



TRABAJO PRÁCTICO

FRBA-HOTELES

MATERIA: GESTIÓN DE DATOS.

AÑO: 2014.

CUATRIMESTRE: 2º.

CURSO: K3022.

GRUPO N°: 29.

INTEGRANTES:

Apellido y Nombre	Nro. Legajo
Cardoso, Ariel	146.525-9
Fernández, Lautaro G.	146.524-7
Rey, Nicolás	146.768-2.

ÍNDICE

Log In	Página 3
Funciones, Roles y Usuarios	Página 3
Habitaciones y Hoteles	Página 3
Clientes	Página 4
Reservas y Estadías	Página 4
Registro de Consumibles y Facturación	Página 5
Listados Estadísticos	Página 5
Consideraciones Generales	Página 5

ESTRATEGIA.

Log In.

Para esto, decidimos que la aplicación empiece logueada con el usuario “guest” y en caso de ser otro usuario, este puede loggarse con su nombre y contraseña, accediendo a las funcionalidades que le correspondan. Luego selecciona un hotel y, en el caso que el usuario tenga más de un rol, podrá elegir el rol que desea, sino se selecciona automáticamente el único rol que tenga. Para poder desarrollar esta función, utilizamos stored procedures.

El usuario puede desloguearse (volviendo al usuario “guest”) y también cambiar su contraseña (estando logueado con su usuario).

Funciones, Roles y Usuarios.

Para los roles, consideramos que ciertas funcionalidades sean exclusivas, por su importancia, para el rol Administrador. Por eso es que para el alta de roles no están disponibles todas las funciones disponibles del sistema.

La contraseña de un usuario se puede cambiar al loggarse, pero desde el formulario de modificación.

Creamos una tabla separada para los datos de los empleados, teniendo estos un único usuario asignado. Además, ningún empleado tiene asignado al usurario “admin”, ni tampoco ningún cliente tiene asignado al usuario “guest”, siendo estos genéricos.

Asimismo, un usuario puede tener más de un rol, y estos dependen del hotel sobre el cual se quiere operar.

Habitaciones y Hoteles.

Cuando se crean habitaciones no se limpian todos los atributos por si el administrador quiere cargar varias habitaciones con aquellos atributos. En la baja de una habitación sólo se muestran las que no están dadas de baja. En la modificación sólo se puede pasar del estado de dado de baja a normal y no al revés, porque para eso se utiliza el formulario de baja.

Cuando se realiza la inhabilitación de un hotel, esta se guarda en la tabla “Periodo_Inhabilitado” (compuesta por el código del hotel, fecha de inicio, fecha de fin y una descripción), pudiendo realizarse esto si la habitación no está asignada a una reserva o está involucrada en una estadía en el futuro.

Para la recarga por estrella de hotel, creamos una tabla aparte que tiene fecha y recarga, dado que este dato era general para todos los hoteles. Existe un procedure encargado de cargar nuevas recargas.

Cientes.

Para los clientes que en la tabla Maestra se encontraban erróneos (cuyo tipo y número de documento eran el mismo pero sus datos personales no coincidían, o el mail estaba repetido) decidimos migrarlos igualmente activando un atributo de erróneo. Con esto, en la funcionalidad que trata a los clientes se presenta una sección donde se pueden visualizar los mismos.

El acceso por parte de la aplicación a los datos de la tabla de clientes se realiza a través de vistas, para abstraer a la aplicación de la estructura de la tabla. De esta manera, para las operaciones sobre la vista realizamos triggers instead of.

Volviendo a la migración, para el tipo de documento entendimos que todos los clientes tenían pasaporte y, además, que todos provenían de la Argentina, más precisamente de la Capital Federal (por el nombre de las calles).

Reservas y Estadías.

Aquí decidimos separarlas porque representan entidades diferentes.

En cuanto a la reserva, en el modelo, decidimos que persistan la fecha en que se realizó la misma, la fecha para la que se desea reservar y la cantidad de días, la identificación del cliente que la realizó (para después poder cobrarle a ese cliente en particular) y el tipo de régimen seleccionado (consideramos que se elige el mismo para todas la reserva), además de un código. A la reserva se la relaciona con las habitaciones concretas que se reservan. Para esto existe un procedure que verifica la disponibilidad y otro que realiza la reserva efectivamente.

Por el lado de la aplicación, para realizar la reserva, habiendo elegido el hotel sobre el cual realizar la misma en el inicio, luego, dentro del formulario de reservas, ingresar el período que se desea, seleccionar un régimen y tipo y cantidad de habitaciones. Luego de ingresado todos los datos, se procede a verificar la disponibilidad. En caso que se pueda realizar la reserva, se informa el precio y se procede al ingreso de los datos del cliente. Al terminar el ingreso de los mismo, se efectiviza la reserva.

Si el cliente no existe a la hora de realizar la reserva, se lo puede dar de alta; en el caso que si exista, si no es erróneo, se lo puede buscar por ciertos datos.

La modificación de la reserva se realiza de manera similar sólo que se no se inserta una nueva reserva. Para cancelar una reserva se realiza una búsqueda de la misma y se procede a dar de baja.

También, existe una entidad que almacena el usuario y fecha en la que se realizó alguna de estas operaciones sobre la reserva. Para la cancelación existe otra entidad aparte de la anterior.

Volviendo a la estadía, esta contiene los usuarios que la registraron, las fechas en que sucedieron los check in y check out, el código de la reserva y un código propio de estadía. Con la estadía se relacionan las habitaciones, los clientes involucrados y los consumibles (detallados por habitación). Lo que se facturará en el paso siguiente será una estadía.

Tanto como para la reserva como para la estadía es necesario ingresar datos ciertos para realizar la búsqueda y no se permite buscar entre los clientes existentes.

Registro de Consumibles y Facturación.

Los consumibles se registran al finalizar la estadía luego de realizar el check-out de la misma, el registro de los consumibles se realizara a nivel habitación y se especificará la cantidad del mismo (por cada tipo de consumible) consumido en esa habitación. Nos valimos de stored procedures para realizar esta función.

Para el medio de pago de la factura, decidimos registrar un código de tipo de pago en la factura (efectivo o tarjeta), y luego insertar los datos del pago con tarjeta, si así fuera.

Las validaciones pertinentes a la facturación se realizan en los stored procedures.

Listados Estadísticos.

Para la consideración del listado de los puntos, decidimos calcular la cantidad de puntos previamente y almacenarlos en la factura por un tema de performance a la hora de calcular el puntaje.

Para resolver este punto usamos funciones específicamente creadas y las utilizamos desde la aplicación.

Consideraciones Generales.

La fecha de inicio de la aplicación será el día después de la última factura registrada en el sistema después de la migración (la misma es: 2017-01-06). Esta se encuentra en la clase Home, al igual que la ruta de la Base de Datos.

Consideramos que una estadía no puede tener el mismo día de ingreso que de egreso, es decir, debe pasar al menos un día.

Cada vez que se presiona el botón volver, se vuelve al menú principal y se debe volver a seleccionar un hotel.

Es muy importante recordar los codigos de reserva como el de estadía, ya que son indispensables para realizar correctamente las funcionalidades.