**Entrega 2 - Diseño de la solución**

**Proyecto Pensamiento Computacional**

|  |  |
| --- | --- |
| **Docente** | *Indique el nombre del docente* |
| **Equipo de Trabajo** | *Indique el nombre de uno de los integrantes del equipo* |
| *Indique el nombre de uno de los integrantes del equipo* |
| *Indique el nombre de uno de los integrantes del equipo* |
|  |

**FUNCIONALIDAD DEL SISTEMA**

*Agregue en este espacio las funcionalidades del sistema. Es decir, las opciones que brindará el sistema al usuario. Esta información se encuentra en la descripción del problema que se le entregó a cada equipo.*

**ENTREGA 2**

**2. Solución del problema**

* 1. **Definiciones, fórmulas, métodos o estructuras conocidas que pueden utilizarse en la solución**

**Objetivo:** Identificar y listar los conceptos reutilizables en la solución, como definiciones, fórmulas, métodos o estructuras conocidas en programación, matemáticas, lógica o gestión de datos.

🔹 **Requisitos:**  
✅ Debe incluir **mínimo 3 elementos y máximo 5 elementos**.  
✅ Es obligatorio que **2 de los elementos sean estructuras de programación**.

**📢 Nota:** La tabla incluye ejemplos como referencia. Antes de completarla, elimine los ejemplos y registre los datos de su proyecto.

|  |  |  |
| --- | --- | --- |
| **Tipo de elemento (definición, fórmula, método, estructura de programación)** | **Elemento** | **Explicación** |
| *Ejemplo: Estructura de programación* | *Ejemplo: Condicional* | *Ejemplo: Se utilizarán estructuras if-else para clasificar los pedidos en disponibles o agotados.* |
| *Ejemplo: Definición* | *Ejemplo: Indice de Masa Corporal (IMC)* | *Ejemplo: IMC es una razón matemática que asocia el* ***peso*** *y la* ***altura*** *de un individuo. Comúnmente clasifica a las personas en infrapeso, peso normal, sobrepeso y obesidad. Esta fórmula será utilizada en el programa para clasificar los clientes y determinar si tienen o no riesgo de sobrepeso.* |
|  |  |  |
|  |  |  |
|  |  |  |

* 1. **Lógica y algoritmia**

**Objetivo:** Representar gráficamente la solución mediante un diagrama de flujo que ilustre la secuencia de procesos y decisiones necesarias para resolver el problema.

**🔹 Requisitos:**✅ Debe incluir todas las etapas principales del proceso.  
✅ Debe representar las estructuras lógicas y condiciones utilizadas en la solución.  
✅ Debe incluir puntos de inicio, procesos, decisiones y salidas claramente identificados.

**📢 Nota:** Se recomienda el uso de herramientas como Draw.io, Lucidchart, Figma, PowerPoint o cualquier otra herramienta de diagramación.

* **Dibujar el diagrama de flujo que represente la solución** 
  1. **Diseño de la interacción con el Sistema propuesto**

**Objetivo:** Esquematizar la interacción del usuario con la solución, mostrando las opciones del menú y las respuestas del sistema.

**🔹 Requisitos:  
✅** Crear un boceto o diagrama que represente el flujo de interacción del usuario con el sistema.  
✅ Indicar qué opciones estarán disponibles en el menú y qué sucede al seleccionar cada una.  
✅ Diferenciar los datos ingresados por el usuario (color verde) de los datos generados por el sistema (color negro).

**📢 Nota:** Se puede realizar a mano y escanearlo, o utilizar herramientas digitales como Figma, Canva, PowerPoint o Draw.io.

**📢 Nota:** Elimine los datos del ejemplo y sustituya la información con los elementos específicos de su proyecto.

Sistema de cálculo de IMC

Ingresar el peso: 85

Ingresar la altura: 1,65

El IMC es: 31.22130394857668

Riesgo de sobrepeso

¿Hay más clientes?: sí

Ingresar el peso: 60

Ingresar la altura: 1,72

El IMC es: 20.281233098972418

Peso saludable

¿Hay más clientes?: no

1. **Validación de la Solución**
   1. **Prueba de escritorio**

📌 **Objetivo:** Evaluar la lógica del diagrama de flujo mediante la simulación manual del programa con diferentes valores de entrada.

**🔹 Requisitos:**  
✅ Se deben probar mínimo 3 casos diferentes.  
✅ Comparar la salida esperada con la salida obtenida y verificar si coinciden.

**📢 Nota:** La tabla incluye ejemplos de referencia. Antes de completarla, elimine los ejemplos y registre los datos de su propia prueba de escritorio.

**📢 Nota:** Elimine los datos del ejemplo y sustituya la información con los elementos específicos de su proyecto.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Iteración** | **Opción Seleccionada** | **Entrada de Datos** | **Estado inicial**  **de las variables** | **Acción Realizada** | **Estado variables después de la Acción** |
| 1 | 1 (Consultar) | N/A | Producto A: 10, 20.5; Producto B: 5, 35.0; Producto C: 20, 15.0 | Mostrar inventario | Producto A: 10, 20.5; Producto B: 5, 35.0; Producto C: 20, 15.0 |
| 2 | 2 (Actualizar) | ID=1, Cantidad=15 | Producto A: 10, 20.5; Producto B: 5, 35.0; Producto C: 20, 15.0 | Actualizar cantidad del Producto A | Producto A: 15, 20.5; Producto B: 5, 35.0; Producto C: 20, 15.0 |
| 3 | 2 (Actualizar) | ID=2, Precio=40.0 | Producto A: 15, 20.5; Producto B: 5, 35.0; Producto C: 20, 15.0 | Actualizar precio del Producto B | Producto A: 15, 20.5; Producto B: 5, 40.0; Producto C: 20, 15.0 |
| 4 | 3 (Eliminar) | ID=1 | Producto A: 15, 20.5; Producto B: 5, 40.0; Producto C: 20, 15.0 | Eliminar Producto A | Producto B: 5, 40.0; Producto C: 20, 15.0 |
| 5 | 4 (Salir) | N/A | Producto B: 5, 40.0; Producto C: 20, 15.0 | Terminar programa | (El programa termina) |
| 6 |  |  |  |  |  |
| 7 |  |  |  |  |  |
| 8 |  |  |  |  |  |

***Descripción de cada columna de la tabla de pruebas de escritorio***



* 1. **Implementación Inicial**

📌 **Objetivo:** Iniciar la implementación en **Python**, creando la estructura base del programa con un **menú interactivo** y al menos **dos funciones operativas**.

🔹 **Requisitos:**  
✅ Implementar el **menú principal** con las opciones de interacción del usuario.  
✅ Desarrollar **dos funciones básicas** del sistema (por ejemplo, inicialización de datos y consulta de información).  
✅ El código debe estar **estructurado y tener comentarios claros**, siguiendo buenas prácticas de programación (por ejemplo, nombramiento de variables).

📢 **Nota:** Se recomienda probar el código y asegurarse de que las funciones desarrolladas operen correctamente antes de la siguiente entrega.

*Coloque aquí el enlace Google Colab al código*

*Coloque aquí el código inicial del proyecto*