

**Grado en Ingeniería de Computadores – Curso 2020-21**

**Ingeniería del Software: Laboratorio**

**Cuaderno de prácticas. P3 - Diseño. (MVC, Modelo de clases, Diagrama de secuencia)**

**Solución del ejercicio práctico:**

Contenido

[1. Catálogo de requisitos 2](#_Toc33110094)

[2. Diagrama de clases aplicando patrón MVC 3](#_Toc33110095)

[2.1. Diagrama de clases completo 4](#_Toc33110096)

[2.2. Diagrama de clases del caso de uso IDENTIFICAR USUARIO 5](#_Toc33110097)

[2.3. Diagrama de clases para el caso de uso BORRAR PEDIDO 6](#_Toc33110098)

[2.4. Diagrama de clases para el caso de uso DAR DE ALTA NUEVO CLIENTE 7](#_Toc33110099)

[2.5. Diagrama de clases para el caso de uso HACER PEDIDO 8](#_Toc33110100)

[3. Diagramas de secuencia 11](#_Toc33110101)

[3.1. Diagrama se secuencia del caso de uso IDENTIFICAR USUARIO 11](#_Toc33110102)

[3.2. Diagrama se secuencia del caso de uso BORRAR PEDIDO 12](#_Toc33110103)

[3.3. Diagrama se secuencia del caso de uso DAR DE ALTA NUEVO CLIENTE 13](#_Toc33110104)

[3.4. Diagrama se secuencia del caso de uso HACER PEDIDO 14](#_Toc33110105)

[4. Diagrama de comunicación 16](#_Toc33110106)

# Catálogo de requisitos

Se identifican nuevos requisitos tecnológicos:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Id | Nombre | Descripción | Fuente | Tipo | Prioridad | Fecha | Autor |
| RT03 | MVC | El diseño de la aplicación seguirá el patrón Modelo-Vista-Controlador (MVC) | J.R. Hilera | T | Alta | 21/02/20 | J.R. Hilera |
| RT04 | Java | Se utilizará Java como lenguaje de programación | J.R. Hilera | T | Alta | 21/02/20 | J.R. Hilera |
| RT05 | Base Datos | La base de datos a utilizar es de tipo NoSQL con tecnología MongoDB | Responsable | T | Alta | 21/02/20 | J.R. Hilera |

# Diagrama de clases aplicando patrón MVC

Cada ventana identificada en los casos de uso se programará en una clase que representa una vista, a la que se asocia una clase que actúe como controlador de dicha vista, que accederá a las clases identificadas en el modelo conceptual que incluyan la información y la lógica de negocio que se necesita en la vista.

Se elabora una tabla con las clases de las tres categorías.

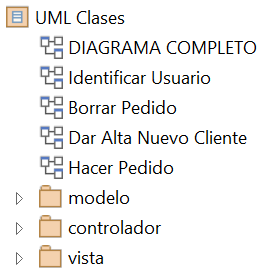
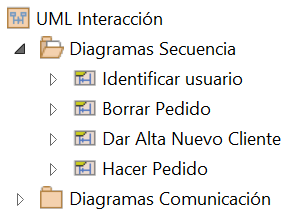
| **Vista** | **Controlador** | **Modelo** |
| --- | --- | --- |
| VistaAltaCliente | ControlAltaCliente | Cliente |
| VistaBorrarPedido | ControlBorrarPedido | Pedido |
| VistaHacerPedido | ControlHacerPedido | Factura |
| Libro |
| Orden de impresión |
| Pedido |
| DetalleLibroPedido |
| DetalleOrdenImpresion |
| Cliente |
| VistaIdentificarUsuario | ControlIdentificarUsuario | Usuario, ~~Empleado, Responsable~~ |
| VistaMenuPrincipal | --- | --- |

Por simplicidad, se toma la decisión de diseño de eliminar las clases Empleado y Responsable y fusionarlas en la clase Usuario, pues sólo se diferencian en un atributo, y manejar las tres clases complicaría la aplicación. Se decide convertir el atributo “confianza” de la clase Empleado, en un nuevo atributo de la clase Usuario con el nombre “tipoUsuario”, que puede tener cuatro valores: “Responsable”, “Empleado de confianza”, “Empleado”, “” (cuando ese usuario no existe en la Base de Datos).

Se hacen los cambios en el modelo de clases, y se añaden las nuevas clases al diagrama de clases, ubicando cada clase en una carpeta con el nombre de la categoría.

Los atributos de las clases del controlador son los objetos del modelo y de la vista que conectan. Los atributos de las clases de la vista son los elementos de pantalla (botones, etiquetas, cajas de texto, etc.).

El siguiente paso es añadir las operaciones a todas las clases, para lo que hay que apoyarse en los diagramas de interacción, haciendo un diagrama de clases y un diagrama de interacción para cada caso de uso. De os dos tipos de diagramas de interacción que existen en UML, se recomienda utilizar el diagrama de secuencia en lugar del diagrama de comunicación.

## Diagrama de clases completo



## Diagrama de clases del caso de uso IDENTIFICAR USUARIO



## Diagrama de clases para el caso de uso BORRAR PEDIDO

****

## Diagrama de clases para el caso de uso DAR DE ALTA NUEVO CLIENTE

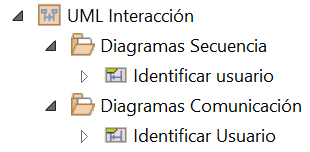
****

## Diagrama de clases para el caso de uso HACER PEDIDO



# Diagramas de secuencia

Para organizar los diagramas de interacción, tanto los de secuencia como los de comunicación. Se crea una vista “UML Interacción” de tipo “Dynamic” y una carpeta para cada uno de ellos. Y para crear un diagrama de secuencia, se elige dentro de la carpeta la opción “Add Element” > Type: Interaction.



## Diagrama se secuencia del caso de uso IDENTIFICAR USUARIO



## Diagrama se secuencia del caso de uso BORRAR PEDIDO



## Diagrama se secuencia del caso de uso DAR DE ALTA NUEVO CLIENTE



## Diagrama se secuencia del caso de uso HACER PEDIDO



# Diagrama de comunicación

Se abre un diagrama de secuencia, se seleccionan todos los elementos, y en el menú que se abre con el botón derecho del ratón se elige Transform > Communication, y se selecciona la carpeta en la que se quiere almacenar el diagrama de comunicación que se va a generar.

El diagrama generado aparece desordenado, hay que colocar los elementos. Es posible que surja algún error ya haya que realizar algunas correcciones manualmente.

Como ejemplo, se muestra el diagrama de comunicación del caso de uso BORRAR PEDIDO.

