

CONTENIDO

ANALISIS

1. Descripción del Problema:

Se solicita desarrollar un programa para una cadena de hoteles, para una mejor y más accesible administración. El programa debe constar de un entorno agradable y de fácil manejo para el usuario. Entre sus funciones principales esta la reservación de habitaciones, control de la entrada y salida de huéspedes así como el uso de los distintos servicios que el hotel presta.

2. Requerimientos del Usuario:

2.1. Medio Físico

El usuario deberá contar con una computadora o dispositivo electrónico en el cual ejecutar el programa.

2.2. Una Plataforma

Se necesita tener instalada una plataforma virtual que utilice el lenguaje Java en la computadora para poder ejecutar con mayor facilidad el programa.

2.3. Interfaz Gráfica

Es la forma en que el usuario se comunicara y manejara el programa.

2.3.1. Menú

El programa realizará varias acciones para la administración del hotel, por lo tanto el menú será el medio por el cual se administraran dichas acciones.

3. Planteo Inicial de la Solución

El programa necesita principalmente de la interfaz gráfica, que será de donde se desglosaran todas sus funciones. Las primeras acciones a programar será la

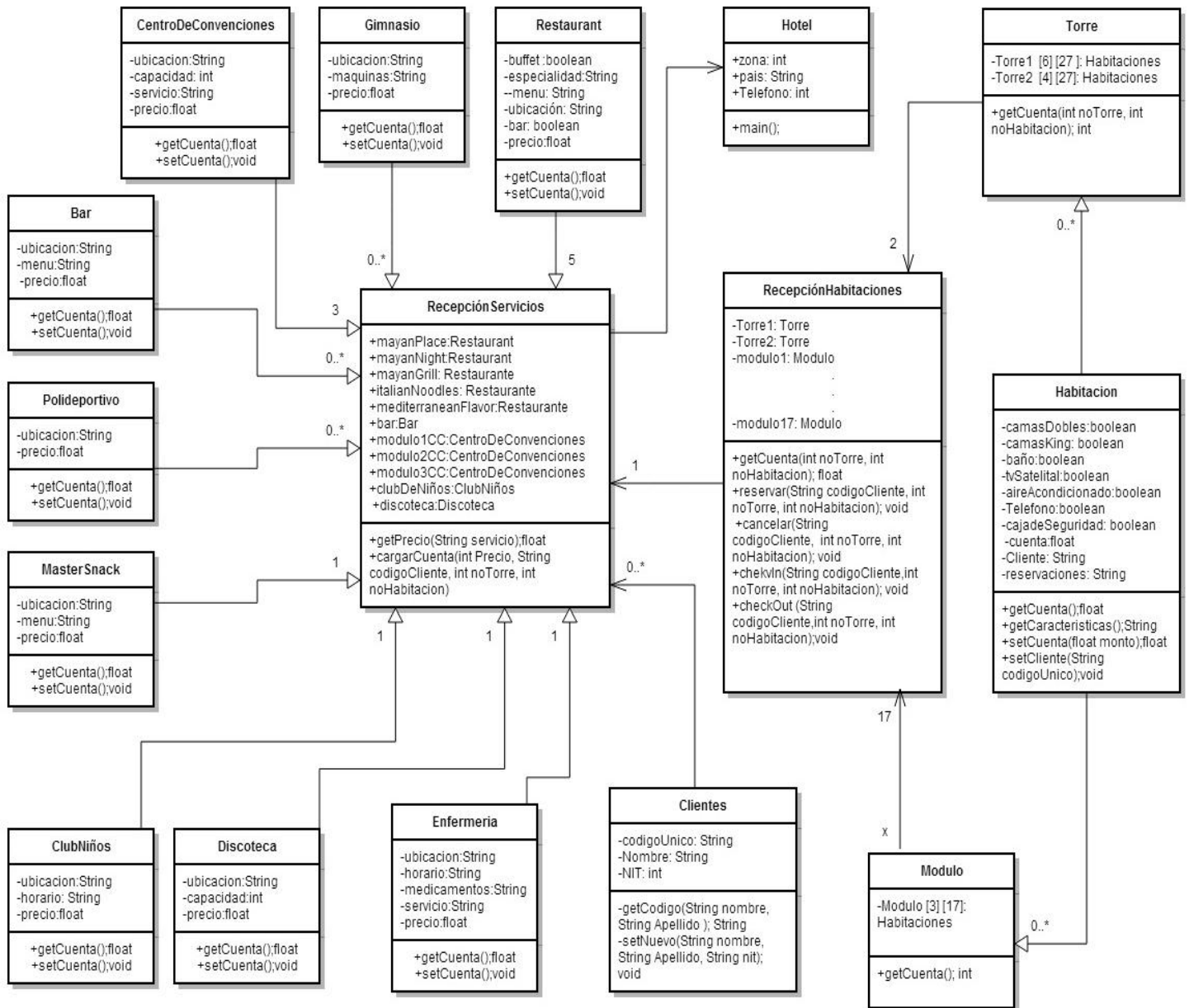
asignación de una habitación a un cliente, la reservación de habitaciones, y el control de chequeos de entrada/ salida. Posteriormente se programará los servicios que el hotel ofrece. Por último, la generación de reportes de todas las actividades del hotel. Anterior a empezar con el código, se necesitan diagramas de clase para dar una idea o un bosquejo de cómo se formara el programa, estos bosquejos fueron diseñados a mano y luego a computadora y se muestran más adelante en este trabajo.

4. Ciclo de Vida del Software

El software tiene un ciclo de vida que empieza desde el requisito de dicho programa, este se tiene que actualizar hasta que los servicios del hotel cambien.

DISEÑO

1. Diagrama de Clases



2. Diccionario de Clases

- 2.1. Hotel: Es la clase main del programa, en ella se instanciará la clase RecepciónServicios.
- 2.2. RecepcionServicios: como su nombre lo indica contiene todos los servicios que el hotel ofrece instanciados, dependiendo de la cantidad que tiene el hotel.
 - 2.2.1. getPrecio(String servicio): es un método que devuelve la cantidad gastada o a deber. Necesita como parámetro el servicio (en letras) que el cliente utilizó.
 - 2.2.2. cargarCuenta(int Precio, String codigoCliente, int noTorre, int noHabitacion): es un método que no devuelve nada, pero le agrega el precio a la cuenta de la habitación deseada. Como parámetro se necesita el monto, el cliente (su código único) y la habitación.
- 2.3. RecepciónHabitaciones: es una clase que controla la asignación de las habitaciones, la salida y entrada de los huéspedes.
 - 2.3.1. checkOut (String codigoCliente,int noTorre, int noHabitacion):es un método que desvincula el cliente con la habitación.
 - 2.3.2. chekvIn(String codigoCliente,int noTorre, int noHabitacion): es un método que vincula el cliente con la habitación.
 - 2.3.3. cancelar (String codigoCliente, int noTorre, int noHabitacion): es un método que cancela las reservaciones. Desvincula los clientes con las reservaciones de las habitaciones.
 - 2.3.4. reservar (String codigoCliente, int noTorre, int noHabitacion): es un método que vincula un cliente a la cola de la reservaciones de cada habitación.
 - 2.3.5. getCuenta(int noTorre, int noHabitacion): devuelve el monto que tiene cargado la habitación.
- 2.4. Clientes: Es una clase de los clientes que han visitado el hotel.

- 2.4.1. setNuevo(String nombre, String Apellido, String nit): es un método que carga a la base de datos un nuevo cliente.
- 2.4.2. getCodigo(String nombre, String Apellido, int Nit): es un método que devuelve el código único del cliente utilizando los parámetros de su nombre y su NIT.
- 2.5. Restaurant: es una clase que tiene los atributos de los restaurantes que existen en el hotel.
 - 2.5.1. setCuenta(): es un método que asigna un monto al atributo precio igual a lo gastado.
 - 2.5.2. getCuenta(): es un método que devuelve el precio de lo gastado.
- 2.6. Gimnasio: la clase de uno de los servicios que el hotel brinda. Da información sobre las máquinas que se encuentran en él.
 - 2.6.1. setCuenta(): es un método que asigna un monto al atributo precio igual a lo gastado.
 - 2.6.2. getCuenta(): es un método que devuelve el precio de lo gastado.
- 2.7. CentroDeConvenciones: la clase de uno de los servicios que el hotel brinda, da información de los paquetes para actividades.
 - 2.7.1. setCuenta(): es un método que asigna un monto al atributo precio igual a lo gastado.
 - 2.7.2. getCuenta(): es un método que devuelve el precio de lo gastado.
- 2.8. Bar: clase de uno de los servicios que el hotel brinda, da información de las bebidas que brinda y sus precios.
 - 2.8.1. setCuenta(): es un método que asigna un monto al atributo precio igual a lo gastado.
 - 2.8.2. getCuenta(): es un método que devuelve el precio de lo gastado.
- 2.9. Polideportivo: clase de uno de los servicios que el hotel brinda, da información de los precios al utilizar las instalaciones.
 - 2.9.1. setCuenta(): es un método que asigna un monto al atributo precio igual a lo gastado.

- 2.9.2. `getCuenta()`: es un método que devuelve el precio de lo gastado.
- 2.10. MasterSnack: clase de uno de los servicios que el hotel brinda. Da información de los snacks y precio de los mismos.
 - 2.10.1. `setCuenta()`: es un método que asigna un monto al atributo precio igual a lo gastado.
 - 2.10.2. `getCuenta()`: es un método que devuelve el precio de lo gastado.
- 2.11. ClubNiños: clase de uno de los servicios que el hotel brinda, da información sobre actividades y el precio de las mismas.
 - 2.11.1. `setCuenta()`: es un método que asigna un monto al atributo precio igual a lo gastado.
 - 2.11.2. `getCuenta()`: es un método que devuelve el precio de lo gastado.
- 2.12. Discoteca: clase de uno de los servicios que el hotel brinda, con información sobre horarios, capacidad entre otros.
 - 2.12.1. `setCuenta()`: es un método que asigna un monto al atributo precio igual a lo gastado.
 - 2.12.2. `getCuenta()`: es un método que devuelve el precio de lo gastado.
- 2.13. Enfermeria: clase de uno de los servicios que el hotel brinda, da información del servicio que puede prestar y su costo.
 - 2.13.1. `setCuenta()`: es un método que asigna un monto al atributo precio igual a lo gastado.
 - 2.13.2. `getCuenta()`: es un método que devuelve el precio de lo gastado.
- 2.14. Torre: clase de una de las formas en que están distribuidas las habitaciones. Consta una matriz de la clase Habitaciones, simulando los niveles y las habitaciones de las torres.
 - 2.14.1. `getCuenta()`: es un método que devuelve el monto de la cuenta de dicha habitación.

2.15. Modulo: otra forma de distribución de las habitaciones. Consta de una matriz de Habitaciones, simulando los niveles y las habitaciones en ellas.

2.15.1. `getCuenta()`: es un método que devuelve el monto de la cuenta de dicha habitación.

2.16. Habitación: es la clase de la habitación, tiene información sobre la misma. Así como el cliente a la cual está reservada o está dada.

2.16.1. `getCuenta()`: obtiene la cuenta, el monto correspondiente de cada habitación.

2.16.2. `getCaracteristicas()`: devuelve las características de cada habitación.

2.16.3. `setCuenta(float monto)`: asigna un monto nuevo a la cuenta de la habitación.

2.16.4. `setCliente(String codigoUnico)`: vincula un cliente a la habitación, por medio del código único del huésped.

3. Glosario de términos

3.1. Atributo: Un atributo de una clase son los adjetivos que las diferencias. Pueden ser cualidades, estados. Es de lo que se conforma cada clase.

3.2. Clase: Se le llama así a la generalización de objetos que comparten los mismos atributos y métodos. Es la forma de separar objetos.

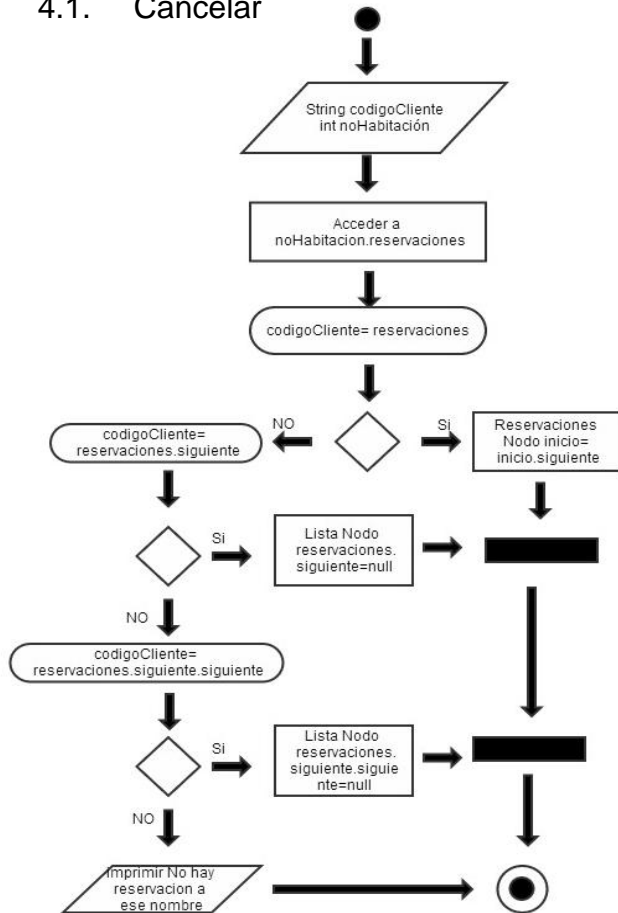
3.3. Herencia: Es un tipo de relación, quiere decir que la clase que hereda, va a contar y disponer de los atributos y métodos de la clase de la cual esta heredando.

3.4. Método: Son las acciones que cada objeto puede tener.

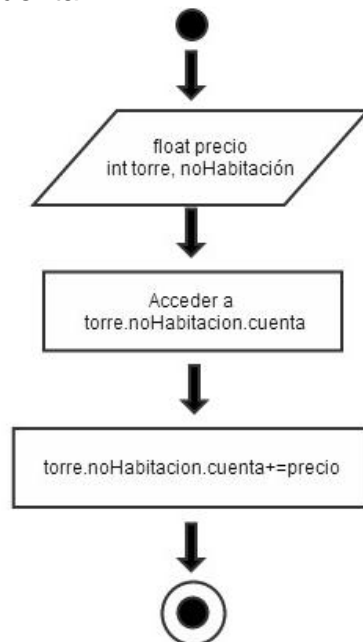
3.5. Relación: Son las asociaciones que pueden tener unas clases con otras.

4. Algoritmos importantes

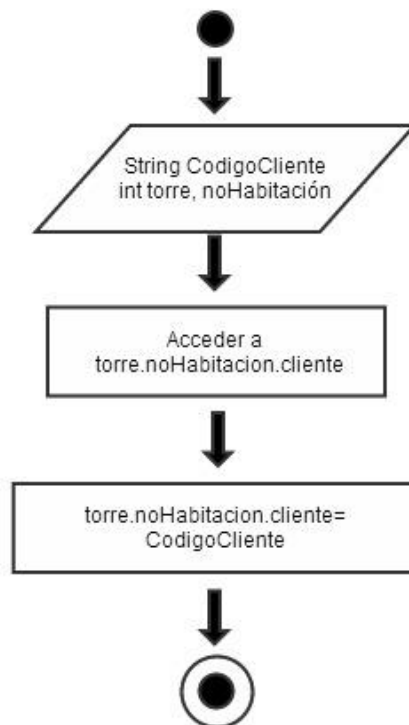
4.1. Cancelar



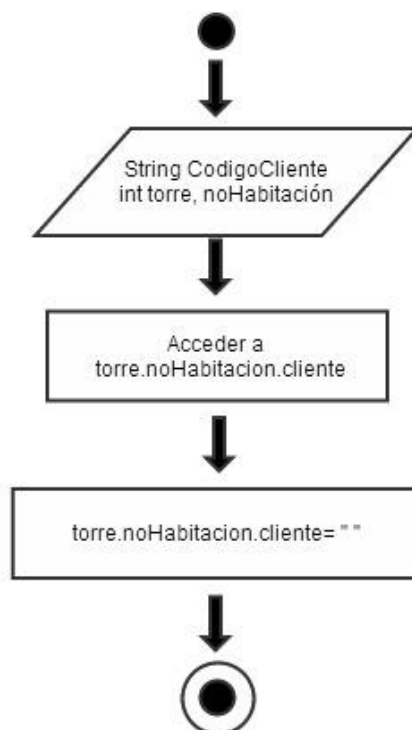
4.2. cargar a la Cuenta



4.3. checkIn



4.4. checkOut



4.5. Reservar

