**Proyecto 2 Diseño**

Mateo Calderón Rincón - 202213589 - m.calderonr

Natali Andrea Bohórquez Jiménez – 202214826 - n.bohorquezj

Juan David García Rojas - 202213434- jd.garciar12

**Ventana Administrador**

Para crear la ventana del administrador necesitábamos primero que todo entender cuales eran las actividades que este realizaba, para que la ventana contenga todo lo necesario para que el administrador pueda desempeñar todas sus funciones. Es importante dividir las tareas por grupos, evidentemente que cada una de esas divisiones tengan una razón para que el administrador pueda entender mejor la ventana y se le facilite usarla. Para ello pensamos en crear 5 paneles, uno para Crear Habitaciones, para las consultar la información sobre las reservas, para modificar las Tarifas de las Habitaciones, para crear y/o modificar los Servicios que el hotel presta y uno para Información.

Evidentemente, en cada uno de estos paneles hay cosas distintas:

* Panel Crear Habitaciones:
* Identificador: opción para ingresar un nombre o número para la habitación que se está creando.
* Tipo de Habitación: que son las que nos da el enunciado.
* Tipo de Camas: hay muchas maneras de representar esta información, para hacerlo más sencillo está el tamaño de la cama y entre paréntesis la capacidad que se tienen por las camas.
* Adicionales: la posibilidad de poner alguno de los adicionales que nos da el enunciado (tomamos la decisión de que solo se pudiera seleccionar alguno de los adicionales o ninguno si es preciso, esto porque buscamos varias formas de representarlo en el panel, intentamos hacerlo mediante una lista plegables, pero donde se pudieran seleccionar varias opciones, pero no nos fue posible, por lo que decidimos dejar que solo se escogiera una).
* Crear Habitación: conforme a los datos ingresados los guarda en el archivo “habitaciones\_creadas.csv”.
* Cargar archivo: abre el explorador de archivos para que el administrador pueda seleccionar el archivo de su preferencia.
* Panel Reservas
* Consultar Reservas: la idea es que al oprimir este botón se abra otra ventana con los números de reservas vigentes (que fueron creados por el recepcionista) para que el administrador las pueda visualizar.
* Mostrar Fechas: se buscaba crear un calendario en donde se pudieran visualizar las reservas (intentamos implementar JCalendar, pero fue imposible ya que no se encontraba incorporado en nuestro Eclipse). Por lo que no se pudo realizar.
* Tarifas Habitaciones
* Tipo de Habitación: en donde se puede seleccionar alguno de los tipos de habitación que nos da el enunciado.
* Fecha Inicio: como lo dice el enunciado se puede seleccionar un rango de fechas (intentamos hacerlo más practico con alguna extensión que permitiera poner la fecha más práctico, pero no lo logramos). Por eso dejamos un campo de texto.
* Fecha Fin: sucede lo mismo que con Fecha Inicio.
* Nueva Tarifa: el administrador pone el valor que adquiere esa habitación por noche en ese rango de fechas.
* Cambiar tarifa: agrega la información que se dejo en los campos anteriores, y lo guarda en el archivo “tarifas\_habitaciones.csv”.
* Servicios
* Cargar Archivo Menú: al igual que anteriormente se usaba un botón para que el administrador seleccionara un archivo del explorador de archivos, ahora sucede lo mismo, pero con el menú.
* Crear Plato: este botón abre la Ventana Crear Plato (que se explicara más adelante).
* Cambiar Tarifas: el botón abre la Ventana Cambiar Tarifas (que también se detallara más adelante).
* Ventana Crear Plato

Esta ventana solo tiene un panel, en donde ingresa:

* Nombre: el administrador ingresa el nombre del plato que desea crear.
* Lugar: una lista plegable, en donde se especificaba si el platillo se podía consumir en el restaurante, la habitación o en ambos,
* Precio: el valor de ese plato.
* Crear Plato: el botón guarda la información de los campos anteriores en el archivo “platos\_creados.csv”.
* Ventana Cambiar Tarifa

Esta ventana solo tiene un panel, y esta para que el administrador modifique el valor del platillo que desea.

* Nombre: el administrador debe poner el nombre del plato del que desea modificar el precio
* Precio: el nuevo valor del plato
* Cambiar Precio: el botón actualiza el archivo “platos\_creados.csv”.

**Ventana Inicio de Sesion**

Para el diseño de la ventana de inicio de sesión, se tomaron las siguientes decisiones:

**Título y tamaño de la ventana:**

Título: "Inicio de sesión" para indicar claramente el propósito de la ventana.

**Tamaño**: Se estableció un tamaño de 500x300 píxeles para proporcionar un espacio suficiente para los elementos del formulario y mantener una apariencia equilibrada.

1. Diseño de la ventana:

**BorderLayout:** Se utilizó un diseño de BorderLayout para colocar los componentes principales en el centro de la ventana y permitir una mejor organización de los paneles.

1. Paneles de inicio de sesión y creación de usuario:

**JTabbedPane:** Se utilizó un JTabbedPane para crear dos pestañas: "Iniciar sesión" y "Crear usuario". Esto proporciona una forma intuitiva de alternar entre las dos funcionalidades.

* Panel inicio de sesión: Contiene los campos para ingresar el nombre de usuario y contraseña, así como un botón "Iniciar sesión" para desencadenar la acción correspondiente.

·**Panel creación de usuario**: Contiene los campos para ingresar el nuevo nombre de usuario, contraseña y correo electrónico, junto con un botón "Crear usuario" para completar el proceso de creación.

1. Acciones y eventos:

·**ActionListener**: Se implementó la interfaz ActionListener para gestionar los eventos de los botones "Iniciar sesión" y "Crear usuario".

·**Lógica de inicio de sesión**: En el evento del botón "Iniciar sesión", se puede agregar la lógica necesaria para verificar las credenciales ingresadas y permitir el acceso correspondiente.

·**Lógica de creación de usuario**: En el evento del botón "Crear usuario", se puede agregar la lógica para validar los campos ingresados y guardar la información del nuevo usuario en algún lugar apropiado (base de datos, archivo, etc.).

Al tomar estas decisiones de diseño, se buscó crear una interfaz de inicio de sesión simple y fácil de usar, que permita a los usuarios acceder al sistema o crear una cuenta de forma intuitiva. Los paneles separados para el inicio de sesión y la creación de usuario facilitan la distinción de las dos funcionalidades principales. Además, la implementación de eventos y lógica adecuada en los botones proporciona la funcionalidad necesaria para autenticar y crear usuarios según sea necesario.

**Ventana para otros empleados**

Panel de Factura:

El panel de factura se utiliza para mostrar la factura generada para una habitación específica, los servicios seleccionados y el total a pagar.

Componentes del panel de factura:

* 1. **habitacionField**: Es un campo de texto donde se muestra el número de habitación.
  2. **serviciosField**: Es un campo de texto donde se muestra la lista de servicios seleccionados.
  3. **totalField**: Es un campo de texto donde se muestra el total a pagar.
  4. **actualizarButton**: Es un botón utilizado para actualizar la factura en caso de que se realicen cambios.

1. **Funcionalidad del panel de factura:**
   1. El método actualizarFactura(int habitacion, String servicios, double total) se utiliza para actualizar los campos de texto con los nuevos valores de habitación, servicios y total. Este método es llamado desde la clase EmpleadoGUI.

**Clase EmpleadoGUI:**

1. La clase EmpleadoGUI es una interfaz gráfica de usuario (GUI) que permite a un empleado generar una factura para una habitación y seleccionar servicios adicionales.

Componentes de la clase:

* 1. **habitacionPanel**: Es un panel que contiene una etiqueta y un campo de texto para ingresar el número de habitación.
  2. **serviciosPanel**: Es un panel que contiene casillas de verificación para seleccionar servicios adicionales.
  3. **facturaPanel**: Es un panel que muestra la factura generada.
  4. **habitacionField**: Es el campo de texto donde se ingresa el número de habitación.
  5. **spaCheckBox**, **comidaCheckBox**, **licorCheckBox**: Son casillas de verificación para seleccionar los servicios de spa, comida y licor, respectivamente.
  6. **generarFacturaButton**: Es un botón que, al hacer clic, genera la factura en función de los servicios seleccionados.
  7. **facturaTextArea**: Es un área de texto donde se muestra la factura generada.

Funcionalidad de la clase EmpleadoGUI:

* 1. El método actionPerformed(ActionEvent e) maneja los eventos de clic en el botón generarFacturaButton. Este método obtiene el número de habitación y los servicios seleccionados, calcula el total de la factura y muestra la factura generada en el facturaTextArea.

La clase EmpleadoGUI se encarga de la creación y organización de los diferentes paneles y componentes en la interfaz gráfica. Permite al empleado ingresar el número de habitación, seleccionar los servicios adicionales y generar una factura con el total a pagar.

En resumen, el panel de factura y la clase EmpleadoGUI trabajan juntos para proporcionar una interfaz gráfica donde se puede ingresar el número de habitación, seleccionar servicios adicionales y generar una factura detallada.

**Ventana Recepcionista**

Primer tomamos en cuenta cuales eran las funciones del recepcionista. Se crea un panel por cada una de las funciones principales que tiene el recepcionista y de esta manera de facilita el desplazamiento por la aplicación.

Evidentemente, en cada uno de estos paneles hay cosas distintas:

* Factura

Muestra la suma la suma total

* Camas

Esta ventana muestra los tipos de camas que existen y que se pueden agregar a la reserva

* King
* Sencilla
* Kid
* Doble
* Queen
* Crear reserva

Esta ventana tiene un panel donde se le pregunta al usuario

* Número de reserva: numero con el que se va a guardar la serva creada.
* Número de personas: el recepcionista ingresa el número de habitaciones para que el sistema le informe que habitaciones tiene disponibles
* Número de camas: al igual que en el anterior el número de camas deseadas en la habitación.
* Precio: el valor deseado habitación.
* Verificar: el botón verificar si existe alguna habitación con los datos ingresados en un archivo .csv llamado “crear habitacio.csv”. si es así se reservará la habitación de lo contrario se le informará al usuario que no existe una habitación así y que debe ingresar nuevos datos.
* Crear Habitación

Esta ventana es para que el recepcionista cree las habitaciones

* Número de personas: el recepcionista ingresa el número de personas para que el sistema le informe que habitaciones tiene disponibles con ese aforo.
* Número de camas: el recepcionista ingresa el número de camas que tendrá disponible inicialmente esa habitación.
* Precio: precio habitación.
* Guardar: este botón guarda la información de la nueva habitación en un archivo .csv llamado “craer habitacio.csv”.
* Check in
* confirma el ingreso del usuario y que por ende la habitación se encuentra ocupada.
* Check out
* confirma la salida del usuario y que por ende la habitación se encuentra desocupada.