**Diseño**

Juan Esteban Álvarez

Santiago Tinjacá

Juan David Roa Moyano

**Estilo de control**

Centralizado

|  |  |
| --- | --- |
| **Clase** | **Estereotipo** |
| Aplicación | Interfacer |
| Hotel | Controller |
| Usuario | Structurer |
| Admin | Coordinator |
| Empleado | Coordinator |
| Persistencia | Service Provider |
| Grupo | Service Provider |
| Reserva | Service Provider |
| Restaurante | Service Provider |
| Tarifa | Information Holder |
| Huesped | Information Holder |
| Servicio | Information Holder |
| ProductoMenu | Information Holder |
| Habitación | Information Holder |
| TipoHabitacion | Information Holder |

**Justificaciones:**

El Hotel puede realizar todas las acciones, sin embargo, las llamadas de las acciones solo las pueden hacer los usuarios. Según el tipo de usuario (Admin o empleado) se pueden o no llamar métodos específicos.

La interfaz de la aplicación solo muestra las opciones permitidas para un tipo de usuario en específico (admin o empleado).

Los servicios se cobran al Grupo y no a la Habitación. Al solicitar un servicio se pide el id de la habitación y con este se puede obtener el id del Grupo gracias al TreeMap Ocupados del Hotel.

En el menú del restaurante se encuentran tanto los platos como las bebidas.

Al realizar el uso de un servicio se tiene la posibilidad de hacer el pago luego de la estadía o al mismo instante de usarlo. En caso de que la persona no pague al instante se guarda el total en un Saldo el cual se verá reflejado en la factura final. De lo contrario no suma, pero si aparece en la factura.

Para calcular la tarifa de una Reserva se consulta un treemap cuyas llaves son los siguientes 365 días a partir de la fecha actual y sus valores son objetos de tipo Tarifa. El precio de la tarifa depende del tipo de Habitación y de la fecha de la Reserva.

En caso de que el admin se equivoque y coloque una tarifa más costosa a una habitación que ya tiene una tarifa asignada, entonces se deja la tarifa más económica. Si coloca una tarifa más económica de la que ya hay asignada se asignará la tarifa más económica. Si se equivoca y pone una tarifa más económica de la deseada y quiere cambiarla por una más costosa, deberá digitar el opuesto del número (esto permitirá asignar intencionalmente tarifas que en principio no serían posibles, a la vez que se evitan los errores que pueda cometer el admin)

Hotel es un atributo estático de los usuarios porque de esta forma todos los usuarios manejan el mismo Hotel y no se crean copias de él. Asi cuando un usuario modifica algo del hotel se modifica también para el resto de los usuarios.

TipoHabitacion es un enum ya que solo existen tres tipos de habitación (estándar, suite, suite double).

Tanto el Hotel como la(s) habitación(es) tiene listas de Servicios, pero cada una tiene sus propios Servicios (son excluyentes).

Para verificar su una habitación se puede reservar se consulta si está en el treemap de Ocupados.

El precio de la reserva se obtiene al momento de reservar y no cambia en caso de que un admin modifique la tarifa para esas fechas.

**Reglas de dominio**

* Las habitaciones se bloquean luego de ser reservadas
* No todos los platos pueden llevarse a la habitación
* La comida no está disponible a todas las horas
* Todos los huespedes del grupo hacen check-in y check-out al mismo tiempo
* La primera posicion de la lista de huespedes corresponde al lider de grupo
* El precio de la reserva se obtiene al momento de reservar y no cambia
* Las reservas solo se pueden cancelar hasta 48 horas antes de la fecha establecida
* Misma tarifa en fechas se asigna la tarifa más económica
* El hotel tiene un treemap donde las llaves son la fecha y el valor es un objeto tarifa
* La persistencia es serializada. Los objetos que se guardaran: TreeSet(Tarifas), HashMap(Reserva), HashMap(Usuario) HashMap(Habitacion)

**Diagramas de clases UML**

* Nivel inicial de descomposición

Diagrama, Esquemático

Descripción generada automáticamente

* Diagrama Diseño UML de alto nivel (nivel final): Objetos/Elementos, relaciones y cardinalidades

Diagrama, Esquemático

Descripción generada automáticamente

* Diagrama Diseño UML detallado (nivel final): Objetos/Elementos, atributos, métodos, relaciones y cardinalidades

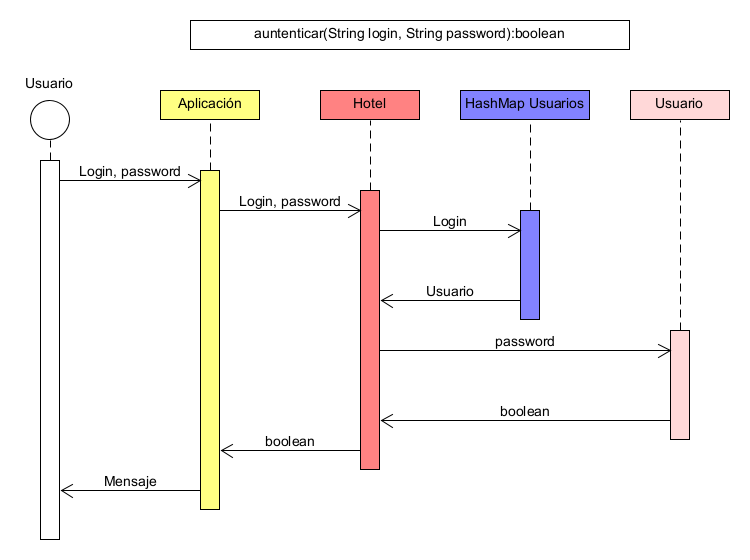
**Diagrama

Descripción generada automáticamente**

**Diagrama de secuencia:**

Requerimientos críticos

* Autenticar un usuario



* Cancelar una reserva

Diagrama

Descripción generada automáticamente

* Consultar menú restaurante

**Diagrama

Descripción generada automáticamente**