



28 DE MAYO DE 2024

**PRÁCTICA FINAL.**  
ENTORNOS DE DESARROLLO.

GRUPO SEMIPRESENCIAL  
EFA EL CAMPICO



# PRÁCTICA FINAL.

## 1. INTRODUCCIÓN.

Siempre se mantiene la intención firme de que el alumno progrese con nuevos retos que desarrollen nuevas destrezas y que adentren al alumno en tareas de investigación adicionales que proporcionen un hábito necesario para el desarrollo profesional.

La aplicación ha pretendido incorporar el uso de las llamadas metodologías ágiles. Qué significa esto, que en la gestión de proyectos se utilizan técnicas que optimizan la entrega de resultados al cliente sin perder calidad en las entregas. Las metodologías ágiles están basadas en el desarrollo iterativo e incremental, a través de equipos auto-organizados y multidisciplinarios, donde los requisitos y soluciones evolucionan con el tiempo a diferencia de las metodologías en cascada basadas en un desarrollo secuencial que ordena rigurosamente las etapas del proceso, cada etapa debe esperar la finalización de la etapa anterior.

A continuación, detallamos el contenido de la práctica final de la asignatura de Entornos de Desarrollo.

Partimos del desarrollo que se ha conseguido en el grupo presencial de la asignatura de dejó la aplicación en el mes de abril con el menú principal y dos opciones desarrolladas:

1. Opción 1 – Mostrar por consola los productos de nuestra tienda.
2. Opción 2 \_ Realizar el proceso de compra en nuestra tienda añadiendo a ese proceso el registro de la información de compra en un ticket.

Nuestra tienda se expande y aparecen nuevas necesidades asociadas a dicha expansión. Para ello desde la dirección de proyectos se han tomado una serie de medidas con carácter urgente que afectan a las siguientes fases del desarrollo del proyecto:

# PRÁCTICA FINAL.

## 2. FASES.

### 2.1. Captura de requisitos:

Dentro de esta fase se van a explicar los requisitos que debe cumplir la aplicación. Estos requisitos no deben ser implementados, simplemente deben ser tomados en cuenta para poder realizar los diagramas solicitados

Dado el éxito que nuestra prueba inicial de tienda ha tenido, la dirección ha decidido anticiparse y plantea una serie de funcionalidades que complementen la aplicación inicial para que pase a producción. Para ello, debemos almacenar información sensible que inicialmente no se contemplaba además de realizar posibles ajustes en procesos existentes.

**A partir de aquí las funcionalidades que a continuación se expresan no deben ser implementadas con código, insisto, solo se describe la funcionalidad, pero no debe ser implementada en ningún caso. Esta información será utilizada para realizar los diagramas solicitados.**

#### 2.1.1. Primera funcionalidad, F1:

Cualquier almacenamiento de datos será bajo el protocolo que ampara la ley de protección de datos adoptado por nuestra compañía. En tal caso:

##### 2.1.1.1. Creación de entidades nuevas.

- **Entidades obligatorias (I).**

Crearemos una nueva entidad llamada **Cliente**. Los clientes vienen identificados inicialmente con un código, numerocliente, nombre, apellidos, dirección, localidad, provincia, país, codigopostal, teléfono, mail, observaciones.

Crearemos una entidad llamada **Pedido**. Los pedidos vienen identificados por ordendepedido, codigocliente, producto, cantidad.

# PRÁCTICA FINAL.

## ○ Entidades obligatorias (II).

- **Proveedor.** Los proveedores vienen identificados inicialmente con un código, nombre, cif, codigopostal, país, nombrecontacto, cargocontacto, mail, teléfono, codigocontable, monedapordefecto, tipoiva, tipodepago, plazodepago, numerodecuenta.
- **Albarán.** Los albaranes vienen identificados inicialmente con un código, numeroalbaran, fechaalbaran, producto, cantidad.
- **Ticket: Completar los atributos que debe tener esta clase en base a vuestras deliberaciones**

Este apartado concreto elegiréis donde desarrollarlo y planteo como opciones las siguientes: El **diagrama de clases** de la aplicación en la fase de ingeniería del software o el **modelo Entidad-Relación** correspondiente a la fase de diseño de BD.

### 2.1.1.2. Carga de productos en la aplicación.

Ya es conocido, que la carga de nuestros productos en la aplicación se hace de una forma un tanto tosca.

Sería interesante que nuestros productos apareciesen cargados en la estructura diseñada a tal fin en la base de datos y posteriormente en su carga en la aplicación se obtuviesen de esa misma estructura.

Esta tarea consiste en generar los cambios necesarios y los procedimientos adecuados para tal fin.

### 2.1.2. Segunda funcionalidad, F2:

Antes de describir este apartado es conveniente realizar una serie de aclaraciones

**F2.1.2.1. Primer apartado:** Se obtendrá un listado de órdenes de compra realizadas por parte de nuestros clientes  
Para no complicar en exceso no contemplaremos inicialmente fecha inicio y fecha fin. Listaremos en tal caso los pedidos efectivos realizados que estén almacenados. Es decir, los pedidos/órdenes de compra que durante el proceso han confirmado una compra.

# PRÁCTICA FINAL.

**Importante recordad estos datos deben almacenarse en la estructura de base de datos que se diseñe.**

## **Aclaración:**

**Para generar una orden de compra / pedido es necesario:**

- **Tener un cliente:** Supondremos para este caso que todos los clientes que van a usar la aplicación están registrados en nuestra base de datos y han pasado el proceso de login pertinente. No hay que hacer un proceso de registro de clientes, no hay que hacer proceso de login para clientes.
- **Asignar uno o varios productos al pedido.**
- **Pagar dicha compra en este caso con tarjeta**
- **Generar el ticket de compra correspondiente y almacenarlo en un fichero, tal y como ya realizaba la aplicación.**
- **Almacenar el pedido en nuestra base de datos.**

**F2.1.2.2. Segundo apartado:** Se integrará el módulo de pago por tarjeta diseñado en la práctica de programación con la única salvedad como es lógico de que el cliente no pueda realizar pago en efectivo.

El motivo de integrar esta funcionalidad es claro, cobrar el pedido que servimos a nuestro cliente.

## **F2.1.2.3. Tercer apartado:**

Se debe establecer comunicación con la base de datos creada para tal fin. A partir de esta actualización dato de deba guardarse o consumirse se cargará desde esta estructura con una única salvedad.

Hay que tener en cuenta que el Ticket generado en la compra se seguirá imprimiendo guardando en formato txt y la información que contiene se almacenará en la estructura generada para tal fin en la base de datos.

# PRÁCTICA FINAL.

## 2.2. Fase Ingeniería del Software:

El sprint programado para esta fase es el siguiente, (consecución 09/06/2024 hora 23:59):

- Diseño de clases completo que incluya el alcance total del proyecto desde sus inicios hasta esta nueva ampliación. Recordad cuales fueron los inicios de esta aplicación. Revisad las funcionalidades del software desde ese momento hasta estas nuevas funcionalidades
- Diagramas de actividad de la funcionalidad, F2.1.2.1

Para este punto deberías utilizar algún software gratuito utilizados en ingeniería del software tales como: Os dejo un enlace donde podéis evaluar la elección de alguna de ellas.

<https://www.ionos.es/digitalguide/paginas-web/desarrollo-web/las-mejores-herramientas-uml/>

Si decidís elegir como idóneo para la realización de este apartado una herramienta del paquete office, también será tenido en cuenta.

## 2.3. Fase diseño BD:

El sprint programado para esta fase es el siguiente, (consecución 09/06/2024 hora 23:59):

- Diseño de BD necesaria para capturar toda la información necesaria para cubrir las necesidades expuestas. Solamente se pide aquí lo que conocemos como el modelo ER.

# PRÁCTICA FINAL.

## 3. RESUMEN.

Recordad no sobreentender nada en la práctica. En caso de explicación defectuosa o poco apropiada por mi parte, comunicad por los canales establecidos para ello cualquier anomalía que exista para que sea debidamente aclarada.

## 4. OBJETIVOS.

**Los objetivos marcados en esta práctica son claros:**

- 4.1. Por un lado, manejar un volumen de información superior al que normalmente manejamos en las prácticas anteriores. Ser capaces de planificar y organizar nuestro trabajo.
- 4.2. Por otro lado, manejarnos con elementos básicos en la planificación de software en la fase inicial del mismo, conocida como fase de ingeniería de software.

## 5. ENTREGA.

**Documento pdf en el que se incluyan los diagramas pedidos, así como cualquier explicación pertinente para que sean entendidos de manera inequívoca.**

**Como recomendación se advierte que al ser normalmente diseños gráficos se suele acompañar de la correspondiente leyenda.**

**Tened presente el formato que deben mantener los trabajos entregados en pdf,**