

PRÁCTICA 1

Partes opcionales realizadas

Creación de hojas de estilos CSS y uso de elementos específicos HTML para la información



Francisco Pérez Díaz (GII + Mat.)
José Pertierra das Neves (GII + Mat.)
Sergio Santamaría Carrasco (GII + Mat.)

Contenido

Introducción.....	1
Clases Básicas.....	1
Aplicación Distribuida en Java	2
Aplicación Web	5

Introducción

La práctica de Sistemas Distribuidos consta de dos proyectos. En el primero implementaremos la aplicación distribuida, que constará de Servlet, de Cliente, y de las clases básicas necesarias. El segundo desarrollaremos la aplicación Web, que constará de Servlet, archivos HTML, archivos CSS y clases básicas en Java.

Dado que ambos proyectos poseen una parte común, las clases básicas en Java comenzaremos realizando una breve descripción de estas.

Clases Básicas

Huésped: Clase utilizada para representar un huésped que reserva habitación. Contiene su nombre, apellidos, nif, domicilio, fecha de nacimiento, teléfono fijo, móvil y email.

Domicilio: Clase utilizada para representar el domicilio de un huésped. Contiene dirección, localidad, provincia y código postal.

Reserva: Clase utilizada para representar la reserva de una habitación. Contiene fechas de entrada y salida, huésped que la renta y número de habitación asignada.

ListaClientes: Clase que representa la lista de huéspedes que están o han estado en el hotel. Contiene métodos útiles para buscar Huéspedes según determinados parámetros.

ListaReservas: Clase que representa la lista de reservas de habitaciones que hay o ha habido en el hotel. Contiene métodos útiles para buscar reservas según determinados parámetros.

Cliente: Aplicación nativa de Java que efectuará el papel de Cliente.

Cabe destacar que las clases de ambos proyectos no son exactamente iguales, pues en uno se requerían métodos que no eran necesarios en el otro y viceversa.

Aplicación Distribuida en Java

La distribución de este primer proyecto en paquetes ha sido la siguiente:



El **paquete por defecto** contiene al Servlet de la aplicación.

El **paquete Aplicación** contiene la clase del Cliente y la clase necesaria para ejecutar esta.

El **paquete ClasesBase** y el paquete Listados contiene las clases base de nuestra práctica descritas anteriormente.

El **paquete Mensajes** contiene las clases usadas para contestar desde el Servlet al Cliente.

A continuación se describen las clases que solo se necesitan en este apartado:

MensajeRespuesta: Clase utilizada para devolver los mensajes de las operaciones realizadas con éxito.

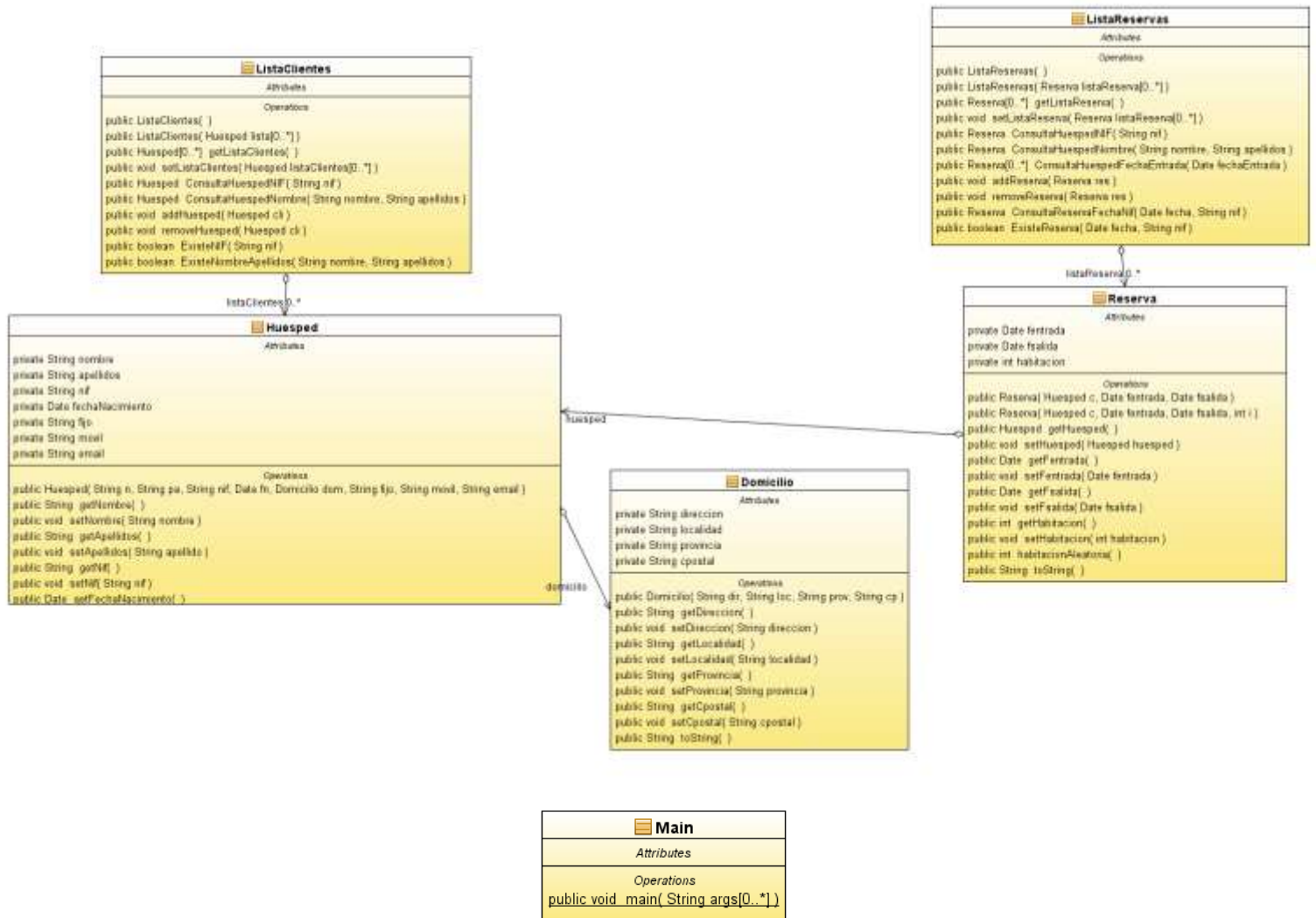
MensajeErrorDate: Clase utilizada para devolver un mensaje de error cuando el resultado de una consulta de reserva por fecha es vacía.

MensajeErrorNameSurname: Clase utilizada para devolver un mensaje de error cuando el resultado de una consulta de Huésped por nombre y apellidos es vacía.

MensajeErrorDateNIF: Clase utilizada para devolver un mensaje de error cuando el resultado de una consulta de Reserva por fecha y NIF es vacía.

MensajeErrorNIF: Clase utilizada para devolver un mensaje de error cuando el resultado de una consulta de Huésped por NIF es vacía.

Para facilitar la comprensión de cómo está estructurado el proyecto se especifica a través de un diagrama de clases generado usando las herramientas de NetBeans




```

classDiagram
    class Cliente {
        package BufferedReader sc = new BufferedReader(new InputStreamReader(System.in, "ISO-8859-1"))
        package PrintStream consolaesp = new PrintStream(System.out, true, "ISO-8859-1")
        package String direccion_ip = "localhost"
        package String proyecto = "Practica_1_Ej_2"
        package String nombre_servlet = "Servlet"
        package String ruta = "http://" + direccion_ip + ":8080/" + proyecto + "/" + nombre_servlet

        +Cliente()
        +void consultarHuésped()
        +void añadirHuésped()
        +void modificarHuésped()
        +void consultarReservas()
        +void modificarReserva()
        +void eliminarReserva()
        +void añadirReserva()
        +void menu_ppal()
    }
    
```

 MensajeErrorDate
Attributes
private Date fecha
Operations
public Date getFecha()
public void setFecha(Date fecha)

 MensajeErrorNameSurname
Attributes
private String Nombre
private String Apellidos
Operations
public String getNombre()
public void setNombre(String Nombre)
public String getApellidos()
public void setApellidos(String Apellidos)

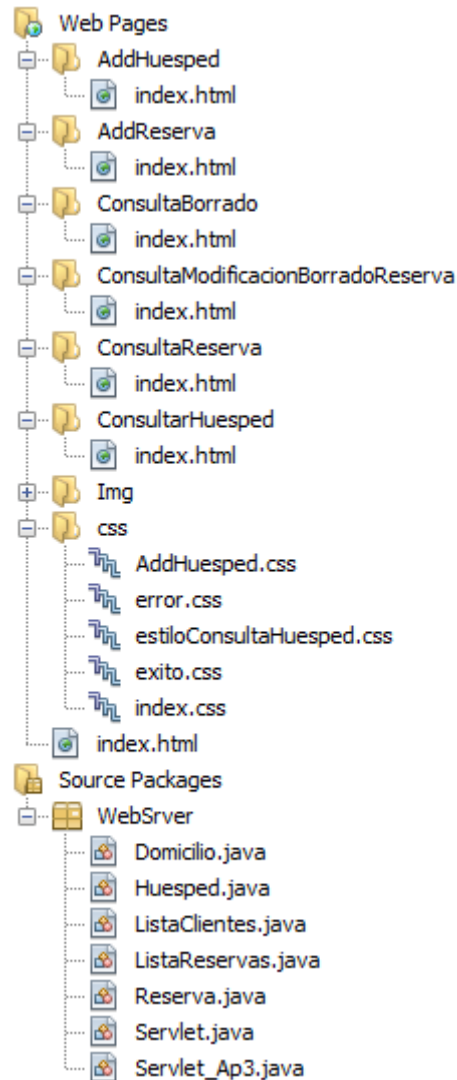
 MensajeErrorDateNIF
Attributes
private Date fecha
private String nif
Operations
public Date getFecha()
public void setFecha(Date fecha)
public String getNif()
public void setNif(String nif)

 MensajeErrorNIF
Attributes
private String NIF
private String texto
Operations
public String getNIF()
public void setNIF(String NIF)
public String getTexto()
public void setTexto(String texto)

 MensajeRespuesta
Attributes
private String texto
Operations
public String getTexto()
public void setTexto(String texto)

Aplicación Web

La distribución de este primer proyecto en paquetes ha sido la siguiente:



En **WebPages** se guardan (en su respectiva subcarpeta) los documentos HTML y CSS

En **WebSrver** se almacenan las clases básicas mencionadas anteriormente y el Servlet.

A continuación se describen los archivos que se han usado para este apartado:

- Archivos HTML Estáticos:

Index.html: Representa la página principal de la aplicación web. Desde ella se le da la posibilidad al usuario de acceder a toda la funcionalidad de la web.

ConsultarHuesped.html: Página desde la que se podrá realizar la consulta de un Huésped bien por NIF o bien por nombre y apellidos.

AddHuesped.html: Página desde la que se permite añadir un nuevo huésped. Incluye un formulario que solicita al usuario los datos básicos.

ConsultaBorrado.html: Página desde donde se solicita al usuario el NIF del huésped que desea borrar o modificar.

ConsultaReserva.html: Página desde donde se solicita la fecha de entrada necesaria para llevar a cabo una consulta de reservas.

AddReserva.html: Página desde donde se solicita la fecha de entrada, la fecha de salida y el NIF necesarios para añadir una reserva.

ConsultaModificacionBorradoReserva.html: Página desde donde se solicita la fecha de entrada y el NIF necesarios para poder modificar o eliminar una reserva.

- Páginas HTML dinámicas: Son implementadas dentro del Servlet y son usadas para mostrar las operaciones de una consulta. Dado que no tienen un nombre específico se nombran usando el método de la clase Servlet_Ap3 que las genera.

generarError: Se mostrará cuando una operación o consulta no se realice correctamente especificando en su cuerpo el motivo del fallo.

generarExito: Se mostrará cuando una operación se realice correctamente redirigiendo a la página principal pasado una pequeña cantidad de tiempo.

generarAddCliente: Similar a AddCliente.html. El usuario será redirigido a ella cuando trate de añadir una reserva con el NIF de un huésped no existente.

generarConsulta2: Muestra los resultados de una consulta de Huésped y permite modificar o eliminar este mismo.

generarConsultaReserva: Muestra los resultados de consultar una lista de reservas por fecha de entrada.

generarModElimReserva: Muestra los resultados de una consulta de reserva por fecha de entrada y NIF, y permite modificar o eliminar dicha reserva.

- Páginas de Estilos CSS:

index: Se encarga del diseño básico de la aplicación (color de background, estilo de botones...) ,por ello en todas las páginas se encuentra importarla.

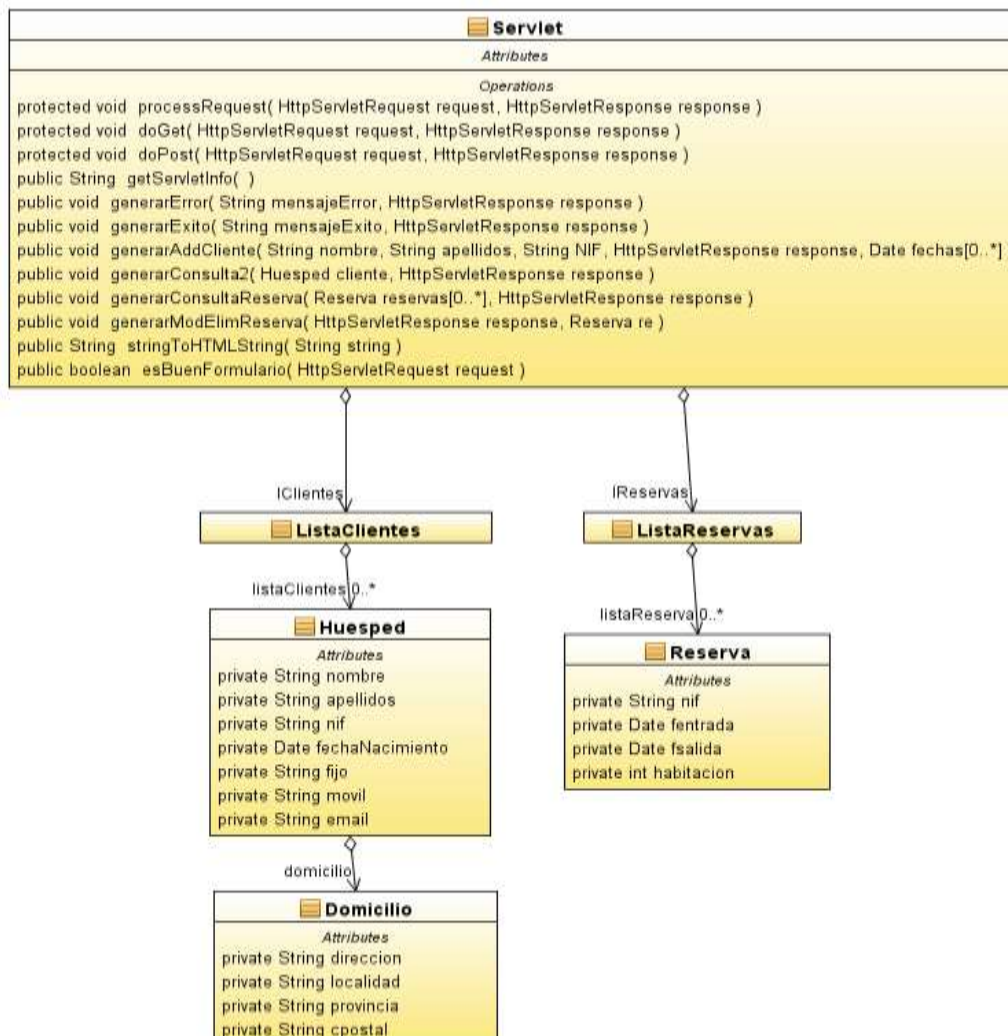
estiloConsultaHuesped: Se encargará del diseño en el documento ConsultarHuesped.html, se centrará en la presentación de los formularios.

addHuésped: Proporciona estilos al formulario de creación de un nuevo huésped en la aplicación.

error: Proporciona estilos a los mensajes de error que aparecerán en el documento del servidor cuando se produzca un fallo en la operación realizada.

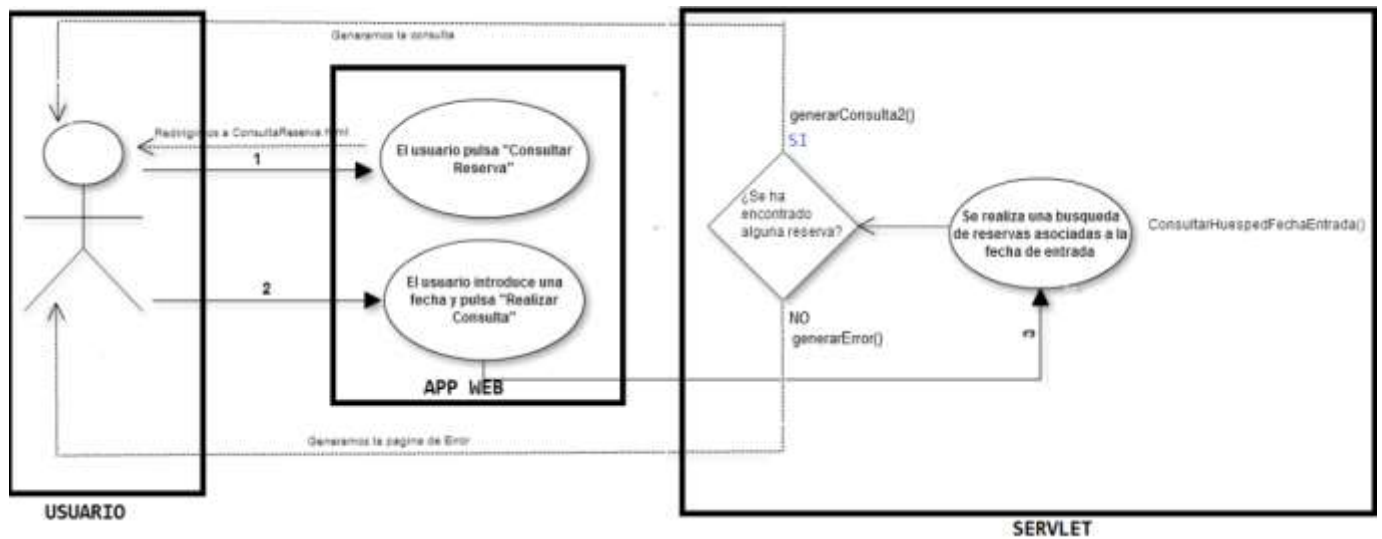
exito: Proporciona estilos a los mensajes de éxito que aparecerán cuando se realice bien una operación de gestión tipo: crear huésped, reserva, eliminar huésped,reserva...

Para facilitar la comprensión de cómo está estructurado el proyecto se especifica a través de un diagrama de clases generado usando las herramientas de NetBeans.



El usuario interactúa con la página web y solicita distintos servicios a través de esta. La información es remitida al Servlet que genera una página HTML dinámica para mostrar los resultados de la operación al usuario.

Para esclarecer el funcionamiento se presenta un diagrama de casos de uso en el que el usuario realiza una consulta de reservas por fecha de entrada.



La **URL** inicial de la aplicación es: <http://localhost:8080/Parte3SD/index.html>