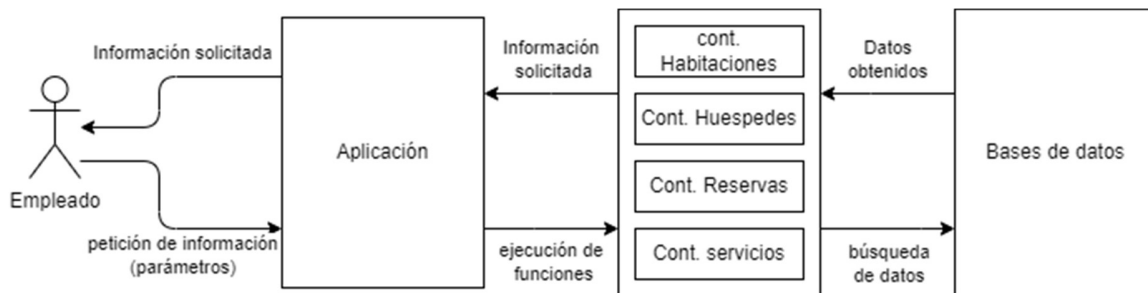


## DOCUMENTO DE IMPLEMENTACIÓN

### CONTEXTO:

El programa PMS está orientado para el uso de los empleados y clientes. El usuario se relaciona directamente solo con la aplicación. Esta aplicación es la que se conecta los controladores del programa, y estos controladores buscan la información en la base de datos, que en este caso está en archivos de texto. Después de obtener la información, se la dan a la aplicación y este se la da al usuario.



### REQUERIMIENTOS FUNCIONALES:

A diferencia del diseño del proyecto anterior, este proyecto está orientado tanto a clientes como empleados. Además, para evitar problemas para el acceso a la información de cada tipo de empleados, se reunieron todos los requerimientos y se les dieron a los empleados en general, por lo que ahora los requerimientos se dividen de la siguiente manera:

- **Requerimientos funcionales empleado:**
  - Crear una habitación en el inventario.
  - Cargar tarifa de habitación por tipo de cuarto, en un rango de fechas y en ciertos días de la semana.
  - Establecer o cambiar la tarifa de un servicio del hotel.
  - Crear un producto de restaurante.
  - Crear reserva a nombre de uno o varios huéspedes.
  - Generar factura para una reserva.
  - Consultar inventario de habitaciones.
  - Registrar el consumo de un servicio a nombre de uno o más huéspedes.
  - Cancelar las reservas para las que falten más de 48 horas.
  - Generar archivo de log con el historial de reservas de un grupo de huéspedes.
  - Pagar el check-out por medio de pasarelas virtuales.
- **Requerimientos funcionales cliente:**
  - Crear reserva a nombre de uno o varios huéspedes.
  - Consultar inventario de habitaciones.
  - Pagar la reserva por medio de pasarelas virtuales.

## RESPONSABILIDADES:

A su vez, cada requerimiento tiene lo cumple un controlador específico. En la siguiente tabla se pueden ver las responsabilidades asignadas a cada controlador.

#	Responsabilidades	Componente
1	Avisar si no hay una tarifa definida en los próximos 365 días.	Controlador Habitaciones
4	Cargar tarifa de habitación por tipo de cuarto, en un rango de fechas y en ciertos días de la semana.	
5	Cargar un archivo .csv con la información de las habitaciones en el inventario.	
6	Consultar inventario de habitaciones.	
9	Crear una habitación en el inventario.	
11	Generar archivo de log con el historial de reservas de un grupo de huéspedes.	Controlador Huéspedes
2	Cargar los datos de los usuarios.	Controlador Reservas
7	Crear reserva a nombre de uno o varios huéspedes.	
12	Generar factura para una reserva.	
3	Cargar menús del restaurante.	Controlador Servicios
8	Crear un producto de restaurante.	
10	Establecer o cambiar la tarifa de un servicio del hotel.	

## EXPLICACIONES ADICIONALES:

1. Para iniciar sesión, se necesita conocer el usuario y contraseña, ya sea de empleado o de cliente. Para esto hay un documento .txt en los datos del proyecto. Adjunto los datos de esto:

```
nombre;usuario;contraseña;rol
Luis García;luis_garcia;abc123;Empleado
Pedro Sánchez;pedro_sanchez;contraseña;Empleado
Javier López;javier_lopez;password;Empleado
Diego González;diego_gonzalez;secret;Empleado
Andrés Torres;andres_torres;admin;Empleado
Jairo;jagc;12345;Cliente
```

Se puede iniciar sesión con cualquiera. Además, se pueden crear más usuarios de clientes utilizando el botón SignIn en la interfaz.



Ilustración 1: Interfaz de inicio de sesión

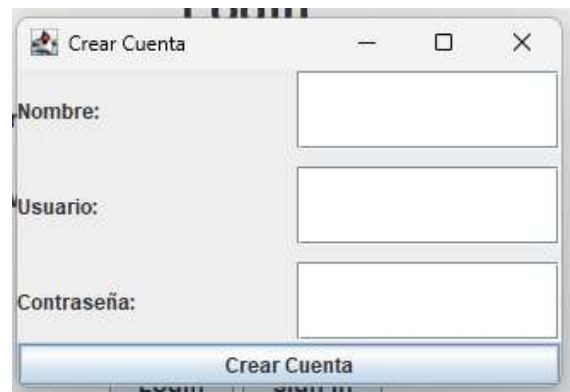


Ilustración 2: Interfaz de crear cuenta

2. Para cargar la tarifa de un servicio o producto, hay que elegir entre servicio o producto del menú. Por problemas en el diseño hay que elegir el servicio/producto por medio de su índice-1. Adjunto fotos con los índices:

```

App.java LoginScreen... EmployeeView... ClientView.java PaymentGUI.java PayPalInterf... ReporteVent... Servicios.txt x 2
1 hombreServicio;tipoCobro;lugarServicio;precio
2 Tour por la ciudad;persona;Hotel;150000.0
3 Masaje relajante;persona;Spa;40000.0
4 Traslado al aeropuerto;grupo;Hotel;50000.0
5 Acceso al gimnasio;persona;Gimnasio;10000
6 Alquiler de bicicletas;grupo;Hotel;7000.0
7 Clase de yoga;persona;Gimnasio;15000
8 Tratamiento facial;persona;Spa;70000
9 Sesión de sauna;persona;Spa;20000.0
10 Sesión de jacuzzi;grupo;Hotel;40000.0
11 Paseo en bote;grupo;Flaca;400000.0
12 Tour de compras;persona;Hotel;40000.0
13 Clase de natación;persona;Piscina;15000.0
14 Excursión en kayak;grupo;Flaca;100000.0
15 Sesión de manicura/pedicura;persona;Spa;40000
16 Sesión de maquillaje;persona;Spa;60000

App.java LoginScreen... ClientView.java PaymentGUI.java PayPalInterf... ReporteVent... MenuRestau... x 3
1 hombreServicio;tipoCobro;lugarServicio;tipoProducto;rangeHoras;precio
2 Desayuno buffet;persona;Restaurante;comida;6:30-10:00;10000.0
3 Almuerzo buffet;persona;Restaurante;comida;12:00-15:00;50000
4 Cena buffet;persona;Restaurante;comida;19:00-22:00;55000
5 Café americano;persona;Restaurante;bebida;6:00-23:00;5000
6 Café con leche;persona;Restaurante;bebida;6:00-23:00;7000
7 Refresco en lata;persona;Restaurante;bebida;6:00-23:00;6000
8 Jugo natural;persona;Restaurante;bebida;6:00-23:00;10000
9 Hamburguesa con papas;persona;Restaurante;comida;11:00-22:00;20000
10 Pizza mediana;persona;Restaurante;comida;11:00-22:00;25000
11 Sandwich mixto;persona;Restaurante;comida;11:00-22:00;14000.0
12 Cerveza nacional;persona;Restaurante;bebida;18:00-23:00;8000
13 Cerveza importada;persona;Restaurante;bebida;18:00-23:00;16000.0
14 Coteal del día;persona;Restaurante;bebida;18:00-23:00;5000.0
15 Botella de vino;persona;Restaurante;bebida;18:00-23:00;70000
16 Ramen;persona;Restaurante;comida;14:30-15:45;50000.0
17 Tostada Asada;persona;Restaurante;comida;14:30-15:45;70000.0
18 Tamal;persona;Restaurante;comida;6:00-10:00;8000.0
19 Pasa pellenas;persona;Restaurante;comida;6:00-18:00;3000.0
20 Cholo;persona;Restaurante;comida;8:00-22:00;10000.0

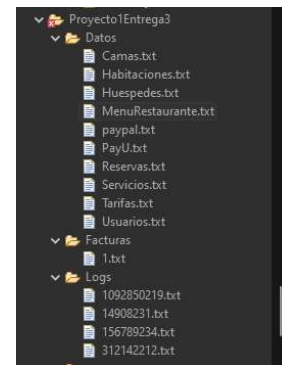
```

Por ejemplo, si quiero cambiar el precio del desayuno buffet ingreso en ID del producto el número 1.



Ocurre lo mismo en el caso de registrar consumo de una habitación. Para los que se ingresa por el índice del servicio/producto.

3. Para la función de crear una reserva, se asume que ya se conoce el inventario de habitaciones, por lo que el ID de las habitaciones, al igual que las fechas en las que están reservadas y demás características aparecen ahí. Al crear una reserva se crea con el ID de la habitación.
4. Las facturas se generan en formato .txt, donde el nombre es el ID de la reserva. Estos documentos se añaden a la carpeta Facturas, que se encuentra en los datos del proyecto. De la misma manera, el historial se crea, pero en vez de usar el ID de la reserva se usa el documento del cliente, y estos se guardan en la carpeta Logs de los datos.
5. Existen tantas clases, porque cada función que se utiliza cuenta con su propia interfaz gráfica.
6. Para el caso de pagar el checkout o la reserva, se crearon dos pasarelas: PayU y PayPal. Para poder simular el uso de una tarjeta se crearon 2 archivos de texto que contienen datos inventados de tarjetas, cada archivo es para cada pasarela. Para pagar la interfaz pide elegir una reserva, y esa tiene su propio costo que depende del costo de la reserva y los servicios adquiridos. En las siguientes fotos se pueden ver el contenido de los archivos.



```

App.java LoginScreen.... ClientView.java
1 Nombre;numCuenta;clave;saldo
2 Paulo;1234567890123456;0007;100000.0
3 Juana;4444222299990033;1234;4970000.0
4 Mario;1234432112344321;1029;200000.0
5

```

Ilustración 2 Tarjetas de PayPal

```

App.java LoginScreen.... ClientView.java Pay
1 Nombre;numCuenta;clave;saldo
2 Paulo;1234567890123456;0007;9960000.0
3 Juana;4444222299990033;1234;50000.0
4 Mario;1234432112344321;1029;1820000.0
5

```

Ilustración 3 Tarjetas de PayU

7. En el proyecto, en la carpeta de datos, se encuentra un archivo de texto llamado pruebas, en la que aparece una prueba para realizar manualmente de cada función. Como las funciones que aparecen en el cliente son iguales a las utilizadas por el empleado, solo se sugieren pruebas para empleado.