**INTERFACES Y MULTIMEDIA**

**DESARROLLO DE UNA APLICACIÓN INTERACTIVA PARA ENSEÑAR ENSAMBLAJE Y CONFIGURACIÓN BÁSICA DE COMPUTADORAS PERSONALES**

**Estudiantes:**

**Baquero López Fabricio Fernando**

**Bravo Macias Javier Neicer**

**Docente:**

**Germán Ñacato Caiza**

**NRC: 1371**

# **Fecha: 25/11/2024**

# **Tema:** Desarrollo de una aplicación interactiva para enseñar ensamblaje y configuración básica de computadoras personales

**Objetivo**: Desarrollar una aplicación interactiva que facilite la enseñanza del ensamblaje y la configuración básica de computadoras personales, proporcionando una experiencia práctica, intuitiva y accesible para usuarios principiantes, con el fin de fortalecer sus habilidades técnicas y fomentar el aprendizaje autónomo mediante simulaciones realistas y recursos educativos complementarios.

1. **Computadoras**

**1.1 Introducción:**

Las computadoras son dispositivos electrónicos diseñados para procesar, almacenar y compartir información mediante hardware y software. Están presentes en múltiples campos como la educación, el trabajo y el entretenimiento.

**1.2 Componentes principales**

Los componentes internos permiten que una computadora funcione de manera eficiente:

**1.2.1** **Tarjeta madre**: Base donde se conectan todos los componentes principales, como procesador, memoria y dispositivos de almacenamiento.

**1.2.2 Procesador (CPU):** Es el cerebro de la computadora, encargado de realizar cálculos y ejecutar instrucciones.

**1.2.3** **Memoria RAM:** Almacén temporal de datos que permite al procesador acceder rápidamente a la información necesaria para las tareas activas.

**1.3 Periféricos básicos**

**Elementos externos que facilitan la interacción usuario-computadora:**

**1.3.1 Entrada:** Permiten introducir datos (teclado, mouse, escáner).

**1.3.2 Salida**: Muestran resultados procesados (monitor, impresora).

1. **Arquitectura del Computador**

**2.1 Entradas:** Dispositivos que envían datos a la computadora para ser procesados, como teclados y micrófonos.

**2.2 Procesamiento:** Se realiza en la CPU, que ejecuta operaciones aritméticas, lógicas y de control basadas en los datos de entrada.

**2.3 Salidas:** Dispositivos que presentan la información procesada al usuario, como pantallas o altavoces.

1. **Mantenimiento**

**3.1 Tipos de Mantenimiento**

**3.1.1 Preventivo**: Acciones regulares para evitar problemas (limpieza, actualización de software).

**3.1.2 Correctivo:** Reparación de fallas o reemplazo de componentes defectuosos.

**3.2 Herramientas esenciales:** Kit de desarmadores, pinzas, alcohol isopropílico, brochas antiestáticas y software de diagnóstico.

**3.3 Prácticas recomendadas:**

* Mantener el equipo limpio y libre de polvo.
* Realizar backups regulares.
* Actualizar hardware y software periódicamente.

1. **Ensamblaje y Configuración**

**4.1 Ensamblaje paso a paso:**

* Instalar la CPU en la tarjeta madre.
* Montar la tarjeta madre en el gabinete.
* Colocar la RAM, el disco duro y la fuente de poder.
* Conectar cables y periféricos.

**4.2 Configuración de hardware y BIOS:**

* Verificar que los dispositivos estén correctamente reconocidos.
* Ajustar el orden de inicio y los parámetros del sistema.

**4.3 Instalación del sistema operativo:**

Seleccionar un sistema operativo adecuado.

Configurar particiones e instalar los controladores necesarios.

1. **Simulaciones Interactivas**

**5.1 Simulaciones de ensamblaje virtual**

Software que permite practicar el ensamblaje sin necesidad de hardware físico.

**5.2 Conexión y pruebas de hardware**

Simuladores que guían al usuario para identificar y conectar correctamente los componentes.

**5.3 Retroalimentación personalizada**

Herramientas que analizan el desempeño del usuario durante las simulaciones y ofrecen sugerencias de mejora.

1**. Existen 3 tipos de interfaz de usuario según su diseño y propósito: software, hardware y software y hardware. En base a su tema de aplicación explique los tipos de interfaz que utilizará en su aplicación.**

Para el Desarrollo de una aplicación interactiva para enseñar ensamblaje y configuración básica de computadoras personales, los tipos de interfaces a utilizar son:

1. **Interfaz de software:**

Está diseñada para generar facilidad en la interacción con el usuario mediante gráficos, menús y simulaciones. Se incluirán diagramas interactivos y simulaciones virtuales para mostrar el ensamblaje y configuración de componentes de una PC. Todo esto para proveer un entorno de fácil uso y que sea intuitivo para aprender los conceptos básicos.

1. **Interfaz de hardware y software combinados:**

Al utilizar la interfaz de hardware y software se realiza combinación de simulaciones virtuales con el hardware físico interactivo. Implementando kits educativos que conecten software de simulación con dispositivos físicos, un microcontrolador que representen el funcionamiento interno de una PC. Con la finalidad de brindar una experiencia de aprendizaje integral que conecte la teoría con la prácticas

Estas interfaces mencionadas permitirán una experiencia didáctica efectiva, combinando el aprendizaje interactivo virtual y la práctica tangible.

**2. En base a su tema de aplicación en desarrollo explique y justifique para que perfil de usuario será desarrollará su aplicación.**

| **Perfil de usuario** | **Justificación** |
| --- | --- |
| 1. Estudiantes de informática, tecnología, etc. | Los estudiantes suelen necesitar una base sólida en hardware para comprender cómo interactúan los componentes de una computadora.  Este público frecuentemente requiere herramientas prácticas para complementar su aprendizaje teórico.  La aplicación les permitirá experimentar en un entorno controlado antes de enfrentarse a equipos reales, reduciendo riesgos de errores costosos. |
| 2. Técnicos principiantes o en formación en soporte técnico | Técnicos que inician su carrera necesitan desarrollar habilidades prácticas para el diagnóstico, ensamblaje y configuración de equipos.  La aplicación puede servir como una herramienta de capacitación que simula escenarios reales, facilitando la práctica sin necesidad de tener acceso constante a hardware físico. |
| 3. Educadores y capacitadores | Profesores o instructores que enseñan sobre hardware pueden usar la aplicación como un recurso didáctico para complementar sus clases.  Proveerá un entorno virtual donde los estudiantes puedan practicar fuera del aula, mejorando el aprendizaje mediante la experimentación. |

La aplicación se desarrollará para un perfil de usuario con conocimientos básicos o nulos en hardware. Su diseño interactivo permitirá atender diferentes niveles de aprendizaje, desde principiantes hasta técnicos en formación, fomentando una experiencia educativa práctica, segura y accesible.

**3. En base a su tema de aplicación documentar la información (ETAPAS, ACTIVIDADES, etc.) tomando como referencia el ciclo de desarrollo en el DISEÑO CENTRADO EN EL USUARIO. En cada una de las fases se debe generar la información en base a su tema, y utilizar las herramientas de software que esté a su alcance. Para su revisión se puede copiar el link en donde se encuentre almacenado los archivos correspondientes generado en la herramienta de software utilizado.**

El Diseño Centrado en el Usuario (DCU) se enfoca en crear productos que satisfagan las necesidades, expectativas y habilidades de los usuarios finales. A continuación, se describen las fases, actividades y herramientas aplicadas para desarrollar la aplicación interactiva sobre ensamblaje y configuración de computadoras personales:

|  | **OBJETIVO** | **ACTIVIDADES** | **HERRAMIENTAS** |
| --- | --- | --- | --- |
| **Investigación y Análisis del Usuario** | Comprender las necesidades, habilidades, limitaciones y objetivos del usuario final. | * **Definición del perfil de usuario:** Identificación de los usuarios objetivo (principiantes, estudiantes de informática, técnicos en formación). * **Encuestas y entrevistas:** Recolectar información sobre el nivel de conocimiento previo, intereses y desafíos comunes en el ensamblaje de computadoras. * **Análisis de tareas:** Identificar las principales tareas relacionadas con el ensamblaje y configuración, como identificar componentes, ensamblar físicamente una PC y configurar el sistema operativo. | * **Google Forms** para encuestas. * **Excel/Google Sheets** para organizar y analizar datos. * **Trello** para documentar hallazgos clave. |
| **Diseño de Requisitos** | Definir las características y funcionalidades de la aplicación según las necesidades del usuario. | * **Redacción de requisitos funcionales:** * Interfaz gráfica intuitiva. * Simulación interactiva de ensamblaje. * Guías paso a paso con elementos visuales. * **Requisitos no funcionales:** * Compatibilidad multiplataforma (Windows, Android). * Respuesta rápida e interfaz fluida. * **Creación de historias de usuario:** Ejemplo: "Como estudiante, quiero aprender a ensamblar una PC con simulaciones para entender el proceso sin riesgo de dañar componentes reales". | * **Miro** para diagramas y mapeo de ideas. * **Jira/Trello** para gestionar tareas. |
| **Diseño Conceptual y Prototipado** | Crear prototipos iniciales para visualizar la aplicación antes del desarrollo. | * **Wireframes:** * Bocetos iniciales del diseño de las pantallas: Menú principal, selección de componentes, simulación de ensamblaje, etc. * **Prototipos interactivos:** * Desarrollo de prototipos navegables que representen la interacción esperada con la aplicación. | * **Figma** para wireframes y prototipos interactivos. * **Adobe XD** para diseño de interfaces. |
| **Evaluación del Diseño** | Probar los prototipos con usuarios reales para identificar mejoras antes de implementar. | * **Pruebas de usabilidad:** * Usuarios prueban el prototipo y realizan tareas específicas, como ensamblar una PC virtual. * **Recopilación de retroalimentación:** * Identificar problemas de navegación, comprensión o funcionalidad. * **Iteración del diseño:** * Ajustar el diseño según la retroalimentación recibida. | * **Maze** para pruebas de usabilidad. * **Google Docs** para recopilar comentarios. |
| **Desarrollo e Implementación** | Construir la aplicación basada en los diseños validados. | * **Desarrollo front-end:** * Creación de la interfaz gráfica. * **Desarrollo back-end:** * Gestión de simulaciones, base de datos de componentes y configuración del flujo de interacción. * **Integración de elementos multimedia:** * Videos tutoriales, animaciones y guías visuales. | * **Unity** para simulaciones interactiva. * **Visual Studio Code** para desarrollo. * **GitHub** para control de versiones. |
| **Pruebas y Evaluación Final** | Garantizar el funcionamiento óptimo de la aplicación. | * **Pruebas funcionales:** * Verificar que cada función (por ejemplo, la simulación del ensamblaje) opere correctamente. * **Pruebas de usabilidad:** * Asegurar que los usuarios logren completar las tareas sin dificultades. * **Pruebas de compatibilidad:** * Evaluar la aplicación en diferentes dispositivos y sistemas operativos. | * **Selenium** para pruebas automatizadas. * **BrowserStack** para pruebas multiplataforma. |

El desarrollo de esta aplicación con base en el DCU asegura que se adapta a las necesidades del usuario final, ofreciendo una herramienta práctica e interactiva que facilita el aprendizaje en ensamblaje y configuración de computadoras personales.

**4. En base a su tema de aplicación en desarrollo explique y justifique sobre el diseño adaptivo que tendrá su aplicación. Para su revisión se puede copiar el link en donde se encuentre almacenado los archivos correspondientes generado en la herramienta de software utilizado.**

### **Diseño Adaptivo de la Aplicación**

El diseño adaptivo de la aplicación se fundamenta en ofrecer una experiencia de usuario óptima al ajustarse automáticamente a diferentes dispositivos y resoluciones. Esto es esencial para garantizar que los usuarios puedan acceder y utilizar la aplicación en cualquier plataforma sin comprometer su funcionalidad o estética.

### **Justificación del Diseño Adaptivo**

1. **Accesibilidad Multiplataforma:**

* Los usuarios pueden utilizar dispositivos variados como teléfonos móviles, tablets, laptops y computadoras de escritorio.
* El diseño adaptivo asegura que la interfaz y las funciones sean intuitivas y funcionales, independientemente del dispositivo.

1. **Optimización de la Experiencia del Usuario:**

* En pantallas pequeñas, como teléfonos, se priorizarán elementos clave como los botones interactivos, texto legible y simulaciones ajustadas.
* En dispositivos más grandes, como laptops, se presentarán detalles adicionales, diagramas amplios y menús expandibles, maximizando el espacio disponible.

1. **Compatibilidad con Diferentes Resoluciones:**

* La diversidad en resoluciones y relaciones de aspecto, desde 720p en móviles hasta 4K en monitores avanzados, exige un diseño adaptable que mantenga la claridad y funcionalidad de los elementos visuales.

1. **Facilidad de Aprendizaje y Uso:**

* Los usuarios objetivo, especialmente los principiantes, necesitan una experiencia coherente y sin barreras. El diseño adaptivo minimiza problemas técnicos y asegura una curva de aprendizaje suave.

### **Características del Diseño Adaptivo**

1. **Diseño de Interfaz (UI):**

* **Elementos escalables:**Uso de unidades de medida relativas para asegurar que los componentes gráficos y textos se ajusten al tamaño de la pantalla.
* **Disposición fluida:**Reorganización automática de menús y elementos visuales mediante sistemas de cuadrícula y *breakpoints*.

1. **Navegación Adaptativa:**

* Menús deslizantes en dispositivos móviles.
* Menús completos con secciones visibles en pantallas grandes.

1. **Simulaciones y Gráficos Responsivos:**

* Simulaciones que ajusten su escala y controles según la resolución del dispositivo.
* Visualizaciones optimizadas para interactuar fácilmente en dispositivos táctiles.

1. **Optimización del Rendimiento:**

* Carga progresiva de recursos en dispositivos con capacidades limitadas.
* Reducción de elementos visuales complejos en pantallas más pequeñas para mejorar la velocidad de respuesta.

### **Tecnologías y Herramientas Utilizadas**

1. **Frameworks Responsivos:**

* **Bootstrap**: Para garantizar la adaptabilidad de la interfaz mediante rejillas flexibles y estilos predefinidos.
* **CSS Media Queries**: Para ajustar la presentación del contenido según el tamaño del dispositivo.

1. **Pruebas y Simulación:**

* **BrowserStack:** Verificación de la apariencia y funcionalidad en distintos dispositivos y navegadores.
* **Figma/Adobe XD:** Creación de prototipos adaptativos para distintas resoluciones.

1. **Desarrollo Técnico:**

* **React Native:** Para construir la aplicación con diseño adaptable y soporte multiplataforma (Android/iOS).
* **Unity:** Optimización de las simulaciones interactivas para diferentes tamaños y tipos de pantalla.

### **Implementación del Diseño Adaptivo**

El diseño se implementará progresivamente:

1. **Prototipos iniciales adaptativos:** Realizados en **Figma** para simular su comportamiento en dispositivos diversos.
2. **Desarrollo Front-End:** Configurado con frameworks como Bootstrap y React Native.
3. **Pruebas iterativas:** Simulación y corrección en dispositivos reales y virtuales.

El diseño adaptivo asegura que la aplicación sea accesible, funcional y visualmente atractiva en cualquier dispositivo. Esto no solo mejora la experiencia del usuario final, sino que también amplía el alcance y la utilidad de la herramienta. La documentación generada (prototipos y ejemplos adaptativos) puede almacenarse en plataformas como Google Drive o GitHub para revisión.

Si se requiere un enlace de ejemplos o diseño preliminar, puede desarrollarse y compartirse como referencia adicional.

**5. En base a su tema de aplicación en desarrollo realice la arquitectura de la información, diseño de interacción y los elementos de experiencia de usuario. Para su revisión se puede copiar el link en donde se encuentre almacenado los archivos correspondientes generado en la herramienta de software utilizado.**

### **Arquitectura de la Información**

La arquitectura de la información define cómo se estructuran y organizan los contenidos y funcionalidades dentro de la aplicación.

#### **Estructura Jerárquica**

* **Inicio**
  1. Presentación (Breve introducción sobre el objetivo de la aplicación).
  2. Menú Principal.
* **Secciones del Menú Principal**
  1. **Teoría Básica:**
     + Conceptos fundamentales.
     + Componentes principales de una computadora (CPU, GPU, RAM, etc.).
  2. **Simulación de Ensamblaje:**
     + Selección de componentes.
     + Ensamblaje paso a paso.
  3. **Configuración Básica del Sistema:**
     + Instalación del sistema operativo.
     + Configuración de BIOS y controladores.
  4. **Evaluaciones:**
     + Cuestionarios interactivos.
     + Simulaciones prácticas para medir el progreso.
  5. **Ayuda y Contacto:**
     + FAQs.
     + Contacto para soporte.

#### **Mapa de Navegación**

Se desarrollará un mapa interactivo usando herramientas como **Figma** para representar las conexiones entre las secciones. Esto asegura que cada contenido sea accesible mediante una estructura lógica y sencilla.

### **2. Diseño de Interacción**

El diseño de interacción asegura que los usuarios puedan realizar acciones intuitivamente y obtener respuestas inmediatas de la aplicación.

#### **Principios Clave del Diseño**

1. **Consistencia:**
   * Uso de colores, botones y patrones uniformes en todas las pantallas.
2. **Acción Contextual:**
   * Botones destacados para "Siguiente paso" o "Intentar de nuevo" en simulaciones.
3. **Retroalimentación Inmediata:**
   * Mensajes y animaciones que confirmen cuando un usuario realiza una acción correctamente o comete un error.

#### **Flujo de Interacción Ejemplo (Simulación de Ensamblaje):**

1. **Inicio de la Simulación:**
   * El usuario selecciona la opción "Simulación de Ensamblaje".
   * Aparece una pantalla con un gabinete vacío y una lista de componentes.
2. **Selección de Componentes:**
   * El usuario arrastra y suelta los componentes en las ubicaciones correspondientes.
   * Retroalimentación visual: Indicador verde (correcto) o rojo (incorrecto).
3. **Finalización:**
   * Una vez completado, se muestra un resumen con las áreas que el usuario realizó correctamente y sugerencias de mejora.

#### **Herramientas Utilizadas**

* **Figma:** Para diseñar prototipos de flujo interactivo.
* **Unity:** Para crear simulaciones dinámicas e interactivas.

### **3. Elementos de Experiencia de Usuario (UX)**

La experiencia de usuario considera cómo se siente el usuario al interactuar con la aplicación, asegurando satisfacción y facilidad de uso.

#### **Elementos Clave de UX**

1. **Usabilidad:**
   * Interfaz intuitiva con íconos reconocibles y descripciones claras.
   * Navegación sencilla con accesos directos en todas las pantallas.
2. **Accesibilidad:**
   * Compatibilidad con dispositivos móviles y de escritorio.
   * Texto ajustable para usuarios con dificultades visuales.
3. **Diseño Visual:**
   * Colores suaves y elementos visuales que no distraigan de las tareas principales.
   * Animaciones ligeras que refuercen las acciones del usuario.
4. **Engagement (Compromiso):**
   * Logros desbloqueables al completar simulaciones o evaluaciones.
   * Puntos o recompensas virtuales para fomentar la práctica constante.
5. **Pruebas de Usuario:**
   * Evaluaciones periódicas con usuarios reales para ajustar el diseño según su retroalimentación.

#### **Herramientas para UX:**

* **Adobe XD:** Diseño visual y prototipado.
* **Maze:** Pruebas de usabilidad.
* **Hotjar:** Análisis de comportamiento de usuarios en la aplicación.

La combinación de una arquitectura de la información clara, un diseño de interacción intuitivo y elementos centrados en la experiencia del usuario garantizará que la aplicación sea funcional, atractiva y efectiva para el aprendizaje. Con un enfoque iterativo, el diseño podrá adaptarse a las necesidades reales de los usuarios.



**6. En base a su tema de aplicación en desarrollo realice en forma detallada las formas de interactividad de su aplicación. Para su revisión se puede copiar el link en donde se encuentre almacenado los archivos correspondientes generado en la herramienta de software utilizado.**

### **1. Interactividad Visual e Interactiva**

* **Arrastrar y Soltar Componentes:**Los usuarios pueden interactuar con una interfaz gráfica donde arrastren y coloquen componentes (procesadores, memorias RAM, tarjetas madre, etc.) en sus ubicaciones correctas dentro de un gabinete virtual.
* **Zoom y Rotación 3D:**Permitir la exploración visual de los componentes desde diferentes ángulos para un mejor entendimiento de su estructura y posición en un ensamblaje.
* **Indicaciones Paso a Paso:**La aplicación incluirá guías interactivas que, paso a paso, expliquen las acciones necesarias para ensamblar cada componente. Estas guías se activarán cuando el usuario selecciona una parte.

### **2. Evaluación de Conocimientos**

* **Cuestionarios Interactivos:**Al finalizar una etapa del ensamblaje, se presentarán preguntas relacionadas con los componentes y su funcionalidad. Las respuestas incorrectas activarán explicaciones detalladas.
* **Modo Desafío:**Un modo en el que los usuarios ensamblen un equipo completo sin asistencia, y la aplicación evalúe la precisión y el tiempo de ejecución.

### **3. Simulación y Configuración**

* **Simulador de BIOS/UEFI:**Una interfaz simulada para configurar opciones básicas del sistema, como el orden de arranque, habilitación de hardware y monitoreo de temperatura.
* **Pruebas de Hardware Virtuales:**Una herramienta para verificar si los componentes ensamblados funcionarían correctamente, señalando posibles errores como incompatibilidades.

### **4. Accesibilidad y Personalización**

* **Lenguaje Multilenguaje:**Los usuarios podrán seleccionar entre distintos idiomas para una mayor accesibilidad.
* **Interactividad por Medio de Gestos (Opcional):**Para dispositivos con pantallas táctiles, se pueden incluir gestos como "pellizcar para hacer zoom" o "deslizar para cambiar componentes".
* **Opciones de Nivel de Dificultad:**Un ajuste que permite al usuario decidir cuánta ayuda recibir (desde tutoriales detallados hasta ensamblajes totalmente autónomos).

### **5. Compatibilidad y Colaboración**

* **Módulo de Consulta:**Los usuarios podrán buscar información adicional sobre cualquier componente directamente desde la aplicación.
* **Modo Multiusuario:**Una opción para colaborar en tiempo real con otros usuarios, ideal para clases grupales o entrenamientos.

### **6. Generación de Reportes**

* **Reporte de Progreso:**La aplicación generará un informe que detalle las actividades completadas, puntuaciones en evaluaciones y tiempo total invertido.
* **Exportación de Resultados:**Los usuarios podrán guardar o compartir sus resultados y configuraciones por medio de archivos PDF o enlaces generados.

**7. En base a su tema de aplicación en desarrollo explique las figuras retoricas que se utilizará en su aplicación. Para su revisión se puede copiar el link en donde se encuentre almacenado los archivos correspondientes generado en la herramienta de software utilizado.**

1. **Metáfora**

* **Propósito:** Facilitar la comprensión de conceptos técnicos al asociarlos con objetos o acciones cotidianas.

En las secciones como **Memoria RAM** o **Tarjeta Madre**, se utilizan explicaciones teóricas que podrían incluir metáforas simples. Por ejemplo, describir la memoria RAM como "el escritorio temporal donde se colocan los datos que necesitas ahora" o la tarjeta madre como "la columna vertebral que conecta todos los componentes de la computadora."

1. **Metonimia**

* **Propósito:** Usar íconos simplificados para que los usuarios identifiquen rápidamente herramientas o funciones.

Las imágenes ilustrativas de componentes como la memoria RAM o el procesador ayudan a que los usuarios asocien de manera directa cada texto teórico con el hardware correspondiente.

1. **Sinécdoque**

* **Propósito:** Representar partes clave de un proceso para simplificar el aprendizaje.

1. **Hipérbole**

* **Propósito:** Enfatizar acciones o errores críticos de forma llamativa.

1. **Personificación**

* **Propósito:** Hacer más amigable el aprendizaje a través de personajes animados.

1. **Elipsis**

* **Propósito:** Agilizar el proceso y mantener la atención del usuario eliminando pasos intermedios que no sean esenciales.

**8. Utilizando una herramienta tecnología realice las retículas en la pantalla de su tema de aplicación en desarrollo. Adjunte a la guía los respectivos archivos. Para su revisión se puede copiar el link en donde se encuentre almacenado los archivos correspondientes generado en la herramienta de software utilizado.**

****

**Link de interfaz:** [**https://www.figma.com/proto/hUpLLXgJCvLa4RyEKUcfdv/Untitled?node-id=0-1&t=EIt22Kthb4focEd9-1**](https://www.figma.com/proto/hUpLLXgJCvLa4RyEKUcfdv/Untitled?node-id=0-1&t=EIt22Kthb4focEd9-1)

**REFERENCIAS BIBLIOGRÁFICAS:**

1. **G. Sevilla. (2008). DESARROLLO DE UN TUTORIAL HIPERMEDIA PARA LA ENSEÑANZA DE ENSAMBLAJE DE COMPUTADORES PERSONALES A NIVEL BÁSICO – PARA EL SECAP. Escuela Politécnica Nacional.** [**https://bibdigital.epn.edu.ec/bitstream/15000/945/1/CD-1403.pdf**](https://bibdigital.epn.edu.ec/bitstream/15000/945/1/CD-1403.pdf)
2. **Universidad Politécnica Salesiana. (2024). Ensamblaje y Mantenimiento de Computadores Personales. UPS.edu.ec** [**https://www.ups.edu.ec/documents/10184/5310383/E108UIO-Curso+Ensamblaje+y+Mantenimiento+de+Computadores+Personales.pdf/af88da3a-f75b-4146-874c-34b6f7ca11c9**](https://www.ups.edu.ec/documents/10184/5310383/E108UIO-Curso+Ensamblaje+y+Mantenimiento+de+Computadores+Personales.pdf/af88da3a-f75b-4146-874c-34b6f7ca11c9)
3. **J. Portugal. (2015). DESARROLLO DE UNA APLICACIÓN PARA DISPOSITIVOS MÓVILES, COMO GUÍA DE CONOCIMIENTOS BÁSICOS, DE ENSAMBLAJE Y SOLUCIÓN DE PROBLEMAS TÉCNICOS COMUNES DE LOS COMPONENTES INTERNOS DE UNA PC, DIRIGIDA A PERSONAS NO INFORMÁTICAS. Universidad Católica Santiago de Guayaquil.** [**http://repositorio.ucsg.edu.ec/bitstream/3317/4398/1/T-UCSG-PRE-ART-IPM-84.pdf**](http://repositorio.ucsg.edu.ec/bitstream/3317/4398/1/T-UCSG-PRE-ART-IPM-84.pdf)
4. **J. Basogoita. (2024). Manten tu pc: Enséñales a armar, mantener y configurar. Edukalab.** [**https://edtk.co/p/88420**](https://edtk.co/p/88420)

**Formato de entrega del archivo**

La guía debe ser entregada en formato .doc y en caso de que las preguntas hagan referencia a la utilización de herramientas tecnológicas debe adjuntar todos los archivos

**Nombre del archivo**

A#.Apellido.Nombre.Asignatura

**Ejemplo**: A1.Castro.Jorge.Interfaces.Multimedia

**Criterios de evaluación**

| Pregunta | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | Total |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Puntaje | 2 | 2 | 3 | 3 | 3 | 3 | 2 | 2 | 20 |