

Innovando con Mecatrónica: Proyectos Creativos de Robótica y Programación para Secundaria

Tecnología e Informática | Pensamiento Computacional | para estudiantes de secundaria (12-15 años) | 8 semanas

Descripción del Curso

Este curso está diseñado para estudiantes de secundaria (12-15 años) interesados en explorar el fascinante mundo de la mecatrónica mediante proyectos prácticos que integran robótica, programación y electrónica básica. A lo largo de 8 semanas, los alumnos desarrollarán habilidades en programación por bloques utilizando Scratch, Micro:bit, Arduino y entornos como Micronit y Minecraft, aplicando conceptos fundamentales de electricidad y mecánica para construir prototipos funcionales.

El curso se enfoca en un aprendizaje activo y basado en proyectos, promoviendo la creatividad, el pensamiento computacional y el trabajo colaborativo. Los estudiantes aprenderán a diseñar, programar y ensamblar mecanismos y máquinas simples, entendiendo cómo interactúan los componentes electrónicos y mecánicos para resolver problemas reales de manera innovadora.

Al finalizar, los participantes serán capaces de crear un proyecto mecatrónico completo que demuestre la integración de robótica y programación, fortaleciendo competencias tecnológicas esenciales para su formación académica y futura inserción en áreas STEM.

Objetivos Generales

- Diseñar y programar prototipos mecatrónicos que integren elementos de robótica, electrónica y programación por bloques.
- Aplicar conceptos fundamentales de electricidad y mecánica para construir mecanismos y máquinas simples funcionales.
- Utilizar plataformas digitales como Scratch, Micronit, Minecraft, Micro:bit y Arduino para el desarrollo de proyectos tecnológicos.
- Analizar y resolver problemas mediante la implementación de soluciones tecnológicas creativas.
- Presentar y documentar proyectos tecnológicos explicando su funcionamiento, diseño y aplicación.

Competencias

- Desarrollar habilidades de programación en bloques para controlar dispositivos electrónicos como Micro:bit y Arduino.
- Aplicar principios básicos de electricidad y mecánica para diseñar y construir prototipos funcionales.
- Integrar conceptos de robótica y electrónica en proyectos creativos y colaborativos.

- Analizar y solucionar problemas técnicos mediante pensamiento lógico y computacional.
- Comunicar de manera efectiva el proceso y resultados de sus proyectos tecnológicos.

Requerimientos

- Conocimientos básicos de computación y uso de software educativo.
- Materiales: kits de Micro:bit, Arduino básico, componentes electrónicos (LEDs, sensores, motores), herramientas de montaje (destornilladores, cables, pilas).
- Acceso a computadora con conexión a internet para uso de Scratch, Micronit y Minecraft Education Edition.
- Espacio adecuado para realizar actividades prácticas y ensamblaje de prototipos.
- Actitud proactiva y disposición para el trabajo en equipo y la experimentación.

Unidades del Curso

Unidad 1: Introducción a la Mecatrónica y Pensamiento Computacional

Objetivos de Aprendizaje

- Al finalizar la unidad, el estudiante será capaz de identificar los componentes básicos de la mecatrónica y la robótica mediante la exploración de ejemplos prácticos.
- Al finalizar la unidad, el estudiante será capaz de describir los principios fundamentales del pensamiento computacional aplicando conceptos de lógica y resolución de problemas simples.
- Al finalizar la unidad, el estudiante será capaz de utilizar plataformas digitales como Scratch y Micro:bit para crear programas básicos que demuestren secuencias lógicas y control de acciones.
- Al finalizar la unidad, el estudiante será capaz de explicar la integración de elementos electrónicos, mecánicos y programables en proyectos mecatrónicos simples, apoyándose en ejemplos visuales y actividades prácticas.
- Al finalizar la unidad, el estudiante será capaz de analizar problemas cotidianos y proponer soluciones tecnológicas sencillas empleando el pensamiento computacional y herramientas digitales introducidas en el curso.

Contenidos Temáticos

1. Introducción a la Mecatrónica y Robótica

- Definición de mecatrónica: integración de electrónica, mecánica y programación.
- Componentes básicos de un sistema mecatrónico: sensores, actuadores, controladores y estructuras mecánicas.
- Ejemplos prácticos de robots sencillos y dispositivos mecatrónicos en la vida cotidiana.
- Importancia de la robótica en la tecnología actual y en el futuro.

2. Fundamentos del Pensamiento Computacional

- Definición y relevancia del pensamiento computacional.
- Descomposición de problemas: dividir problemas complejos en partes manejables.
- Reconocimiento de patrones y abstracción.
- Diseño de algoritmos y secuencias lógicas para la resolución de problemas.
- Ejemplos prácticos de pensamiento computacional aplicado a situaciones cotidianas.

3. Introducción a Plataformas Digitales: Scratch y Micro:bit

- Presentación de Scratch: entorno visual para la programación por bloques.
- Primeros proyectos en Scratch: creación de secuencias sencillas y control de movimientos y acciones.
- Presentación de Micro:bit: microcontrolador programable y sus componentes básicos (botones, LED, sensores).
- Programación básica en Micro:bit para controlar salidas y leer entradas.
- Comparación de Scratch y Micro:bit para comprender distintos enfoques en programación.

4. Integración de Elementos Mecatrónicos en Proyectos Simples

- Relación entre componentes electrónicos, mecánicos y programables.
- Exploración de ejemplos visuales de proyectos mecatrónicos básicos (e.g., un semáforo controlado por Micro:bit).
- Actividades prácticas para identificar cada tipo de componente en proyectos simples.
- Discusión sobre cómo la programación controla el comportamiento de sistemas físicos.

5. Aplicación del Pensamiento Computacional para Proponer Soluciones Tecnológicas

- Análisis de problemas cotidianos para identificar posibles soluciones tecnológicas.
- Proceso de ideación: uso de pensamiento computacional para diseñar soluciones sencillas.
- Uso de Scratch y Micro:bit para prototipar soluciones básicas.
- Reflexión sobre la importancia de la tecnología y la innovación en la vida diaria.

Actividades

Actividad 1: Explorando Componentes de un Robot Simple

Objetivo: Identificar los componentes básicos de la mecatrónica y la robótica mediante la exploración de ejemplos prácticos.

Descripción:

- Se presentan a los estudiantes imágenes y videos de robots sencillos y dispositivos mecatrónicos.
- En grupos pequeños, los estudiantes identifican y clasifican los componentes (sensores, actuadores, partes mecánicas, controladores).
- Discusión guiada para explicar la función de cada componente.

Organización: Grupos de 3-4 estudiantes.

Producto esperado: Lista clasificada de componentes con breve descripción de su función.

Duración: 45 minutos.

Actividad 2: Resolución de Problemas con Pensamiento Computacional

Objetivo: Describir los principios fundamentales del pensamiento computacional aplicando conceptos de lógica y resolución de problemas simples.

Descripción:

- Se presenta un problema cotidiano simple (por ejemplo, organizar la mochila para la escuela).
- Los estudiantes descomponen el problema en pasos, identifican patrones y diseñan un algoritmo o secuencia lógica para resolverlo.
- Se realiza puesta en común con toda la clase para comparar soluciones y discutir eficiencia.

Organización: Individual o parejas.

Producto esperado: Algoritmo escrito o diagrama de flujo básico para la solución del problema.

Duración: 45 minutos.

Actividad 3: Primeros Pasos en Scratch

Objetivo: Utilizar Scratch para crear programas básicos que demuestren secuencias lógicas y control de acciones.

Descripción:

- Introducción práctica a la interfaz de Scratch.
- Guía paso a paso para crear una animación sencilla que incluya movimientos, sonidos y secuencias lógicas (por ejemplo, un gato que camina y saluda).
- Experimentación libre para modificar y personalizar el proyecto.

Organización: Individual.

Producto esperado: Proyecto en Scratch con secuencias programadas y control de acciones.

Duración: 1 hora.

Actividad 4: Programando Micro:bit para Controlar LEDs

Objetivo: Utilizar Micro:bit para crear programas básicos que controlen salidas y respondan a entradas.

Descripción:

- Breve explicación del hardware de Micro:bit y su entorno de programación.
- Programar un código básico para encender y apagar LEDs usando botones del Micro:bit.
- Prueba y ajuste del programa para entender el control de señales.

Organización: Parejas.

Producto esperado: Programa funcional en Micro:bit que controle LEDs con botones.

Duración: 1 hora.

Actividad 5: Proyecto de Solución Tecnológica Simple

Objetivo: Analizar problemas cotidianos y proponer soluciones tecnológicas sencillas empleando pensamiento computacional y herramientas digitales.

Descripción:

- En grupos, identificar un problema cotidiano sencillo.
- Aplicar pensamiento computacional para descomponer y diseñar una solución.
- Crear un prototipo básico usando Scratch o Micro:bit que represente la solución.
- Presentar el proyecto a la clase explicando la integración de elementos mecánicos, electrónicos y programables.

Organización: Grupos de 4 estudiantes.

Producto esperado: Prototipo funcional y presentación oral.

Duración: 2 horas (puede dividirse en dos sesiones).

Evaluación

Evaluación Diagnóstica

Qué se evalúa: Conocimientos previos sobre mecatrónica, robótica y pensamiento computacional.

Cómo se evalúa: Cuestionario corto con preguntas abiertas y de opción múltiple para identificar ideas previas y experiencias con programación y tecnología.

Instrumento sugerido: Formato impreso o digital con preguntas como: ¿Qué entiendes por robot?, menciona ejemplos de tecnología que usen programación, describe qué harías para organizar una tarea compleja.

Evaluación Formativa

Qué se evalúa: Progreso en la identificación de componentes mecatrónicos, aplicación del pensamiento computacional, y habilidades básicas en Scratch y Micro:bit.

Cómo se evalúa: Observación directa durante actividades, revisión de productos parciales como listas de componentes, algoritmos escritos, proyectos Scratch y códigos Micro:bit.

Instrumento sugerido: Rúbricas de desempeño para cada actividad que consideren claridad conceptual, aplicación de lógica, creatividad y funcionalidad de los programas.

Evaluación Sumativa

Qué se evalúa: Capacidad para integrar conocimientos y habilidades en un proyecto mecatrónico simple, aplicando pensamiento computacional y programación.

Cómo se evalúa: Presentación final del proyecto grupal que incluya prototipo, explicación de componentes y programación, y análisis del problema y solución.

Instrumento sugerido: Rúbrica de evaluación del proyecto que contemple: identificación y explicación de componentes, aplicación correcta de pensamiento computacional, funcionalidad del prototipo, trabajo en equipo y presentación oral.

Unidad 2: Fundamentos de Programación por Bloques con Scratch

Objetivos de Aprendizaje

- Al finalizar la unidad, el estudiante será capaz de identificar y describir los bloques básicos de programación en Scratch para construir secuencias lógicas simples.
- Al finalizar la unidad, el estudiante será capaz de diseñar y programar animaciones o simulaciones sencillas utilizando estructuras básicas como bucles y condicionales en Scratch.
- Al finalizar la unidad, el estudiante será capaz de aplicar conceptos de programación por bloques para controlar dispositivos simples simulados en Scratch, siguiendo instrucciones específicas.
- Al finalizar la unidad, el estudiante será capaz de analizar y corregir errores en programas desarrollados en Scratch para mejorar su funcionamiento y lógica.
- Al finalizar la unidad, el estudiante será capaz de presentar y documentar un proyecto programado en Scratch explicando la lógica y el uso de bloques para resolver un problema específico.

Contenidos Temáticos

1. Introducción a Scratch y la programación por bloques

- ¿Qué es Scratch? Presentación del entorno de programación visual.
- Ventajas de la programación por bloques para principiantes.
- Elementos básicos de la interfaz de Scratch: escenario, sprites, bloques y área de scripts.

2. Bloques básicos y construcción de secuencias lógicas

- Tipos de bloques: movimiento, apariencia, sonido, control, sensores y variables.
- Cómo crear secuencias simples con bloques: orden y ejecución paso a paso.
- Ejemplos de secuencias lógicas sencillas para controlar un sprite.

3. Estructuras básicas de programación: bucles y condicionales

- Uso de bucles (repetir, repetir hasta) para automatizar acciones.
- Condicionales (si, si - entonces - sino) para tomar decisiones en el programa.
- Combinación de bucles y condicionales para crear animaciones y simulaciones.

4. Control de dispositivos simples simulados en Scratch

- Concepto de simulación de dispositivos usando sprites y bloques.
- Programación para controlar movimientos y respuestas de dispositivos simulados.
- Ejemplos prácticos: simulación de semáforo, robot móvil básico, sensor de obstáculos.

5. Identificación y corrección de errores en programas Scratch

- Tipos comunes de errores: sintaxis, lógica y ejecución.
- Estrategias para detectar errores: prueba y observación del comportamiento del programa.
- Técnicas para corregir y optimizar programas para mejor funcionamiento.

6. Presentación y documentación de proyectos en Scratch

- Cómo explicar la lógica detrás del proyecto y el uso de bloques.
- Elementos básicos de documentación: propósito, diseño, código y resultados.
- Formas de presentar proyectos: exposición oral, demostración en vivo y entrega digital.

Actividades

Actividad 1: Explorando la interfaz y bloques básicos de Scratch

Objetivo: Identificar y describir bloques básicos para construir secuencias lógicas simples.

Descripción:

- El docente presenta la interfaz de Scratch y los tipos de bloques.
- Los estudiantes abren Scratch y exploran los bloques de movimiento y apariencia.
- Realizan un ejercicio guiado para mover un sprite en secuencia: avanzar, girar y cambiar disfraz.
- Discusión en grupo sobre qué hace cada bloque y el orden lógico de ejecución.

Organización: Individual

Producto esperado: Un programa simple que mueve y cambia la apariencia de un sprite siguiendo una secuencia lógica.

Duración estimada: 1 hora

Actividad 2: Programando animaciones con bucles y condicionales

Objetivo: Diseñar y programar animaciones usando estructuras básicas como bucles y condicionales.

Descripción:

- Explicación breve sobre bucles y condicionales y ejemplos en Scratch.
- Los estudiantes crean una animación donde un sprite se mueve repetidamente y cambia de dirección si toca el borde.
- Se incluye un condicional para cambiar el color o disfraz del sprite según una condición.
- Prueban y ajustan el programa para que la animación sea fluida y lógica.

Organización: Parejas

Producto esperado: Animación programada con bucles y condicionales que responde a condiciones del entorno.

Duración estimada: 2 horas

Actividad 3: Simulación de un semáforo controlado en Scratch

Objetivo: Aplicar conceptos de programación por bloques para controlar dispositivos simulados siguiendo instrucciones específicas.

Descripción:

- Introducción a la simulación: explicación del semáforo y sus estados.
- Los estudiantes programan un semáforo con luces que cambian en secuencia usando bloques de control.
- Implementan temporizadores con bucles y condicionales para cambiar luces automáticamente.
- Prueban la simulación y corrigen errores detectados en el funcionamiento.

Organización: Grupos de 3-4 estudiantes

Producto esperado: Proyecto funcional de semáforo simulado en Scratch con lógica de control.

Duración estimada: 3 horas

Actividad 4: Diagnóstico y mejora de programas Scratch

Objetivo: Analizar y corregir errores en programas desarrollados para mejorar su funcionamiento y lógica.

Descripción:

- El docente proporciona proyectos Scratch con errores intencionales.
- Los estudiantes ejecutan los programas, identifican fallas y anotan posibles causas.
- En parejas, proponen correcciones y aplican cambios para mejorar el programa.
- Comparan resultados antes y después de la corrección.

Organización: Parejas

Producto esperado: Informe breve con errores identificados, correcciones aplicadas y programa corregido.

Duración estimada: 2 horas

Actividad 5: Presentación y documentación del proyecto final en Scratch

Objetivo: Presentar y documentar un proyecto programado explicando la lógica y el uso de bloques para resolver un problema.

Descripción:

- Los estudiantes eligen o diseñan un proyecto sencillo en Scratch que integre secuencias, bucles, condicionales y simulación.
- Preparan una presentación oral y digital que incluya: objetivo del proyecto, explicación del código y demostración práctica.
- Elaboran documentación escrita que describa la lógica y bloques usados.
- Presentan su proyecto al grupo y responden preguntas.

Organización: Individual o parejas

Producto esperado: Proyecto completo en Scratch, presentación oral y documentación escrita.

Duración estimada: 4 horas (divididas en diseño, documentación y presentación)

Evaluación

Evaluación Diagnóstica

Qué se evalúa: Conocimiento previo sobre programación y familiaridad con interfaces digitales.

Cómo se evalúa: Cuestionario breve y práctica exploratoria en Scratch para identificar bloque básicos.

Instrumento sugerido: Lista de cotejo para observación y cuestionario de opción múltiple.

Evaluación Formativa

Qué se evalúa: Progreso en el uso de bloques básicos, estructuras de control, corrección de errores y aplicación en simulaciones.

Cómo se evalúa: Revisión continua de actividades prácticas, retroalimentación individual y grupal, autoevaluación y coevaluación entre pares.

Instrumento sugerido: Rúbricas para actividades prácticas y listas de chequeo para corrección de errores y aplicación de estructuras.

Evaluación Sumativa

Qué se evalúa: Capacidad para diseñar, programar, corregir y presentar un proyecto completo en Scratch que integre los conceptos aprendidos.

Cómo se evalúa: Presentación final del proyecto, con revisión del código, explicación oral y documentación escrita.

Instrumento sugerido: Rúbrica de evaluación del proyecto que incluya criterios de diseño, funcionalidad, corrección de errores, explicación y documentación.

Unidad 3: Electrónica Básica y Principios de Electricidad

Objetivos de Aprendizaje

- Al finalizar la unidad, el estudiante será capaz de describir los conceptos básicos de electricidad y circuitos eléctricos utilizando terminología adecuada en presentaciones orales y escritas.
- Al finalizar la unidad, el estudiante será capaz de identificar y clasificar componentes electrónicos como resistencias, LEDs, sensores y actuadores en circuitos básicos mediante esquemas y prácticas de laboratorio.
- Al finalizar la unidad, el estudiante será capaz de construir y probar circuitos simples con sensores y actuadores integrados, siguiendo instrucciones y utilizando materiales proporcionados.
- Al finalizar la unidad, el estudiante será capaz de diagnosticar fallas comunes en circuitos básicos y proponer soluciones para corregirlas durante la elaboración de proyectos mecatrónicos.
- Al finalizar la unidad, el estudiante será capaz de integrar circuitos electrónicos básicos en prototipos mecatrónicos, asegurando su funcionamiento correcto y documentando el proceso técnico.

Contenidos Temáticos

1. Introducción a la electricidad y circuitos eléctricos

- Conceptos básicos de electricidad: carga eléctrica, corriente, voltaje y resistencia
- Tipos de corriente eléctrica: corriente continua (CC) y corriente alterna (CA)
- Elementos de un circuito eléctrico: fuente de energía, conductores, carga y controladores
- Terminología básica y símbolos eléctricos estándares
- Leyes fundamentales: Ley de Ohm y Ley de Kirchhoff (introducción conceptual)

2. Componentes electrónicos básicos

- Resistencias: función, tipos, código de colores y aplicación práctica
- Diodos y LEDs: funcionamiento, características y uso en circuitos
- Sensores básicos: definición, tipos comunes (sensor de luz, temperatura, proximidad) y funcionamiento
- Actuadores: definición, tipos básicos (motores, servomotores, zumbadores) y aplicaciones
- Identificación y clasificación de componentes mediante esquemas y etiquetas

3. Construcción y análisis de circuitos eléctricos simples

- Montaje de circuitos en protoboard (placa de pruebas)
- Conexiones en serie y paralelo: características y diferencias
- Integración de sensores y actuadores en circuitos básicos
- Uso de multímetro para medir voltaje, corriente y resistencia
- Interpretación y elaboración de esquemas eléctricos básicos

4. Diagnóstico y solución de fallas en circuitos

- Identificación de fallas comunes: circuito abierto, cortocircuito, componentes mal conectados o dañados
- Técnicas para localizar y corregir errores en circuitos
- Pruebas de funcionamiento con sensores y actuadores integrados
- Documentación de problemas encontrados y soluciones implementadas

5. Integración de circuitos electrónicos en prototipos mecatrónicos

- Concepto de prototipo mecatrónico y su importancia
- Diseño y ensamblaje de circuitos electrónicos en prototipos
- Pruebas funcionales para asegurar el correcto desempeño del circuito integrado
- Documentación técnica: esquemas, lista de materiales, procedimiento de montaje y resultados
- Presentación oral y escrita del proyecto mecatrónico con énfasis en la electrónica aplicada

Actividades

Actividad 1: Explorando conceptos básicos de electricidad

Objetivo: Describir los conceptos básicos de electricidad y circuitos eléctricos utilizando terminología adecuada.

Descripción:

- El docente explica los conceptos básicos y muestra ejemplos visuales de circuitos simples.
- Los estudiantes elaboran un glosario con términos clave y sus definiciones.
- En parejas, preparan una breve presentación oral utilizando el glosario para explicar un concepto asignado (ej. corriente, voltaje, resistencia).
- Presentan al grupo para reforzar el vocabulario y la comprensión.

Organización: Individual para el glosario, parejas para presentación.

Producto esperado: Glosario escrito y presentación oral grupal.

Duración estimada: 2 horas.

Actividad 2: Identificación y clasificación de componentes electrónicos

Objetivo: Identificar y clasificar componentes electrónicos en circuitos básicos mediante esquemas y prácticas de laboratorio.

Descripción:

- El docente muestra diferentes componentes electrónicos y sus símbolos.
- Los estudiantes reciben kits con resistencias, LEDs, sensores y actuadores.
- En grupos pequeños, clasifican los componentes y dibujan sus símbolos en un esquema sencillo.
- Realizan una actividad práctica de reconocimiento y etiquetado en protoboard.

Organización: Grupos de 3-4 estudiantes.

Producto esperado: Esquema con símbolos y lista clasificada de componentes con sus funciones.

Duración estimada: 3 horas.

Actividad 3: Construcción y prueba de circuitos simples con sensores y actuadores

Objetivo: Construir y probar circuitos simples con sensores y actuadores integrados siguiendo instrucciones.

Descripción:

- Los estudiantes reciben instrucciones detalladas para montar un circuito con LED, resistencia y sensor de luz.
- Construyen el circuito en protoboard, conectando correctamente cada componente.
- Prueban el funcionamiento y registran observaciones sobre el comportamiento del sensor y el actuador.
- Discuten en grupo las posibles causas si el circuito no funciona correctamente.

Organización: Parejas.

Producto esperado: Circuito funcional y bitácora con resultados y observaciones.

Duración estimada: 3 horas.

Actividad 4: Diagnóstico y solución de fallas en circuitos básicos

Objetivo: Diagnosticar fallas comunes en circuitos básicos y proponer soluciones durante la elaboración de proyectos mecatrónicos.

Descripción:

- El docente presenta circuitos intencionadamente con fallas comunes.
- Los estudiantes analizan los circuitos y realizan pruebas con multímetro para identificar problemas.
- Proponen y aplican soluciones correctivas para que el circuito funcione.
- Documentan el proceso de diagnóstico y resolución en un informe técnico breve.

Organización: Grupos de 3 estudiantes.

Producto esperado: Informe técnico de diagnóstico y corrección.

Duración estimada: 2.5 horas.

Actividad 5: Integración y presentación de prototipos mecatrónicos con circuitos electrónicos

Objetivo: Integrar circuitos electrónicos básicos en prototipos mecatrónicos, asegurando su funcionamiento y documentando el proceso técnico.

Descripción:

- Los estudiantes diseñan un pequeño prototipo mecatrónico que incluya circuitos eléctricos básicos con sensores y actuadores.
- Construyen el prototipo y prueban su funcionamiento.
- Elaboran un documento técnico con esquemas, lista de materiales, procedimiento y resultados.
- Preparan una presentación oral para explicar su proyecto y mostrar la integración electrónica.

Organización: Grupos de 4 estudiantes.

Producto esperado: Prototipo funcional, documentación técnica y presentación oral.

Duración estimada: 5 horas.

Evaluación

Evaluación diagnóstica

Qué se evalúa: Conocimientos previos sobre conceptos básicos de electricidad y familiaridad con componentes electrónicos.

Cómo se evalúa: Cuestionario escrito breve y discusión guiada en clase.

Instrumento sugerido: Test de opción múltiple y preguntas abiertas.

Evaluación formativa

Qué se evalúa: Progreso en la identificación, construcción, diagnóstico y documentación durante las actividades prácticas.

Cómo se evalúa: Observación directa, revisión de productos parciales (glosario, esquemas, bitácoras, informes técnicos).

Instrumento sugerido: Rúbricas para presentaciones orales, esquemas, bitácoras y reportes técnicos.

Evaluación sumativa

Qué se evalúa: Comprensión integral de los conceptos, habilidad para construir, diagnosticar y documentar circuitos electrónicos y su integración en prototipos mecatrónicos.

Cómo se evalúa: Examen escrito teórico-práctico, evaluación del prototipo final, documentación técnica y presentación oral del proyecto.

Instrumento sugerido: Examen con preguntas teóricas y prácticas, rúbrica de evaluación de prototipo y presentación.

Unidad 4: Programación y Control con Micro:bit

Objetivos de Aprendizaje

- Al finalizar la unidad, el estudiante será capaz de identificar las funciones básicas del dispositivo Micro:bit y sus componentes principales mediante actividades prácticas.
- Al finalizar la unidad, el estudiante será capaz de diseñar y programar secuencias simples en Micro:bit para controlar sensores y motores utilizando programación por bloques, aplicando lógica básica de programación.
- Al finalizar la unidad, el estudiante será capaz de implementar soluciones a problemas prácticos mediante la programación de Micro:bit, evaluando el funcionamiento y realizando ajustes necesarios para mejorar el control de dispositivos.
- Al finalizar la unidad, el estudiante será capaz de documentar el proceso de programación y control realizado con Micro:bit, explicando el diseño, la lógica aplicada y el funcionamiento del prototipo en un contexto mecatrónico.

Contenidos Temáticos

1. Introducción al Micro:bit

- ¿Qué es el Micro:bit? – Descripción general del dispositivo y su propósito en la educación tecnológica.
- Componentes principales del Micro:bit – Pantalla LED, botones, sensores integrados (acelerómetro, brújula), pines de entrada/salida, puerto USB.
- Funciones básicas del Micro:bit – Encendido, conexión a la computadora, uso del editor de programación.

2. Programación básica con Micro:bit usando programación por bloques

- Introducción a la programación por bloques – Uso del editor MakeCode de Microsoft.
- Estructura básica de un programa – Bloques de inicio, eventos y acciones.
- Control de la pantalla LED – Mostrar imágenes, texto y animaciones simples.

- Uso de botones para interacción – Programar respuestas a presionar botones A, B y A+B.
- Uso de sensores integrados – Lectura básica del acelerómetro y brújula para controlar acciones.

3. Control de sensores y motores con Micro:bit

- Introducción a sensores externos – Tipos comunes y conexión al Micro:bit.
- Control de motores simples – Motores DC y servomotores: conceptos básicos y conexión.
- Programación para controlar sensores y motores – Lectura de entradas y control de salidas mediante bloques.
- Uso de lógica básica – Condicionales (si-entonces), ciclos y variables para la toma de decisiones en el programa.

4. Desarrollo de proyectos prácticos con Micro:bit

- Diseño de prototipos – Planificación del proyecto: identificación del problema y selección de sensores/motores.
- Programación y prueba – Codificación, simulación y prueba en el dispositivo real.
- Evaluación y ajustes – Identificación de errores, ajuste de lógica y mejora del control.

5. Documentación y presentación del proyecto

- Registro del proceso – Cómo documentar el diseño, código y pruebas realizadas.
- Explicación de la lógica aplicada – Descripción clara y sencilla de cómo funciona el programa.
- Presentación del prototipo – Estrategias para comunicar el funcionamiento y utilidad del proyecto en contextos mecatrónicos.

Actividades

Actividad 1: Explorando el Micro:bit

Objetivo: Identificar las funciones básicas y componentes del Micro:bit.

Descripción:

- Distribuir Micro:bit a cada estudiante o pareja.
- Guiar una exploración guiada del dispositivo: identificar pantalla LED, botones, pines y sensores integrados.
- Realizar una conexión básica a la computadora y abrir el editor MakeCode.
- Ejecutar un programa simple para mostrar un corazón en la pantalla LED.
- Discutir en grupo las funciones observadas y registrar en un cuaderno.

Organización: Individual o parejas.

Producto esperado: Registro escrito con descripción y dibujos de los componentes y función básica del Micro:bit.

Duración estimada: 1 hora.

Actividad 2: Programando interacciones básicas

Objetivo: Diseñar y programar secuencias simples para controlar la pantalla y botones del Micro:bit.

Descripción:

- En el editor MakeCode, crear un programa que muestre diferentes imágenes o textos según el botón presionado (A, B, A+B).
- Probar el programa en el simulador y luego en el Micro:bit real.
- Agregar uso del acelerómetro para que el dispositivo muestre una imagen cuando se incline.
- Compartir los programas y explicar la lógica usada en pequeños grupos.

Organización: Parejas.

Producto esperado: Programa funcional en Micro:bit con interacciones básicas y explicación escrita o verbal de la lógica.

Duración estimada: 2 horas.

Actividad 3: Controlando sensores y motores

Objetivo: Implementar programación para controlar sensores y motores usando lógica básica.

Descripción:

- En grupos, conectar un sensor externo (por ejemplo, sensor de luz o temperatura) y un motor básico al Micro:bit.
- Diseñar un programa que active el motor cuando el sensor detecte un valor específico.
- Incluir condicionales para controlar el motor (por ejemplo, encender/apagar según el valor del sensor).
- Probar y ajustar el programa para mejorar la respuesta y control.

Organización: Grupos de 3-4 estudiantes.

Producto esperado: Prototipo funcional con sensor y motor controlados por Micro:bit y código comentado.

Duración estimada: 3 horas.

Actividad 4: Documentando y presentando el proyecto

Objetivo: Documentar el proceso y explicar el diseño, lógica y funcionamiento del prototipo.

Descripción:

- Registrar paso a paso el diseño, programación y pruebas realizadas en un cuaderno o documento digital.
- Preparar una presentación breve (oral o con diapositivas) que explique el problema abordado, la solución y cómo funciona el prototipo.
- Presentar a la clase y responder preguntas.

Organización: Grupos.

Producto esperado: Documento de proceso y presentación oral o visual del proyecto.

Duración estimada: 2 horas.

Evaluación

Evaluación diagnóstica

Qué se evalúa: Conocimientos previos sobre dispositivos electrónicos simples y lógica básica de programación.

Cómo se evalúa: Cuestionario breve con preguntas de opción múltiple y respuesta corta sobre conceptos básicos.

Instrumento sugerido: Cuestionario en papel o digital al inicio de la unidad.

Evaluación formativa

Qué se evalúa: Progreso en el diseño, programación y prueba de secuencias en Micro:bit; aplicación de lógica básica; capacidad para resolver problemas y hacer ajustes.

Cómo se evalúa: Observación directa del docente durante las actividades, revisión de código y prototipos, retroalimentación continua.

Instrumento sugerido: Lista de cotejo con criterios de programación, uso de componentes y trabajo colaborativo.

Evaluación sumativa

Qué se evalúa: Producto final: prototipo funcional con Micro:bit, documentación completa y presentación clara del proyecto.

Cómo se evalúa: Revisión del prototipo, análisis del código y documentación, evaluación de la presentación oral o visual.

Instrumento sugerido: Rúbrica que considere funcionamiento, lógica aplicada, calidad de la documentación y claridad en la presentación.

Unidad 5: Introducción a Arduino y Montaje de Prototipos

Objetivos de Aprendizaje

- Al finalizar la unidad, el estudiante será capaz de identificar los componentes básicos de una placa Arduino y sus funciones, describiendo su uso en prototipos mecatrónicos simples.
- Al finalizar la unidad, el estudiante será capaz de programar instrucciones básicas en Arduino para controlar salidas digitales, aplicando conceptos fundamentales de electrónica.
- Al finalizar la unidad, el estudiante será capaz de ensamblar circuitos electrónicos sencillos conectados a Arduino, integrando elementos mecánicos básicos para construir prototipos funcionales.
- Al finalizar la unidad, el estudiante será capaz de probar y depurar programas y circuitos en Arduino, evaluando el correcto funcionamiento de mecanismos controlados.
- Al finalizar la unidad, el estudiante será capaz de documentar el proceso de diseño y montaje de un prototipo con Arduino, explicando el funcionamiento y la integración de sus componentes electrónicos y mecánicos.

Contenidos Temáticos

1. Introducción a Arduino y sus Componentes Básicos

- 1.1 ¿Qué es Arduino?

- Descripción general de Arduino como plataforma de prototipado electrónico y su aplicación en proyectos mecatrónicos.
- 1.2 Componentes de una placa Arduino
 - Microcontrolador
 - Alimentación (conector USB y fuente externa)
 - Entradas y salidas digitales
 - Entradas analógicas
 - Conectores de alimentación y tierra (GND)
 - LED integrado (LED de estado)
 - Función y características de cada componente en el contexto de un prototipo mecatrónico.
- 1.3 Herramientas y accesorios básicos para proyectos con Arduino
 - Protoboard o placa de pruebas
 - Cables jumper
 - Componentes electrónicos básicos (resistencias, LEDs, sensores simples)
 - Importancia de cada herramienta para el montaje y prueba de prototipos.

2. Programación Básica en Arduino para Control de Salidas Digitales

- 2.1 Introducción al entorno de desarrollo Arduino IDE
 - Instalación y configuración básica.
- 2.2 Estructura básica de un programa Arduino (sketch)
 - Funciones setup() y loop()
 - Declaración de variables
- 2.3 Comandos básicos para controlar salidas digitales
 - pinMode(): configurar pines como entrada o salida
 - digitalWrite(): encender y apagar LEDs o actuadores
 - delay(): pausas temporales en el programa
- 2.4 Ejemplo práctico: Encender y apagar un LED
- 2.5 Introducción a la depuración básica de programas

3. Ensamblaje de Circuitos Electrónicos Sencillos con Arduino

- 3.1 Identificación y uso de componentes electrónicos básicos
 - LEDs, resistencias, botones, sensores simples (por ejemplo, sensor de luz LDR)
- 3.2 Uso correcto de la protoboard para montaje de circuitos

- Conexión de componentes y cableado con cables jumper
- 3.3 Integración de elementos mecánicos básicos
 - Servomotores y motores DC básicos
 - Accionamiento básico y montaje mecánico simple
- 3.4 Montaje de un prototipo funcional simple (por ejemplo, semáforo con LEDs o brazo robótico básico)

4. Prueba y Depuración de Programas y Circuitos con Arduino

- 4.1 Verificación del montaje físico y conexiones
- 4.2 Uso del monitor serial para depuración básica
- 4.3 Identificación y solución de errores comunes en hardware y software
- 4.4 Pruebas funcionales de mecanismos controlados

5. Documentación del Proceso de Diseño y Montaje de Prototipos

- 5.1 Importancia de la documentación en proyectos mecatrónicos
- 5.2 Elementos clave a documentar
 - Descripción del prototipo y objetivo
 - Componentes usados y su función
 - Diagramas eléctricos simples
 - Código fuente comentado
 - Proceso de montaje y pruebas
- 5.3 Presentación de resultados y explicación del funcionamiento

Actividades

Actividad 1: Explorando la placa Arduino y sus componentes

Objetivo: Identificar los componentes básicos de una placa Arduino y sus funciones.

Descripción:

- Distribuir una placa Arduino a cada estudiante o pareja.
- Solicitar que identifiquen y marquen con etiquetas adhesivas cada componente básico (microcontrolador, pines digitales, pines analógicos, alimentación, LEDs, etc.).
- Guiar una discusión grupal para describir en voz alta la función de cada componente.
- Completar una hoja con preguntas de reflexión y descripción breve de cada parte.

Organización: Parejas.

Producto esperado: Placa etiquetada y hoja de respuestas con descripciones.

Duración estimada: 45 minutos.

Actividad 2: Programando un LED intermitente

Objetivo: Programar instrucciones básicas en Arduino para controlar salidas digitales.

Descripción:

- Presentar el entorno Arduino IDE y explicar la estructura básica del sketch.
- Guiar a los estudiantes para escribir un programa que encienda y apague un LED con intervalos de un segundo.
- Subir el programa a la placa y observar el funcionamiento.
- Modificar el tiempo de encendido y apagado para experimentar con la función delay().

Organización: Individual o parejas.

Producto esperado: Código funcionando que controla un LED intermitente.

Duración estimada: 60 minutos.

Actividad 3: Montaje de un prototipo sencillo con LED y botón

Objetivo: Ensamblar circuitos electrónicos sencillos conectados a Arduino e integrar controles mecánicos básicos.

Descripción:

- Proveer materiales: protoboard, LED, resistencia, botón, cables jumper y placa Arduino.
- Enseñar el montaje del circuito para que al presionar el botón, el LED se encienda.
- Programar el Arduino para leer el estado del botón y controlar el LED.
- Probar y ajustar el montaje y el código hasta que funcione correctamente.

Organización: Grupos de 3-4 estudiantes.

Producto esperado: Prototipo funcional con botón y LED.

Duración estimada: 90 minutos.

Actividad 4: Documentando el prototipo y presentación final

Objetivo: Documentar el proceso de diseño y montaje del prototipo y explicar su funcionamiento.

Descripción:

- Guiar a los estudiantes para que elaboren un reporte que incluya: descripción del prototipo, lista de componentes, diagrama eléctrico sencillo, código comentado y explicación de pruebas realizadas.
- Preparar una breve presentación oral en la que expliquen su prototipo y respondan preguntas.

Organización: Grupos.

Producto esperado: Reporte escrito y presentación oral.

Duración estimada: 2 sesiones de 45 minutos.

Evaluación

Evaluación Diagnóstica

Qué se evalúa: Conocimientos previos sobre electrónica básica, componentes y programación.

Cómo se evalúa: Cuestionario corto con preguntas abiertas y de opción múltiple sobre conceptos básicos de electricidad, circuitos y programación.

Instrumento sugerido: Prueba escrita o digital de 15 minutos al inicio de la unidad.

Evaluación Formativa

Qué se evalúa: Progreso en la identificación de componentes, habilidad para programar y montar circuitos, capacidad de depuración y trabajo en equipo.

Cómo se evalúa: Observación directa durante actividades prácticas, revisión de códigos y prototipos, retroalimentación continua.

Instrumento sugerido: Lista de cotejo para docentes con criterios de desempeño, diario de clase y autoevaluaciones breves de los estudiantes.

Evaluación Sumativa

Qué se evalúa: Capacidades integrales para identificar componentes, programar y montar un prototipo funcional, depurar problemas y documentar el proceso.

Cómo se evalúa: Producto final (prototipo funcional), reporte escrito y presentación oral grupal.

Instrumento sugerido: Rúbrica detallada que incluya criterios de funcionamiento del prototipo, calidad del código, documentación y claridad en la presentación.

Unidad 6: Integración de Robótica y Mecánica Básica

Objetivos de Aprendizaje

- Al finalizar la unidad, el estudiante será capaz de identificar y explicar los principios básicos de mecánica aplicados en máquinas simples mediante ejemplos y esquemas.
- Al finalizar la unidad, el estudiante será capaz de diseñar y construir una máquina simple que integre componentes mecánicos y electrónicos utilizando materiales proporcionados, siguiendo un plan estructurado.
- Al finalizar la unidad, el estudiante será capaz de programar el control electrónico de la máquina simple utilizando plataformas digitales como Arduino o Micro:bit, aplicando bloques de programación adecuados para su funcionamiento.
- Al finalizar la unidad, el estudiante será capaz de evaluar el funcionamiento de su prototipo mecatrónico, identificando y solucionando problemas técnicos con base en pruebas prácticas.
- Al finalizar la unidad, el estudiante será capaz de presentar y documentar el proyecto de máquina simple, explicando su diseño, funcionamiento y aplicación, utilizando recursos digitales para apoyar su exposición.

Contenidos Temáticos

1. Introducción a la Mecánica Básica y Máquinas Simples

- Conceptos fundamentales de mecánica: fuerza, trabajo, energía y movimiento.
- Tipos de máquinas simples: palanca, plano inclinado, polea, rueda y eje, cuña y tornillo.
- Principios físicos aplicados en máquinas simples: ventaja mecánica y eficiencia.
- Ejemplos cotidianos de máquinas simples y su función.
- Representación gráfica y esquemas básicos de máquinas simples.

2. Diseño y Construcción de una Máquina Simple Integrada

- Proceso de diseño: definición del problema, lluvia de ideas y selección de la máquina a construir.
- Planificación del proyecto: materiales, herramientas y pasos a seguir.
- Integración de componentes mecánicos: ensamblaje de piezas y mecanismos básicos.
- Incorporación de componentes electrónicos: sensores, actuadores y microcontroladores.
- Seguridad y buenas prácticas en la construcción y manejo de materiales.

3. Programación para el Control Electrónico de Máquinas Simples

- Introducción a plataformas digitales: Arduino y Micro:bit.
- Conceptos básicos de programación por bloques: secuencia, eventos y ciclos.
- Programación de sensores y actuadores para controlar la máquina simple.
- Pruebas y depuración del código para asegurar el funcionamiento adecuado.
- Documentación del código y explicación de su lógica.

4. Evaluación y Resolución de Problemas Técnicos en el Prototipo

- Pruebas funcionales: cómo realizar y registrar observaciones.
- Identificación de fallas comunes en la parte mecánica y electrónica.
- Estrategias para solucionar problemas técnicos: ajustes y mejoras.
- Iteración del diseño y programación en función de resultados obtenidos.
- Trabajo colaborativo para la mejora continua del prototipo.

5. Presentación y Documentación del Proyecto de Máquina Simple

- Elaboración de informes técnicos: estructura y contenido relevante.
- Uso de recursos digitales para apoyar la presentación: diapositivas, videos y gráficos.
- Desarrollo de habilidades comunicativas: explicar diseño, funcionamiento y aplicaciones.
- Feedback entre pares y autoevaluación de la presentación.
- Reflexión final sobre el aprendizaje y posibles mejoras futuras.

Actividades

Actividad 1: Explorando Máquinas Simples en la Vida Cotidiana

Objetivo: Identificar y explicar los principios básicos de mecánica aplicados en máquinas simples mediante ejemplos y esquemas.

Descripción paso a paso:

- Los estudiantes forman grupos pequeños y reciben imágenes o objetos de máquinas simples.
- Analizan en grupo el funcionamiento, tipo de máquina simple y principio mecánico aplicado.
- Realizan un esquema sencillo que explique el principio de funcionamiento de la máquina asignada.
- Presentan su análisis y esquema al resto de la clase.

Organización: Grupos de 3-4 estudiantes

Producto esperado: Esquema ilustrativo y explicación oral del principio mecánico aplicado.

Duración estimada: 1 sesión de 50 minutos

Actividad 2: Diseño y Construcción de una Máquina Simple Mecatrónica

Objetivo: Diseñar y construir una máquina simple que integre componentes mecánicos y electrónicos siguiendo un plan estructurado.

Descripción paso a paso:

- En grupos, definen el tipo de máquina simple que desean construir y elaboran un plan de trabajo.
- Seleccionan y organizan los materiales y componentes proporcionados.
- Construyen la estructura mecánica de la máquina simple.
- Integran los componentes electrónicos necesarios (motores, sensores, microcontroladores).

Organización: Grupos de 4-5 estudiantes

Producto esperado: Prototipo físico de máquina simple integrada mecánicamente y electrónicamente.

Duración estimada: 3 sesiones de 50 minutos

Actividad 3: Programación y Control del Prototipo

Objetivo: Programar el control electrónico de la máquina simple utilizando plataformas digitales como Arduino o Micro:bit.

Descripción paso a paso:

- Introducción básica a la plataforma de programación (Arduino o Micro:bit).
- Desarrollo de un programa por bloques para controlar sensores y actuadores del prototipo.
- Prueba y ajustes del código en el prototipo para asegurar su correcto funcionamiento.
- Documentación básica del código y explicación de su lógica.

Organización: Grupos de 4-5 estudiantes (mismos de la construcción)

Producto esperado: Código funcional y prototipo operativo programado.

Duración estimada: 2 sesiones de 50 minutos

Actividad 4: Evaluación, Solución de Problemas y Presentación Final

Objetivo: Evaluar el funcionamiento del prototipo, solucionar problemas técnicos y presentar el proyecto utilizando recursos digitales.

Descripción paso a paso:

- Realizan pruebas funcionales del prototipo, anotando fallas o aspectos a mejorar.
- Discuten y aplican soluciones técnicas para corregir problemas identificados.
- Preparan una presentación digital que incluya diseño, funcionamiento, programación y aplicación.
- Presentan su proyecto ante la clase, respondiendo preguntas y recibiendo retroalimentación.
- Realizan una reflexión grupal sobre el proceso de aprendizaje y posibles mejoras.

Organización: Grupos de 4-5 estudiantes

Producto esperado: Prototipo funcional corregido, presentación digital y reflexión escrita.

Duración estimada: 2 sesiones de 50 minutos

Evaluación

Evaluación Diagnóstica

Qué se evalúa: Conocimientos previos sobre mecánica básica y máquinas simples.

Cómo se evalúa: Cuestionario escrito o discusión guiada con preguntas clave sobre conceptos básicos y tipos de máquinas simples.

Instrumento sugerido: Lista de cotejo o rúbrica para respuestas orales y escritas.

Evaluación Formativa

Qué se evalúa: Progreso en diseño, construcción, programación y capacidad para identificar y resolver problemas.

Cómo se evalúa: Observación directa durante actividades, revisión de esquemas, planes y código, retroalimentación continua.

Instrumento sugerido: Rúbrica de desempeño para cada etapa (diseño, construcción, programación, solución de problemas).

Evaluación Sumativa

Qué se evalúa: Producto final (prototipo funcionando), calidad de la programación, presentación y documentación del proyecto.

Cómo se evalúa: Presentación oral y digital del proyecto, entrega de informe técnico y demostración práctica del prototipo.

Instrumento sugerido: Rúbrica integral que incluya criterios de diseño mecánico, integración electrónica, programación, solución de problemas y comunicación.

Unidad 7: Uso de Micronit y Minecraft para Simulación y Programación

Objetivos de Aprendizaje

- Al finalizar la unidad, el estudiante será capaz de diseñar y simular proyectos mecatrónicos utilizando las plataformas Micronit y Minecraft Education Edition aplicando conceptos básicos de programación por bloques.
- Al finalizar la unidad, el estudiante será capaz de programar secuencias lógicas en Minecraft Education Edition para controlar mecanismos virtuales que representen componentes electrónicos y mecánicos.
- Al finalizar la unidad, el estudiante será capaz de analizar y corregir errores en simulaciones realizadas en Micronit para optimizar el funcionamiento de sus prototipos virtuales.
- Al finalizar la unidad, el estudiante será capaz de documentar y presentar el proceso y resultados de sus proyectos simulados en Micronit y Minecraft, explicando el diseño, programación y aplicación práctica.

Contenidos Temáticos

1. Introducción a la Simulación y Programación en Entornos Virtuales

- Conceptos básicos de simulación y su importancia en mecatrónica: definición, beneficios y aplicaciones.
- Introducción a Micronit como plataforma de simulación mecatrónica: características principales y entorno de trabajo.
- Introducción a Minecraft Education Edition para programación y simulación: interfaz, herramientas y posibilidades para proyectos mecatrónicos.

2. Diseño de Proyectos Mecatrónicos en Micronit

- Creación de componentes básicos y ensamblaje: sensores, actuadores y partes mecánicas virtuales.
- Configuración de parámetros de los componentes para simular comportamientos reales.
- Uso de herramientas de simulación para validar proyectos: pruebas, ajustes y análisis de resultados.

3. Programación por Bloques en Minecraft Education Edition

- Conceptos de programación por bloques: secuencias, condiciones y ciclos.
- Uso del editor de código por bloques en Minecraft para controlar mecanismos virtuales.
- Programación de secuencias lógicas para activar componentes electrónicos y mecánicos dentro del juego.
- Ejemplos prácticos: abrir puertas automáticas, activar elevadores y controlar luces virtuales.

4. Análisis y Corrección de Errores en Simulaciones

- Identificación de errores comunes en simulaciones de Micronit y Minecraft.
- Estrategias para diagnosticar fallos y optimizar el funcionamiento de prototipos virtuales.
- Uso de retroalimentación visual y funcional para mejorar el diseño y la programación.

5. Documentación y Presentación de Proyectos Simulados

- Organización de la documentación: objetivos, diseño, programación y resultados.

- Uso de formatos digitales para presentar procesos y evidencias (videos, capturas de pantalla, reportes).
- Técnicas para presentar proyectos ante el grupo: claridad, explicación técnica y aplicación práctica.

Actividades

Actividad 1: Explorando Micronit y Minecraft Education Edition

Objetivo: Familiarizarse con las interfaces y herramientas básicas de Micronit y Minecraft Education Edition para la simulación y programación.

Descripción:

- Los estudiantes inician sesión en ambas plataformas y exploran sus menús y opciones principales.
- Realizan un recorrido guiado para identificar componentes, herramientas de simulación y programación por bloques.
- Ejecutan una simulación y un programa de ejemplo sencillo para observar el funcionamiento básico.

Organización: Individual

Producto esperado: Capturas de pantalla de las interfaces y breve reporte escrito con observaciones.

Duración estimada: 1 hora

Actividad 2: Diseño y Simulación de un Proyecto Mecatrónico en Micronit

Objetivo: Diseñar y simular un proyecto simple en Micronit aplicando conceptos básicos de mecatrónica.

Descripción:

- En grupos, seleccionan un proyecto sencillo (por ejemplo, un sistema de semáforo o un brazo mecánico).
- Diseñan el proyecto ensamblando componentes y configurando sus parámetros.
- Simulan el funcionamiento y ajustan el diseño para corregir errores y mejorar desempeño.

Organización: Grupos de 3-4 estudiantes

Producto esperado: Proyecto simulado en Micronit con reporte de diseño, simulación y ajustes realizados.

Duración estimada: 3 horas

Actividad 3: Programación de Secuencias Lógicas en Minecraft Education Edition

Objetivo: Programar secuencias lógicas para controlar mecanismos virtuales en Minecraft.

Descripción:

- En parejas, eligen un mecanismo virtual para controlar (puerta automática, elevador, luz).
- Diseñan y programan la secuencia lógica usando programación por bloques.
- Prueban y ajustan su programa para asegurar el correcto funcionamiento del mecanismo.

Organización: Parejas

Producto esperado: Programa funcional en Minecraft con explicación escrita del código y su lógica.

Duración estimada: 2 horas

Actividad 4: Presentación y Documentación de Proyectos Simulados

Objetivo: Documentar y presentar los procesos y resultados de los proyectos realizados en Micronit y Minecraft.

Descripción:

- Cada grupo o pareja prepara un reporte digital que incluya diseño, programación, simulación, problemas encontrados y soluciones.
- Preparan una presentación oral apoyada con evidencias visuales (videos, capturas, diagramas).
- Presentan ante el grupo y responden preguntas para compartir aprendizajes y mejoras.

Organización: Grupos o parejas según proyecto

Producto esperado: Reporte digital y presentación oral documentada y clara.

Duración estimada: 2 horas

Evaluación

Evaluación Diagnóstica

¿Qué se evalúa? Conocimientos previos sobre simulación, programación por bloques y uso básico de plataformas digitales.

¿Cómo se evalúa? Cuestionario breve de selección múltiple y preguntas abiertas relacionadas con conceptos básicos y experiencia previa.

Instrumento sugerido: Cuestionario digital o impreso al inicio de la unidad.

Evaluación Formativa

¿Qué se evalúa? Progreso en el diseño, programación, análisis de errores y documentación durante las actividades prácticas.

¿Cómo se evalúa? Observación directa, revisión de proyectos simulados, retroalimentación continua y autoevaluación guiada.

Instrumento sugerido: Rúbrica para evaluar diseño, programación, corrección de errores y calidad de documentación durante las actividades.

Evaluación Sumativa

¿Qué se evalúa? Competencia para diseñar y simular proyectos, programar secuencias lógicas, analizar y corregir errores, y presentar resultados claramente.

¿Cómo se evalúa? Evaluación del proyecto final completo: simulación en Micronit, programación en Minecraft, documentación y presentación oral.

Instrumento sugerido: Rúbrica detallada con criterios para diseño, programación, análisis, documentación y comunicación.

Unidad 8: Diseño y Presentación de Proyecto Final Mecatrónico

Objetivos de Aprendizaje

- Al finalizar la unidad, el estudiante será capaz de diseñar un prototipo mecatrónico integrando conceptos de robótica, electrónica y programación por bloques, utilizando plataformas digitales como Micro:bit o Arduino.
- Al finalizar la unidad, el estudiante será capaz de construir y programar un prototipo funcional aplicando principios básicos de electricidad y mecánica, asegurando la correcta operación del sistema.
- Al finalizar la unidad, el estudiante será capaz de documentar el proceso de diseño, construcción y programación de su proyecto mecatrónico, incluyendo diagramas, códigos y explicaciones claras.
- Al finalizar la unidad, el estudiante será capaz de presentar oralmente su proyecto mecatrónico, explicando su funcionamiento, diseño y aplicación, utilizando recursos visuales y tecnológicos adecuados.
- Al finalizar la unidad, el estudiante será capaz de evaluar y mejorar su prototipo mecatrónico mediante la identificación de problemas y la implementación de soluciones creativas basadas en retroalimentación recibida.

Contenidos Temáticos

1. Diseño del prototipo mecatrónico

- Introducción al diseño mecatrónico: Conceptos básicos y componentes integradores (robótica, electrónica y programación).
- Selección de plataforma digital: Ventajas y características de Micro:bit y Arduino para proyectos escolares.
- Planificación del diseño: Elaboración de bocetos, diagramas y especificaciones técnicas.
- Integración de sensores y actuadores: Tipos, funcionamiento y aplicaciones en el prototipo.

2. Construcción y programación del prototipo

- Montaje mecánico: Herramientas, materiales y técnicas para construcción segura y funcional.
- Fundamentos de electricidad aplicados: Circuitos básicos, conexiones seguras y uso de componentes electrónicos.
- Programación por bloques: Uso de entornos visuales para programar Micro:bit o Arduino.
- Integración hardware-software: Pruebas y ajustes para asegurar correcto funcionamiento.

3. Documentación del proyecto mecatrónico

- Técnicas para documentar el diseño: Diagramas eléctricos, esquemas mecánicos y flujo del programa.
- Elaboración de manuales y reportes: Estructura, lenguaje claro y uso de imágenes.
- Registro del proceso: Bitácoras de trabajo y evidencias del desarrollo.

4. Presentación oral del proyecto

- Planificación de la presentación: Organización de ideas y uso de recursos visuales (diapositivas, videos, prototipo en vivo).

- Técnicas de comunicación oral: Claridad, lenguaje apropiado y manejo del tiempo.
- Uso de tecnologías para la presentación: Herramientas digitales para apoyar la exposición.

5. Evaluación y mejora del prototipo

- Identificación de problemas y fallas: Métodos para detectar errores en diseño, construcción y programación.
- Análisis de retroalimentación: Interpretación de comentarios y sugerencias de compañeros y docentes.
- Implementación de mejoras: Propuestas creativas para optimizar el prototipo.
- Pruebas posteriores a las mejoras: Verificación de la efectividad de las modificaciones.

Actividades

Actividad 1: Diseño colaborativo del prototipo

Objetivo: Desarrollar un diseño inicial integrando conceptos de robótica, electrónica y programación usando Micro:bit o Arduino.

Descripción:

- Formar grupos de 3-4 estudiantes.
- Analizar el problema o función que resolverá el prototipo.
- Elaborar bocetos y diagramas del diseño mecánico y electrónico.
- Seleccionar los componentes necesarios y plataforma digital.
- Presentar el diseño preliminar al docente para retroalimentación.

Organización: Grupos

Producto esperado: Documento con bocetos, lista de componentes y descripción del diseño.

Duración: 2 sesiones de 50 minutos.

Actividad 2: Construcción y programación del prototipo

Objetivo: Construir y programar un prototipo funcional aplicando principios de electricidad y mecánica.

Descripción:

- Montar el chasis y conectar los componentes electrónicos siguiendo el diseño.
- Programar el prototipo con bloques para controlar los sensores y actuadores.
- Realizar pruebas para verificar el funcionamiento correcto.
- Realizar ajustes necesarios en hardware o software.

Organización: Grupos

Producto esperado: Prototipo mecatrónico funcionando conforme al diseño.

Duración: 3 sesiones de 50 minutos.

Actividad 3: Documentación del proyecto

Objetivo: Documentar el proceso de diseño, construcción y programación con diagramas, códigos y explicaciones claras.

Descripción:

- Registrar el esquema eléctrico y mecánico del prototipo.
- Incluir el código fuente con comentarios explicativos.
- Describir paso a paso el proceso y las decisiones tomadas.
- Preparar un manual básico para el uso y mantenimiento del prototipo.

Organización: Grupos

Producto esperado: Documento de proyecto completo y comprensible.

Duración: 2 sesiones de 50 minutos.

Actividad 4: Presentación y retroalimentación del proyecto

Objetivo: Presentar oralmente el proyecto explicando su funcionamiento y recibir retroalimentación para mejoras.

Descripción:

- Preparar una presentación con apoyo visual (diapositivas, videos, demostración en vivo).
- Exponer el proyecto ante el grupo y docente, explicando diseño, funcionamiento y aplicaciones.
- Recibir preguntas y sugerencias del público.
- Registrar la retroalimentación para implementar mejoras.

Organización: Grupos

Producto esperado: Presentación oral y lista de mejoras propuestas.

Duración: 2 sesiones de 50 minutos.

Evaluación

Evaluación diagnóstica

Qué se evalúa: Conocimientos previos sobre conceptos básicos de robótica, electrónica y programación por bloques.

Cómo se evalúa: Cuestionario breve y discusión grupal sobre experiencias previas.

Instrumento sugerido: Cuestionario escrito o digital con preguntas de opción múltiple y preguntas abiertas.

Evaluación formativa

Qué se evalúa: Progreso en diseño, construcción, programación y documentación del prototipo.

Cómo se evalúa: Observación directa, revisión de avances, retroalimentación continua en actividades, y rúbrica para cada etapa.

Instrumento sugerido: Rúbricas específicas para diseño, montaje, programación y documentación; listas de cotejo; diarios de trabajo.

Evaluación sumativa

Qué se evalúa: Producto final (prototipo funcional y documentación), presentación oral y capacidad para identificar y proponer mejoras.

Cómo se evalúa: Presentación ante grupo y docente, entrega del documento completo, y defensa del proyecto con respuestas a preguntas.

Instrumento sugerido: Rúbrica integral que considere funcionalidad del prototipo, calidad de documentación, claridad en la presentación y capacidad de análisis crítico para mejoras.