



**Tecnologías Avanzadas de Desarrollo**  
**Grado en Ingeniería Informática en Sistemas de Información - Curso 20232024**  
**EPD 3: Frameworks – Laravel:Paneles de Administración**

La entrega del trabajo se hará a través de la tarea correspondiente en el Campus Virtual y se completará cuando se reciba resguardo de la entrega. Los alumnos deben comprobar que el archivo depositado en la plataforma es correcto descargándolo de la misma y comprobando su contenido. Pasado el límite de entrega se aceptará el envío del trabajo, con una penalización de 0.2 puntos sobre 1 de la calificación por cada hora o fracción de retraso. La entrega consistirá en un único fichero comprimido en formato ZIP cuyo nombre deberá ser de la forma Apellido1Apellido2Nombre.zip. En caso de ser una entrega grupal, el alumno debe incluir en un fichero README.txt quiénes son los integrantes de su grupo incluyéndose a sí mismo. Dentro de dichas carpetas se incluirán exclusivamente los archivos necesarios en la resolución del correspondiente ejercicio o problema, pueden ser código, documentos de texto o cualquier formato/s requeridos. Las rutas de los ficheros empleados serán relativas, a fin de que las resoluciones a los ejercicios y problemas puedan ser examinadas en cualquier equipo. Cualquier entrega que no cumpla las reglas de nombrado, el formato de compresión del archivo o el contenido de los archivos del mismo será penalizado con 0.2 puntos sobre 1 por cada incumplimiento.

### Objetivos

---

- Operaciones CRUD con PHP – Laravel integrando Bootstrap 5
- Análisis de la base de datos e implementaciones de caso de uso principal
- Uso del sistema de Login-Signin
- Diseño y necesidades de un panel de administración
- Configuración de SMTP con Laravel y uso en la recuperación de contraseñas

### Conceptos

---

Todos los conceptos y experimentos vistos en clase están en las diapositivas relativas a la EPD3.2.

### Bibliografía

---

Composer Oficial:

<https://getcomposer.org/>

Node.js Oficial:

<https://nodejs.org/es/>

Documentación oficial Laravel 9:

<https://laravel.com/docs/9.x/releases>

Tutoriales y otros:

<https://laracasts.com/>

<https://desarrolloweb.com/home/laravel>

Libro "LARAVEL DE CERO A DIEZ":

<https://norvicsoftware.com/laravel-de-cero-a-diez/>

Fortify:

<https://laravel.com/docs/10.x/fortify>

<https://diarioprogramador.com/laravel-9-autenticacion-con-fortify-y-bootstrap-5/>

SMTP:

<https://mailtrap.io/home>



### **Problema 0: Creación de un proyecto Laravel en base a unos requisitos dados**

---

El Proyecto se hará en grupos de 3, si no tiene grupo debe comunicar con su profesor/a.

Cada grupo tendrá un color asignado, que será el color primario de la aplicación web, su temática debe estar relacionada con el color suministrado, use la psicología del color y cree su imagen de marca.

Se debe crear un comercio electrónico lo más realista posible.

La temática es libre, pero:

- a) Se debe presentar la arquitectura de la base de datos basada en modelo UML.
- b) Objetivos y requisitos del proyecto.
- c) Casos de uso – funcionalidad muy detallada. Caso de uso principal: Poder comprar productos.
- d) Mockups (papel, en línea, bocetos, etc.)
- e) Debe de hacerse uso de Bootstrap 5 para la creación de interfaces y se pueden usar plantillas base, pero requerirán modificación, si la plantilla no contiene modificaciones significativas el grupo será calificado como suspenso.
- f) Sistema de control de Versiones con Git y GitHub. Se debe entregar un enlace a una release en el repositorio del proyecto, por tanto, éste debe ser público.
- g) Documentar decisiones de diseño importantes.

Se recuerda a los alumnos que este es el proyecto examen de la asignatura, que deben exponer el día del examen de ésta, y que iremos ampliando en funcionalidad.

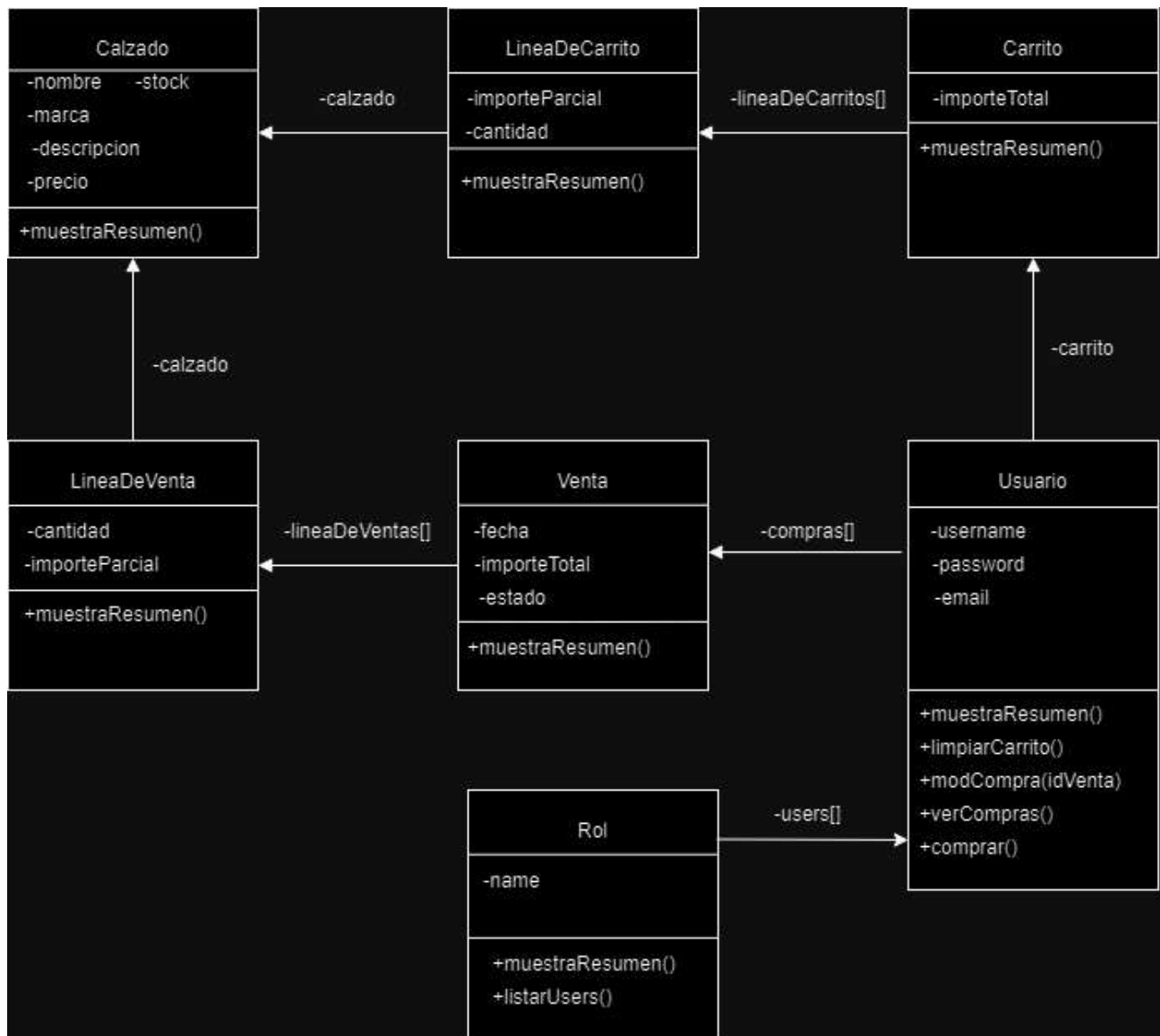


### Problema 1: Crear esquema de bases de datos

Cada grupo debe presentar un esquema UML de la arquitectura de la base de datos relativa al comercio electrónico a realizar. Debe ser lo más completa posible, sin excedernos. Productos, carrito de la compra, compras, usuarios, direcciones de envío, favoritos, descuentos, categorías, etc.

El único requisito es que al menos tiene que haber una relación de cada una de las cardinalidades 1:1, 1:N y N:M, aunque como descubrirán en breve, les será imposible crear un sistema sin ellas.

#### Diagrama UML:



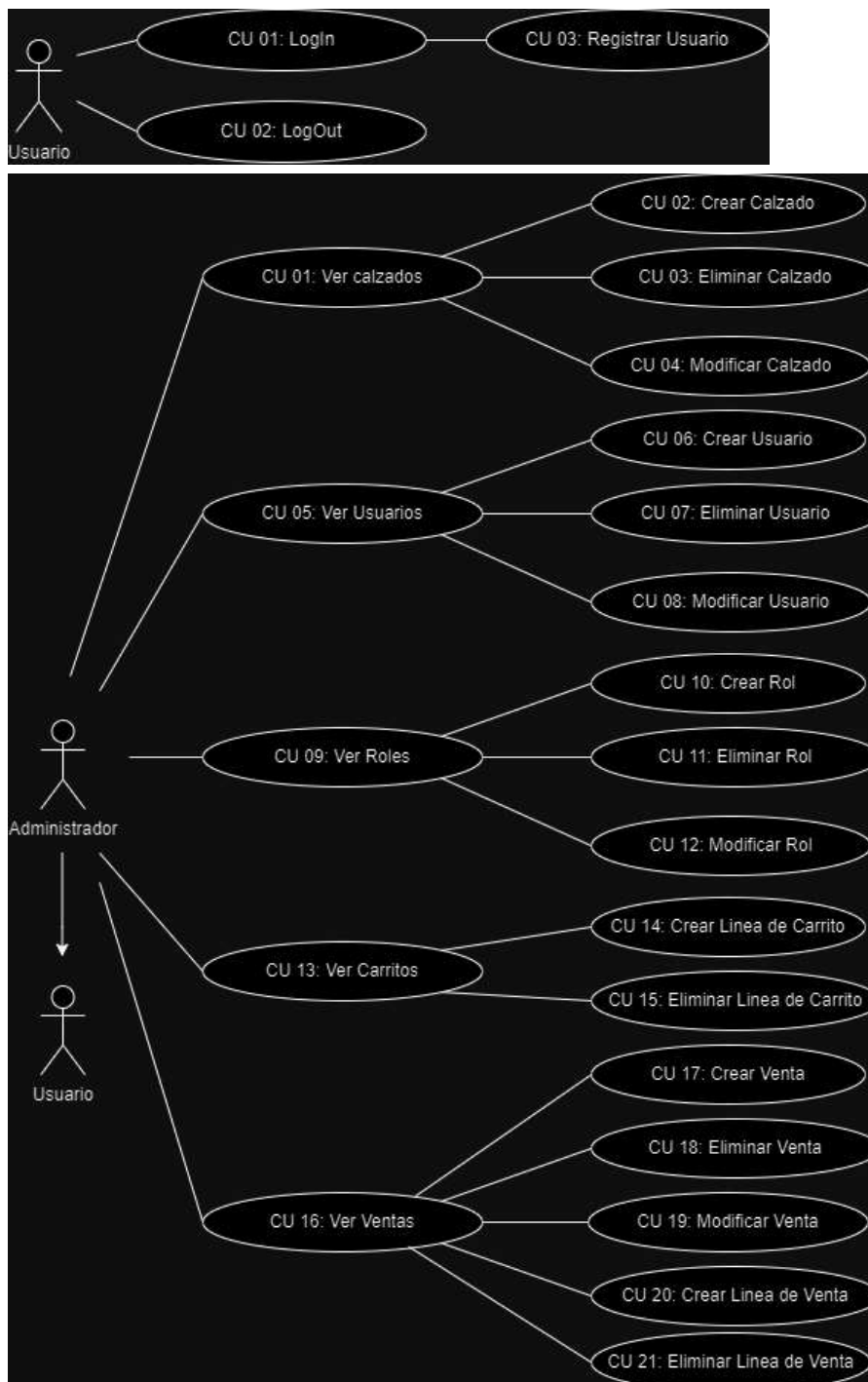


## Problema 2: Desarrollo del caso de uso principal: v1.0

En la versión 1 de nuestro proyecto debemos ser capaces de poder comprar productos siendo usuarios y ver el estado de nuestro pedido. Cuando un usuario efectúe una compra (botón comprar y notificación correspondiente a nivel visual en la web, también recibirá un correo electrónico con los detalles de su pedido).

El administrador por su parte debe ser capaz de ejercer las operaciones propias sobre dichos productos y observar los pedidos realizados por los usuarios.

Se recuerda que un usuario nunca puede tener acceso a las funcionalidades del administrador, debe de contar con un sistema de autenticación completo.





### Problema 3: Memoria

---

Cada grupo debe documentar todo lo pedido en el Problema 0. En esta versión faltarán cosas, irá ampliándose a medida que el proyecto crezca, hay que centrarse en el caso de uso principal.

**Repositorio GitHub:** <https://github.com/Miguesg95/ProyectoTAD3.2>

**Temática:** Portal de venta online de calzado.

**Paleta de colores:**



**Logo:**



**Vistas:** Se ha creado una plantilla básica usando Blade para la cabecera y el pie de página.



## Panel de Administrador:

Calzados

Usuarios

Roles

Carritos

Ventas

### Calzados en el sistema

Nombre	Marca	Descripción	Precio	Stock		
Sandalias de deo	adidas	con talla y color	30	35	Actualizar	Eliminar
zapatos de baila	flamenco	arsa y ole	400	15	Actualizar	Eliminar
botas de montaña	puma	pa montañear	20	30	Actualizar	Eliminar

### Alta Calzado

Nombre

Marca

Descripción

Precio

Stock

El panel de Administrador consiste en 5 desplegables controlados por los botones superiores. Cada uno lleva a una de las secciones de la página. En cada sección podemos ver una tabla que lista y detalla el objeto dedicado en la sección y el administrador tiene la posibilidad de ejecutar las operaciones CRUD pertinentes sobre ellos. Las celdas de las tablas son editables para poder modificar la información que tenga sentido poder actualizar. Además podemos encontrar formularios para crear objetos.

Calzados

Usuarios

Roles

Carritos

Ventas

### Ventas en el sistema

Usuario	Identificador	Fecha de Realización	Estado	Contenido	Importe Total		
pepe12	1	2024-05-04 16:00:55	En preparacion	<div>X Sandalias de deo - cantidad: 2 - 30\$</div> <div>X zapatos de baila - cantidad: 1 - 400\$</div> <div>X botas de montaña - cantidad: 1 - 20\$</div>	450	Actualizar	Eliminar
Margari	2	2024-05-04 14:41:45	Confirmado	<div>X zapatos de baila - cantidad: 2 - 800\$</div>	800	Actualizar	Eliminar
usuario	4	2024-05-04 16:21:00	Confirmado	<div>X zapatos de baila - cantidad: 2 - 800\$</div> <div>X Sandalias de deo - cantidad: 3 - 90\$</div>	890	Actualizar	Eliminar

Usuario	Identificador	Fecha de Realización	Estado	Contenido	Importe Total		
---------	---------------	----------------------	--------	-----------	---------------	--	--

Algunas de estas tablas tienen la posibilidad de eliminar contenido pulsando el botón con la X. Los datos no editables son o bien datos que no deben ser modificados, como el identificador o las fechas de realización de compra, o datos que derivan de otros, como los importes parciales y totales que se actualizan solos.

Una compra se crea a partir de un carrito y vacía este en el proceso.



## Datos de la Práctica:

### Estimación temporal:

- Parte presencial: 240 minutos.
  - Explicación y resolución de dudas: 120 minutos.
  - Creación del proyecto y organización: 60 minutos.
  - Establecimiento de sistema de autenticación: 60 minutos
- Parte no presencial: 20-40 horas dependiendo de la destreza técnica del alumno.
  - Lectura y estudio del guión y bibliografía básica: 60-90 minutos.
  - Problemas: 20-40h.

**Autor del documento:** Olga M. Moreno Martín (Abril 2024).