

Grupo 2
Maria Fernanda de la Hoz
Valeria Marin
Natalí Bohórquez

Contexto y Alcance

El propósito del sistema de PMS descrito en este documento es facilitar el manejo de las operaciones de un hotel. El diagrama y diseño del sistema se construyeron pensando en una solución que simplifique las relaciones que debe haber entre las entidades responsables de las diferentes acciones dentro de este.

En esta primera iteración del diseño los usuarios del sistema son el administrador, los empleados del hotel que pueden facturar consumo y los recepcionistas que se encargan de registrar a los huéspedes, crear y cancelar reservas. Estos roles interactúan con la información del stock del hotel, que incluye un listado de habitaciones con su información, otro de tarifas donde se pueden actualizar y cambiar las tarifas de las habitaciones, la información de los menús de cada restaurante en el hotel y el catálogo de servicios a los cuales los huéspedes pueden acceder durante su estadía. Por último, la facturación de estos bienes y servicios que se pueden adquirir es manejada por el sistema, que genera facturas para que se puedan almacenar en las bases de datos del hotel que necesita mantener.

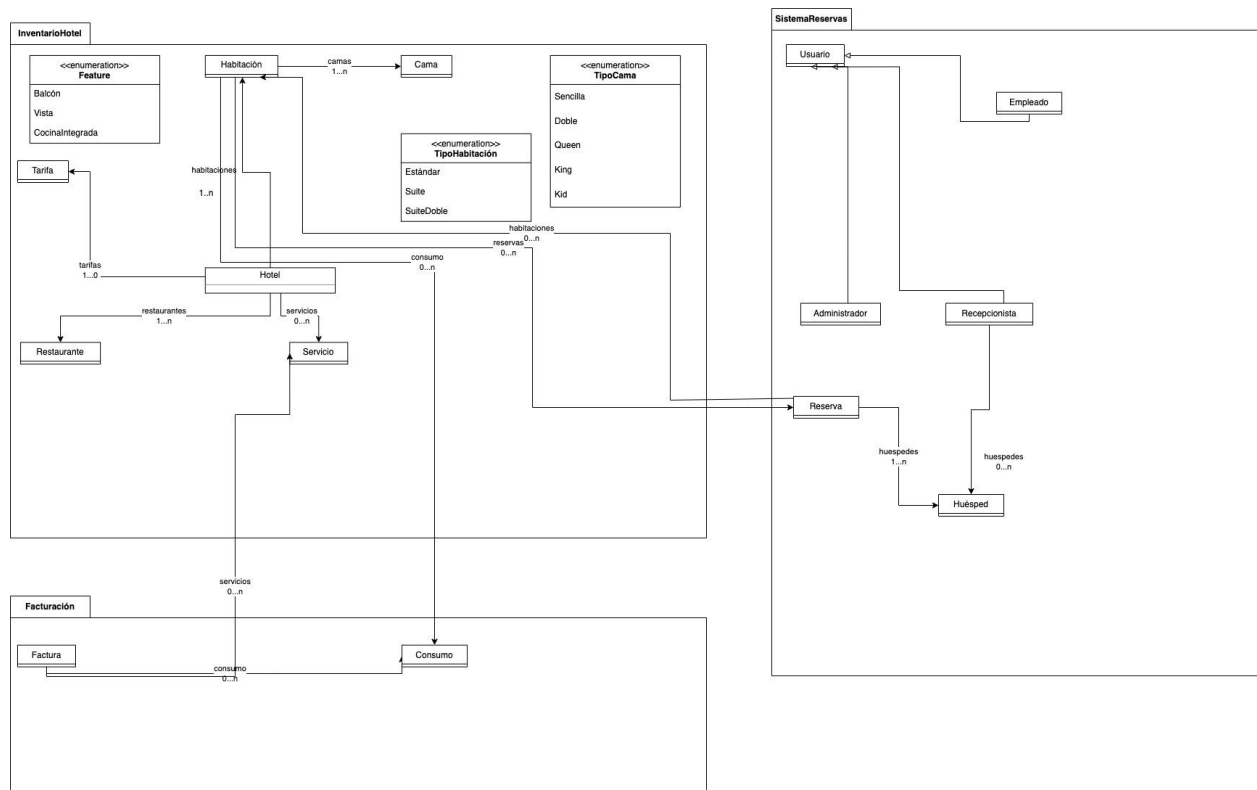
Diagrama de secuencia reservaciones

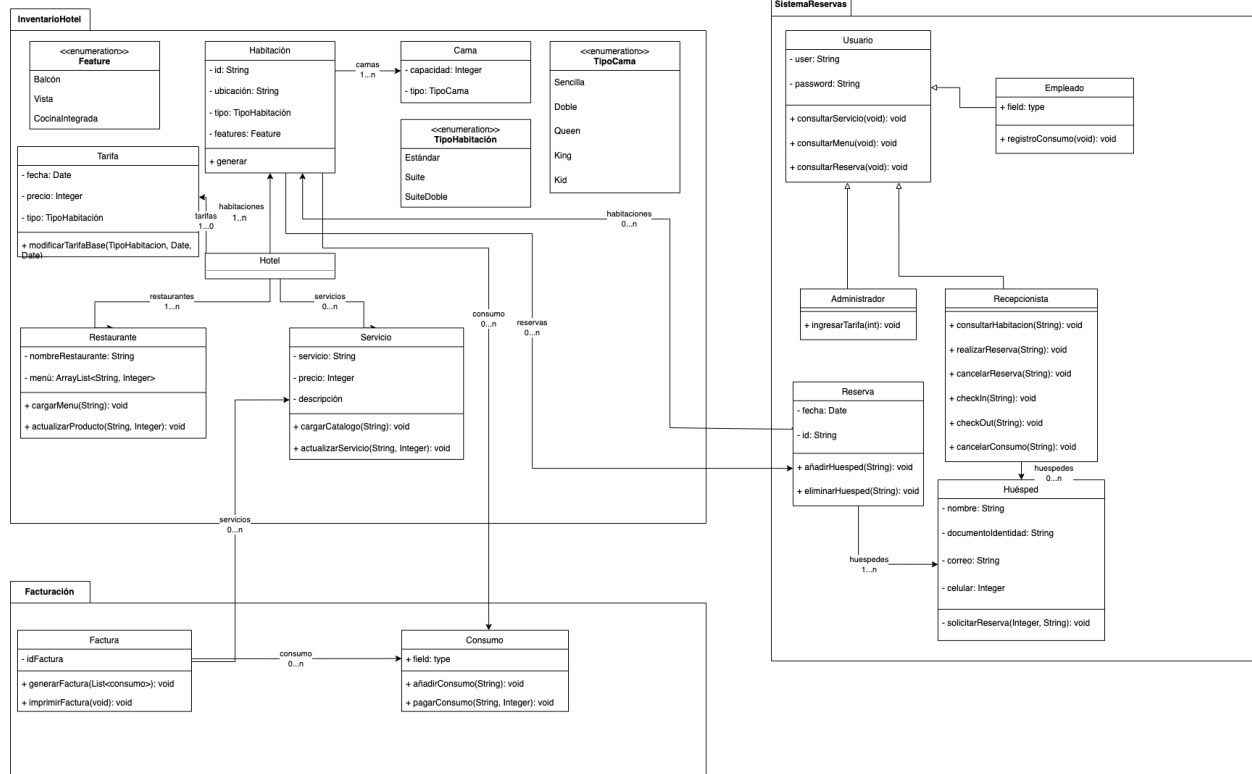
Usuario llama a la recepción ->	Recepcionista pide detalles y ofrece información de habitaciones ->	Se realiza la reserva/ se cancela la reserva ->	Se modifica el atributo de reserva dentro de la clase Habitación ->	 fin
--	--	--	--	---------------------

Diagrama de clases

- Se supone que todas las clases tendrán getters y setters, no se tienen en cuenta en la representación gráfica del diagrama para no congestionar el diseño de la solución.

- Se hace entrega del diagrama simplificado en este documento de diseño y en el github de la entrega está el diagrama completo por razones de facilidad de visualización.





Supuestos

InventarioHotel

- Los precios se modelan como enteros ya que se pensó el sistema para manejar precios en \$ COP.
- Las habitaciones tienen una capacidad fija de adultos y otra de niños dada por la cantidad de camas que tenga la habitación.
- Solo existen 5 tipos diferentes de camas, esta información se modelará como un enumeration dentro del sistema. Cada cama tiene una capacidad establecida.
- Las camas de adulto no pueden ser utilizadas por niños. La única excepción es en el caso de los menores de 2 años como indican los requerimientos del enunciado y sólo puede haber 1 menor por cama en todo caso.
- Los tipos de habitaciones estarán modelados como una enumeration.
- Las tarifas se almacenarán en un HashMap con llave fecha (tipo Date en Java) y valor HashMap que a su vez tendrá como llave el tipo de la habitación y como valor un entero que representa el precio de esta. Las fechas incluyen todo el calendario de mediano plazo de un hotel (24 meses) que se usa para la planeación de tarifas.

- Existe una tarifa base para cada tipo de habitación. (Importante: Este valor por default no interfiere con la condición requerida por el enunciado en donde especifican que si el Manager entra a fijar una tarifa más alta para una habitación que ya tiene tarifa, se escogerá la menor) El programa será capaz de diferenciar entre esta tarifa base y las tarifas posteriores que quiera modificar el Manager.
- Los add-ons o Features de las habitaciones se suman a la tarifa de una habitación como porcentaje de la tarifa. Para esto se implementará dentro del enum no solo el feature que se puede añadir al costo de la habitación sino también el porcentaje de la tarifa final que el feature representa para la habitación.
- La clase hotel recopila la información de su stock, infraestructura, precios y servicios.
- El menú se implementará como un HashMap, en el que se incluyen, los nombres de los platos, precios, horas en las que se encuentra disponible un plato y sí es posible consumirlo en la habitación.
- Es importante que se tenga en cuenta que el inventario se va a encontrar en una carpeta diferente al código fuente de la aplicación.

SistemaReservas

- La clase Empleado y Administrador heredan de la clase padre Usuario los atributos y funciones requeridas por todos para acceder al sistema.
- La clase Recepcionista hereda del empleado debido a que requiere las funciones de hacer registro de servicios pero además tiene la responsabilidad de crear las reservas de los huéspedes.
- Los huéspedes tienen un atributo adicional a los requeridos por el enunciado: edad. Este dato es importante para poder registrar a los niños y asignar la capacidad de la habitación.
- Cada reserva tiene un atributo habitación que permite la consulta rápida de la información detallada de esta.

Facturación

- El consumo de cada habitación queda identificado durante la estadía del/los huésped/es dentro de la clase Habitación solo durante el tiempo de su estadía. Para recibir a un nuevo huésped se borra el consumo y se inicia en blanco un nuevo registro. Esta funcionalidad es parte del check-out.
- Sin embargo, todo el consumo queda almacenado en formato de facturas dentro de una estructura de datos dentro del hotel debido a la importancia de este para la contabilidad.