

Fundamentos de hardware y software

Tecnología e Informática | Pensamiento Computacional

Descripción del Curso

Este curso de Pensamiento Computacional está dirigido a estudiantes a partir de los 17 años, sin límite superior de edad. Su objetivo es desarrollar de manera integral la capacidad de pensar de forma estructurada para resolver problemas reales, trasladando estrategias computacionales a situaciones cotidianas y diversas áreas del saber. El curso propone un aprendizaje activo, basado en proyectos, exploración guiada y reflexión crítica sobre el impacto de la tecnología en la vida diaria y la sociedad. Objetivo general: Desarrollar en los estudiantes habilidades de pensamiento computacional para abordar problemas complejos mediante la descomposición, la abstracción, el reconocimiento de patrones y el diseño de soluciones algorítmicas, con énfasis en la creatividad, la colaboración y la ciudadanía digital. Específicos: - Comprender y aplicar conceptos clave del pensamiento computacional y su relación con situaciones reales. - Desarrollar la capacidad de descomposición de problemas, identificación de patrones y modelado de soluciones. - Diseñar, analizar y comunicar algoritmos y procesos de solución de forma clara y justificable. - Implementar prototipos simples y evaluarlos en contextos prácticos, considerando criterios de eficiencia y seguridad. - Trabajar de forma colaborativa, gestionar proyectos y reflexionar críticamente sobre el uso ético de la tecnología. - Transferir las habilidades aprendidas a distintos campos y situaciones de la vida diaria, académica o profesional. La estructura del curso se apoya en 4 unidades interrelacionadas: Unidad 1 (Fundamentos del pensamiento computacional), Unidad 2 (Descomposición, patrones y abstracción), Unidad 3 (Algoritmos, diagramas y resolución de problemas) y Unidad 4 (Proyectos y aplicaciones en contextos reales). Cada unidad combina teoría breve, práctica guiada y proyectos que requieren comunicar resultados de forma clara y con evidencia de razonamiento lógico. Al finalizar, se espera que el estudiante haya desarrollado un portafolio de soluciones y una reflexión personal sobre el uso responsable de las tecnologías.

Competencias

- Aplicar el pensamiento computacional para analizar problemas y diseñar soluciones efectivas en contextos reales.
- Descomponer problemas complejos en componentes manejables y modelarlos mediante estructuras simples.
- Reconocer patrones, abstraer información relevante y generalizar soluciones a nuevos escenarios.
- Diseñar, expresar y comunicar algoritmos y procesos de solución de forma clara y razonable.
- Desarrollar proyectos de forma colaborativa, gestionando tareas, tiempos y responsabilidades.
- Emplear herramientas digitales de manera ética, segura y responsable, respetando la propiedad intelectual y la privacidad.
- Aplicar el pensamiento computacional en áreas diversas, fomentando la creatividad y la resolución de problemas multidisciplinarios.

Requerimientos

- Computadora o dispositivo con conexión a internet estable y acceso a herramientas de programación educativa.
- Navegador moderno y capacidad para usar entornos de desarrollo en línea o instalar software básico recomendado por el curso.
- Cuenta en la plataforma de aprendizaje utilizada por el curso y acceso al correo institucional para comunicaciones oficiales.
- Material de apoyo proporcionado por el docente (guías, lecturas y cuaderno de trabajos) y espacio para organización de tareas (físico o digital).
- Compromiso de asistencia, participación activa, entrega puntual de tareas y trabajo colaborativo.
- Conocimientos básicos de lectura y escritura en español y habilidades mínimas de gestión de información digital.

Unidades del Curso

Unidad 1: Unidad 1: Introducción a hardware y software

Objetivos de Aprendizaje

- Definir hardware y software y explicar su relación en un sistema informático.
- Explicar cómo el software utiliza el hardware para ejecutar programas y tareas.
- Diferenciar componentes de entrada, procesamiento y salida y dar ejemplos de cada uno.

Contenidos Temáticos

1. Arquitectura general de un sistema informático — breve descripción
 1. Definición de hardware y software.
 2. Relación entre el software y el hardware para realizar tareas.
2. Tipos de software — breve descripción
 1. Software de sistema vs software de aplicación.
 2. Papel del sistema operativo en la gestión de recursos.
3. Interacciones básicas entre usuario y equipo — breve descripción
 1. Encendido y ejecución de un programa sencillo.
 2. Interfaz de usuario y flujo básico de uso.

Actividades

- **Actividad 1: Debate guiado - ¿Qué hace un ordenador?** Discusión en grupo sobre el rol del hardware y del software y ejemplos de tareas cotidianas. Puntos clave: el hardware realiza operaciones físicas y el software da instrucciones; aprendizajes: entender la interacción entre ambos.

- **Actividad 2: Mapa conceptual colaborativo** Crear en equipo un diagrama que relacione hardware y software con ejemplos, utilizando tarjetas o herramientas digitales. Puntos clave: categorías de hardware, tipos de software; aprendizajes: organizar conceptos y sus relaciones.
- **Actividad 3: Flujo de ejecución simple** Dibujar o simular el flujo de un programa desde el clic inicial hasta obtener el resultado. Puntos clave: entrada, procesamiento y salida; aprendizajes: entender secuencias de ejecución.
- **Actividad 4: Observación de componentes de hardware** Identificar físicamente componentes en un equipo real o simulado y anotar funciones básicas. Puntos clave: CPU, RAM, almacenamiento; aprendizajes: reconocimiento y función general.

Evaluación

- Evaluación del Objetivo General: participación y aportes en el debate y las actividades de clase.
- Evaluación de Objetivos Específicos: preguntas cortas de definición y clasificación, y un mini cuestionario de reconocimiento de componentes.

Unidad 2: Unidad 2: Componentes de hardware y su función

Objetivos de Aprendizaje

- Identificar la CPU, la memoria (RAM), el almacenamiento y la placa base, y explicar su función básica.
- Reconocer la función de la GPU y de los dispositivos de entrada/salida en el procesamiento de información.
- Establecer relaciones entre capacidad, velocidad y rendimiento del sistema.

Contenidos Temáticos

1. CPU y rendimiento — breve descripción
 1. Función de la CPU como unidad de procesamiento central.
 2. Frecuencia, núcleos y rendimiento básico.
2. Memoria RAM y almacenamiento — breve descripción
 1. Diferencias entre RAM y almacenamiento (HDD/SSD).
 2. Impacto de la memoria en la multitarea y en la velocidad de carga.
3. Periféricos de entrada y salida — breve descripción
 1. Ejemplos de dispositivos de entrada y de salida.
 2. Rol en la interacción con el usuario y con el sistema.
4. Placa base y buses — breve descripción
 1. Funciones básicas de la placa base.
 2. Conectores y buses que permiten la comunicación entre componentes.

Actividades

- **Actividad 1: Análisis de diagramas de arquitectura** Estudio de diagramas simplificados para identificar CPU, RAM, almacenamiento y buses. Puntos clave: ubicación y función de cada componente; aprendizajes: lectura de diagramas y relación entre piezas.
- **Actividad 2: Comparación de rendimiento** Analizar ejemplos de escenarios con diferentes cantidades de RAM y velocidades de disco para justificar el rendimiento. Puntos clave: impacto de la memoria y del almacenamiento; aprendizajes: correlación entre especificaciones y experiencia de usuario.
- **Actividad 3: Identificación de periferia** Lista de dispositivos de entrada/salida y su uso práctico en tareas diarias. Puntos clave: roles y ejemplos; aprendizajes: clasificación de periféricos.
- **Actividad 4: Montaje conceptual de un PC** Crear un diagrama o maqueta simple que muestre cómo encajan CPU, RAM, almacenamiento y placa base en una configuración típica. Puntos clave: conectores y compatibilidad; aprendizajes: visión de conjunto de un sistema.

Evaluación

- Evaluación del Objetivo General: cuestionario corto de correspondencia entre componentes y funciones.
- Evaluación de Objetivos Específicos: actividad de identificación de piezas en un diagrama y explicación de su función con ejemplos prácticos.

Unidad 3: Unidad 3: Software y sistemas operativos

Objetivos de Aprendizaje

- Diferenciar software de sistema y software de aplicación, y explicar sus roles.
- Describir las funciones principales de un sistema operativo (gestión de recursos, interfaz, seguridad).
- Reconocer conceptos básicos de drivers, compatibilidad y actualizaciones de software.

Contenidos Temáticos

1. Tipos de software — breve descripción
 1. Software de sistema vs software de aplicación.
 2. Ejemplos y funciones típicas de cada tipo.
2. Sistemas operativos y gestión de recursos — breve descripción
 1. Gestión de procesos y memoria.
 2. Interfaz de usuario y seguridad básica.
3. Instalación y actualización de software — breve descripción
 1. Procedimiento básico de instalación.
 2. Importancia de actualizaciones y control de versiones.

4. Drivers y compatibilidad — breve descripción

1. Qué son drivers y por qué son necesarios.
2. Consideraciones de compatibilidad entre hardware y software.

Actividades

- **Actividad 1: Clasificación de software** Clasificar ejemplos en software de sistema y de aplicación, justificando la elección. Puntos clave: funciones típicas; aprendizajes: clasificación y ejemplos reales.
- **Actividad 2: Explorando un sistema operativo** Análisis guiado de funciones básicas de un sistema operativo (gestión de ventanas, tareas, seguridad). Puntos clave: interacciones de usuario; aprendizajes: comprensión de la interfaz y la gestión de recursos.
- **Actividad 3: Instalación simulada de software** Simulación de instalación de una aplicación y discusión de pasos y requerimientos. Puntos clave: dependencias, permisos y actualizaciones; aprendizajes: flujo de instalación y verificación de éxito.
- **Actividad 4: Drivers y compatibilidad** Estudio de casos simples sobre la necesidad de drivers y cómo la compatibilidad influye en el rendimiento. Puntos clave: actualización de drivers; aprendizajes: importancia de compatibilidad.

Evaluación

- Evaluación del Objetivo General: tareas de clasificación y un cuestionario corto sobre conceptos de software y sistema operativo.
- Evaluación de Objetivos Específicos: actividad de correspondencia entre funciones del SO y ejemplos de software; verificación de comprensión de drivers y actualizaciones.

Unidad 4: Unidad 4: Seguridad, mantenimiento y resolución de problemas básicos

Objetivos de Aprendizaje

- Identificar riesgos comunes de seguridad y medidas preventivas (actualizaciones, contraseñas, respaldo de datos).
- Describir procedimientos básicos de mantenimiento (limpieza, actualizaciones, antivirus).
- Aplicar estrategias de resolución de problemas simples de hardware y software mediante pasos sistemáticos.

Contenidos Temáticos

1. Seguridad y actualizaciones — breve descripción

1. Importancia de actualizaciones de software y firmware.
2. Buenas prácticas de seguridad (contraseñas, antivirus, control de acceso).

2. Mantenimiento básico — breve descripción

1. Limpieza física y limpieza digital (archivos basura, optimización básica).

2. Plan de mantenimiento periódico y verificación de configuraciones.
3. Resolución de problemas comunes — breve descripción
 1. Guía de pasos: identificar síntomas, aislar causas, proponer soluciones.
 2. Estrategias simples de diagnóstico sin herramientas avanzadas.
4. Copias de seguridad y recuperación — breve descripción
 1. Importancia de respaldos; niveles de copia de seguridad (local, en la nube).
 2. Procedimientos básicos de recuperación ante fallo.

Actividades

- **Actividad 1: Simulación de actualizaciones y seguridad** Identificar zonas de riesgo en un escenario simulado y proponer medidas de mitigación. Puntos clave: parches de seguridad y contraseñas robustas; aprendizajes: cultura de seguridad.
- **Actividad 2: Mantenimiento preventivo** Plan de mantenimiento de un equipo ficticio con calendario y tareas. Puntos clave: limpieza física y optimización; aprendizajes: hábitos de mantenimiento.
- **Actividad 3: Resolución de problemas guiada** Estudio de casos simples (p. ej., equipo que no enciende, programa que no se ejecuta) y aplicación de un protocolo de diagnóstico. Puntos clave: pasos de diagnóstico; aprendizajes: enfoque estructurado para la solución de problemas.
- **Actividad 4: Copias de seguridad y restauración** Simulación de crear y restaurar una copia de seguridad, discutiendo diferentes métodos y herramientas. Puntos clave: importancia de respaldos; aprendizajes: planes de recuperación.

Evaluación

- Evaluación del Objetivo General: desempeño en la realización de un plan de mantenimiento y en la simulación de respuestas ante fallos.
- Evaluación de Objetivos Específicos: ejercicios prácticos de seguridad, mantenimiento y restauración de datos, con rúbricas de precisión y claridad en el procedimiento.