| **NOMBRE Y APELLIDOS:**  **Eleazar Gonzalez Moreno** | | | | **FECHA: 18-08-2023** | | |
| --- | --- | --- | --- | --- | --- | --- |
| **DOCENTE: MANUEL MACÍAS PÉREZ** | | | | **NOTA:** | | |
| **(IFCD0210) DESARROLLO DE APLICACIONES CON TECNOLOGÍAS WEB.** | | | | **Nº CURSO: 22-35/008902** | | |
| MF: | 0492 | UNIDADES DE APRENDIZAJE A LAS QUE RESPONDE: | | UA1 | Duración: | 2 h |
| UF: | 1844 |
| PRÁCTICA Nº: | E1 |
| DENOMINACIÓN: **Arquitecturas de aplicaciones web.** | | | | | | |
| **DESCRIPCIÓN**  1.- El alumno de forma individual deberá realizar un proyecto de una aplicación web aplicando el proceso de desarrollo de software:  Aplicación para hacer pedidos. (TEMAS: Ropa, Juegos, Ilustraciones, Servicios)  En el cual se debe detallar:   * La entrada de datos. (Como interactuará el usuario - Formularios – Botones – Elementos interactivos) * Proceso de manipulación o transformación de esos datos. (Cómo se trabajará con los datos – Cómo se guardarán los datos – Cómo se accederá a los datos). * Procesos de salida. (Cómo se mostrarán los datos – Cómo se enviarán los datos).   Ver ejemplo de esquema.  Pegar en este Archivo el resultado de la prueba y convertir en pdf. Enviar o Subir a Github.  La práctica se realizará de manera individual.  **MEDIOS PARA SU REALIZACIÓN**   * Equipo informático. * Aplicación Visual Code Studio instalada en el equipo. * Navegadores actualizados   **PAUTAS DE ACTUACIÓN DEL FORMADOR**  *Al inicio de la práctica, que se desarrollará de manera individual por cada uno de los alumnos, el formador/a realizará las siguientes actuaciones:*  - Fijará los objetivos de la práctica.  - Aportará las instrucciones necesarias a los alumnos/as para la realización de la misma, haciendo hincapié en aquellos aspectos más relevantes.  - Facilitará a cada alumno/a la documentación necesaria para el desarrollo de la práctica.  - Resolverá las dudas que se planteen durante el transcurso de la práctica, con objeto de que el alumnado aprenda y pueda concluir la realización de la misma.  Durante la realización de la práctica el formador/a supervisará el desarrollo de esta para evaluar tanto los procedimientos como el resultado final.  Al finalizar la práctica el formador examinará el desarrollo que han realizado los/as alumnos/as, proponiendo las medidas de corrección, en caso necesario.  ESPECIFICACIONES PARA LA EVALUACIÓN DE LA PRÁCTICA | | | | | | |
| **Resultados a comprobar** | | | **Indicadores de logro** | | | |
| 1. En un supuesto práctico en el que se pide gestionar componentes software en el entorno del servidor mediante herramientas de desarrollo y lenguajes de programación contando con documentación de diseño detallado:   – Crea y archiva componentes software.  – Modifica y elimina componentes software.  – Depura y verifica los componentes software elaborados.  Conforme el criterio de evaluación CE 1.1 | | | * 1. Crear y archivar componentes software | | | |
| * 1. Modificar y eliminar componentes software | | | |
| * 1. Depurar y verificar los componentes software elaborados | | | |
| 1. Formula estructuras de datos y flujo de control mediante lenguajes de programación orientados a objetos según la funcionalidad del componente software a desarrollar   Conforme el criterio de evaluación CE 1.3 | | | 2.1 Formula estructuras de datos mediante lenguajes de programación orientados a objetos según la funcionalidad del componente software a desarrollar | | | |
| 2.2 Formula flujos de control mediante lenguajes de programación orientados a objetos según la funcionalidad del componente software a desarrollar | | | |

**Sistema de valoración**

Definición de indicadores y escalas de medida

Los indicadores que se van a establecer, será una hoja de chequeo, sistema de valoración, que complementa a este documento, donde se evalúan todos los resultados a comprobar (tareas). En este documento, se establecerán a su vez los indicadores de logro que se han de tener en cuenta, para conseguir los resultados a comprobar.

Mínimo exigible

El mínimo exigible para la superación de la práctica es de 50 puntos sobre 100 puntos

# SUPUESTO PRÁCTICO

1.- El alumno de forma individual deberá realizar un proyecto de una aplicación web aplicando el proceso de desarrollo de software:

Aplicación para hacer pedidos. (TEMAS: Ropa, Juegos, Ilustraciones, Servicios)

En el cual se debe detallar:

* La entrada de datos. (Como interactuará el usuario - Formularios – Botones – Elementos interactivos)
* Proceso de manipulación o transformación de esos datos. (Cómo se trabajará con los datos – Cómo se guardarán los datos – Cómo se accederá a los datos).
* Procesos de salida. (Cómo se mostrarán los datos – Cómo se enviarán los datos).

[Ver ejemplo de esquema.](#_heading=h.gjdgxs)

Pegar en este Archivo el resultado de la prueba y convertir en pdf. Enviar o Subir a Github.

La práctica se realizará de manera individual.

# Ejemplo:

* Interacción del usuario:
  + A través de botones en la parte superior – derecha:
    - Botón 1 – Acceder
    - .
    - .
    - .
  + A través de formularios de pedidos donde se recogerán los siguientes datos
    - Formulario de pedido
      * Nombre
      * Correo electrónico
      * Teléfono
    - Formulario de consulta
    - .
    - .
    - .

1. **Entrada de datos(interactuar con el usuario**
   1. Formulario

Tendrá 3 formularios: acceso, registro y contacto.

**Acceso**, recogerá nombre y contraseña para validar con los usuarios registrados.

**Registro**, recogerá los datos del cliente: nombre completo,contraseña, correo electrónico,edad, teléfono, dirección y método de pago.

**Contacto**, formulario para solucionar dudas de los clientes o para recoger sus reclamaciones. Se pedirá nombre de usuario del cliente, correo, un asunto y el contenido del email.

* 1. Botones

Tendrá 10 botones, 5 visibles a primera vista y otros dos inyectados.

**Nuestra Tienda**, el botón muestra en pantalla información de la empresa, como trabaja y publicita alguno de los productos.

**Tienda**, el botón muestra en pantalla a la izquierda un filtro que mostrará los productos según unos atributos y una serie de artículos. Si el filtro no tiene nada seleccionado, mostrará todos los productos de la tienda.  
**Acceso**, el botón muestra el formulario de acceso y un botón para validar los datos, al mismo tiempo mostrará un botón para registrarse.

**Validar**, Este botón valida los datos recogidos en acceso.

**Registrarse**, Muestra el formulario de registro y un botón que guarda los datos de registro del usuario.

**Registro**, Este botón guarda los datos del cliente.

**Contacto**, muestra el formulario de contacto y el botón para enviar el correo.

**Enviar**, envía el correo de contacto almacenado en el formulario de contacto.

**Confirmar**, confirmar el pedido.

**Configuración**, muestra ajustes de alto contraste.

1. **Cómo procesamos los datos y cómo los transformamos**
   1. Almacenar los datos

El formulario de acceso recogerá los datos de usuario y contraseña en un array.

En el formulario de registro se almacenarán los datos en json que se guardarán en el servidor.

Al seleccionar un objeto para comprar se almacenará en un array.

* 1. Validación de datos

Se comparará los datos almacenados en el acceso con los de los usuarios registrados para buscar coincidencias.

* 1. Proceso de datos

Los datos serán almacenados en el servidor como objetos.

1. **Salida de datos**
   1. ¿Cómo van a salir los datos?

Los productos se mostrarán como imagen de los mismos y al seleccionarlos se marcarán como elegidos

Se mostrará en pantalla los elementos seleccionados por si hay que hacer correcciones y junto a cada uno el tallaje a elegir para el pedido y un botón de confirmación del pedido

**SISTEMAS DE VALORACIÓN MF 0492\_3 – UF1844 – E1**

| **RESULTADOS A COMPROBAR** | **INDICADORES DE LOGRO** | **ESCALA DE MEDIDAS** | |  |
| --- | --- | --- | --- | --- |
| 1. En un supuesto práctico en el que se pide gestionar componentes software en el entorno del servidor mediante herramientas de desarrollo y lenguajes de programación contando con documentación de diseño detallado:   – Crea y archiva componentes software.  – Modifica y elimina componentes software.  – Depura y verifica los componentes software elaborados.  Conforme el criterio de evaluación CE 1.1 | 1.1 Crea y archiva componentes software | - Crea y archiva componentes software entre un 75% y 100%  - Crea y archiva componentes software entre un 50 % y 75%  - Crea y archiva componentes software por debajo de un 50 % | B  R  M | 20  10  0 |
| 1.2 Modifica y elimina componentes software | - Modifica y elimina componentes software entre un 75% y 100%.  - Modifica y elimina componentes software entre un 50% y 75%.  - Modifica y elimina componentes software por debajo de un 50%. | B  R  M | 20  10  0 |
| 1.3 Depura y verifica los componentes software elaborados | - Depura y verifica los componentes software elaborados entre un 75% y 100%.  - Depura y verifica los componentes software elaborados entre un 50% y 75%.  - Depura y verifica los componentes software elaborados por debajo de un 50%. | B  R  M | 10  5  0 |
| 1. Formula estructuras de datos y flujo de control mediante lenguajes de programación orientados a objetos según la funcionalidad del componente software a desarrollar   Conforme el criterio de evaluación CE 1.3 | 2.1 Formula estructuras de datos mediante lenguajes de programación orientados a objetos según la funcionalidad del componente software a desarrollar | - Formula estructuras de datos mediante lenguajes de programación orientados a objetos entre un 75% y 100%.  - Formula estructuras de datos mediante lenguajes de programación orientados a objetos entre un 50% y 75%.  - Formula estructuras de datos mediante lenguajes de programación orientados a objetos por debajo de un 50%. | B  R  M | 20  10  0 |
| 2.2 Formula flujos de control mediante lenguajes de programación orientados a objetos según la funcionalidad del componente software a desarrollar | - Formula flujos de control mediante lenguajes de programación orientados a objetos entre un 75% y 100%.  - Formula flujos de control mediante lenguajes de programación orientados a objetos entre un 50% y 75%.  - Formula flujos de control mediante lenguajes de programación orientados a objetos por debajo de un 50% | B  R  M | 30  15  0 |
|  | **Valor mínimo exigible: 50** | **Valor máximo: 100** | |  |