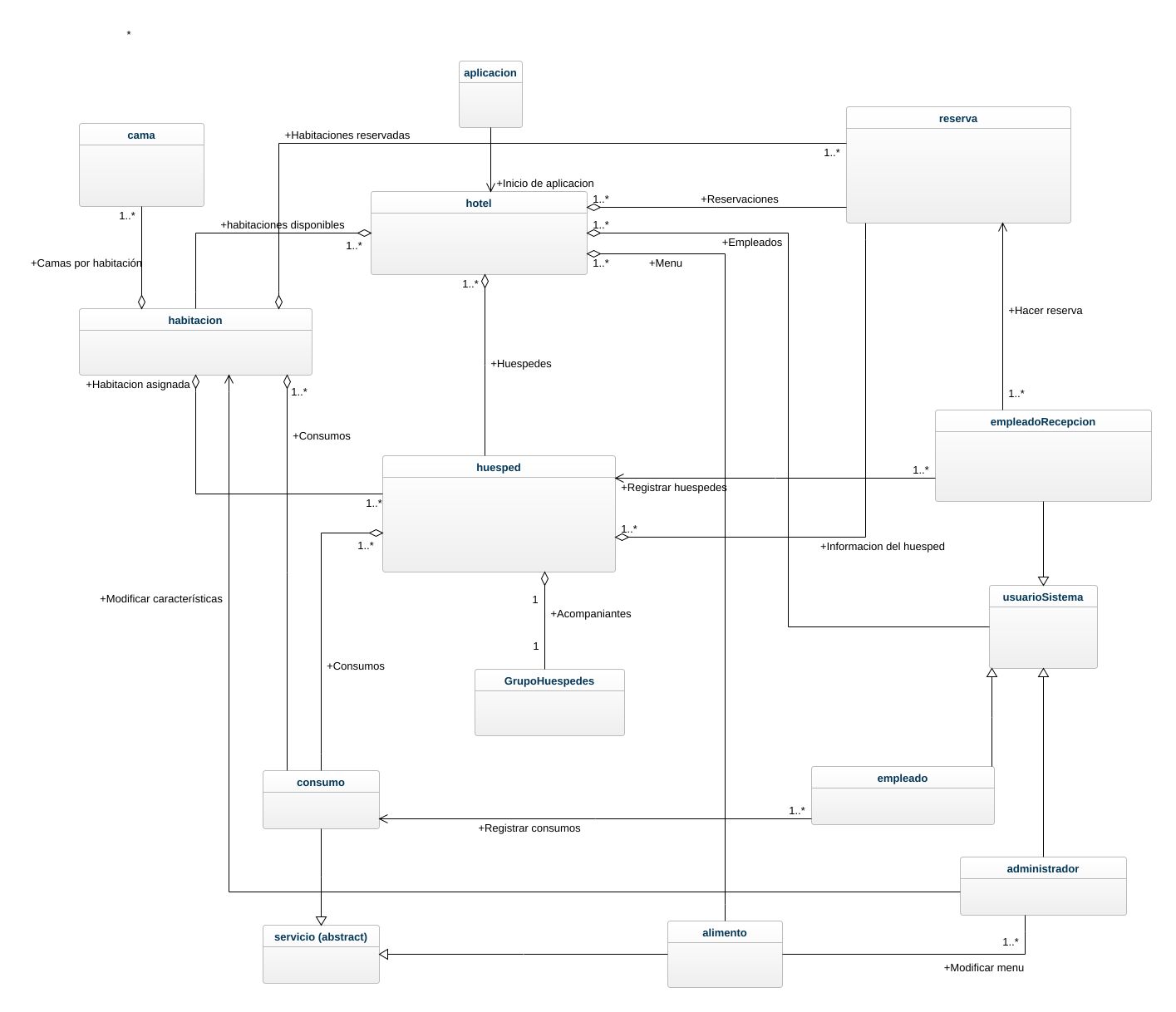


Diagrama de Clases Resumido:



**Notas especiales sobre los diagramas de clases:**

* Las flechas representan composiciones, que fueron modelados correctamente en la herramienta, sin embargo, al momento de exportar presenta estas modificaciones no intencionales, por favor tomarlas como composiciones.

1. **Consideraciones Adicionales:**

Gestión de fechas:

* Para bloquear habitaciones en ciertas fechas, se emplea base de datos que almacene la disponibilidad de habitaciones en diferentes fechas. Luego, cuando un usuario intenta reservar una habitación en una fecha determinada, el programa verificará si la habitación está disponible y permitirá o denegará la reserva en consecuencia.

Para cambiar promociones por fechas específicas, se emplea base de datos que almacena información sobre las promociones actuales y sus fechas de validez.