

# ¡Silencio en Verde! Diseño y Construcción de un Semáforo de Control de Ruido con Arduino

Tecnología e Informática | Pensamiento Computacional | Aprendizaje Basado en Proyectos

## Descripción

En este plan de clase, los estudiantes de secundaria explorarán el mundo del pensamiento computacional y la electrónica a través del diseño y construcción de un semáforo que controla los niveles de ruido utilizando Arduino. Aprenderán cómo medir el sonido ambiente, interpretar datos y programar dispositivos para emitir señales visuales que ayuden a mantener un ambiente tranquilo. Este proyecto es relevante porque conecta la tecnología con situaciones reales de la vida diaria, como el control del ruido en aulas, bibliotecas o espacios comunitarios, promoviendo la conciencia sobre la contaminación acústica y el respeto al entorno. Además, fomenta habilidades colaborativas, el pensamiento lógico y la creatividad, todo dentro de una experiencia de aprendizaje activo y significativo.

## Objetivos de Aprendizaje

- Analizar el funcionamiento básico de un sensor de sonido y un semáforo LED para controlar niveles de ruido.
- Diseñar y programar un sistema con Arduino que active señales visuales según la intensidad del sonido ambiente.
- Construir un prototipo funcional de semáforo de control de ruido utilizando materiales electrónicos y herramientas digitales.
- Colaborar efectivamente en equipo para resolver problemas técnicos y presentar resultados.
- Evaluar el impacto del dispositivo en el control del ruido y reflexionar sobre aplicaciones prácticas.

## Recursos Necesarios

- Placas Arduino UNO (1 por equipo)
- Sensores de sonido (1 por equipo)
- LEDs rojos, amarillos y verdes (3 por equipo)
- Resistencias (220Ω, 3 por equipo)
- Protoboard (1 por equipo)
- Cables puente (varios por equipo)
- Computadoras con software Arduino IDE instalado (1 por equipo)
- Proyector y computadora para presentación inicial
- Material impreso con esquema básico de conexión y ejemplo de código
- Multímetro (opcional, para medición de conexiones)

- Acceso a videos cortos demostrativos sobre sensores de sonido y semáforos LED

## Requisitos Previos

- Conocimientos básicos sobre corriente eléctrica y circuitos simples.
- Familiaridad previa con el uso de computadoras y software básico.
- Conceptos iniciales de programación (variables, condicionales).
- Experiencia previa en trabajo colaborativo en el aula.

## Actividades

### Fase de Inicio

**Tiempo estimado:** 30 minutos

#### Propósito de la sesión

**Docente:** Explica a los estudiantes que en esta sesión aprenderán a diseñar y construir un semáforo que detecta el ruido para ayudar a controlar ambientes ruidosos, algo que puede aplicarse en su escuela o casa.

#### Activación de conocimientos previos

**Docente:** Pregunta a los estudiantes: "¿Han notado lugares donde el ruido es un problema? ¿Cómo creen que se podría controlar sin necesidad de un adulto presente todo el tiempo?"

**Estudiantes:** Responden con ejemplos y opiniones, generando un diálogo breve.

#### Motivación y enganche

**Docente:** Muestra un video corto (2-3 minutos) donde se ve un semáforo de control de ruido en una biblioteca real y comparte un dato curioso: "¿Sabían que niveles altos de ruido pueden afectar su concentración y salud?"

**Estudiantes:** Observan el video y comentan sus impresiones.

#### Contextualización

**Docente:** Relaciona el proyecto con su vida cotidiana: "Ustedes pueden crear este dispositivo para mejorar espacios donde estudian, juegan o descansan, cuidando el ambiente y ayudando a sus compañeros".

**Estudiantes:** Reconocen la importancia y se preparan para iniciar el proyecto.

---

### Fase de Desarrollo

**Tiempo estimado:** 120 minutos

#### Presentación del contenido

**Docente:** Explica brevemente cómo funciona un sensor de sonido y cómo se puede programar un Arduino para que encienda diferentes LEDs según el nivel de ruido, usando un lenguaje sencillo y apoyado con esquemas visuales impresos.

### Actividad 1: Explorando el sensor de sonido y LEDs

- **Objetivo:** Analizar el funcionamiento básico del sensor y LEDs.
- **Instrucciones:**
  - **Docente:** Divide a los estudiantes en equipos de 3-4 y entrega los materiales.
  - Los equipos conectan el sensor de sonido y un LED verde en la protoboard siguiendo un esquema básico proporcionado.
  - Con ayuda del docente, prueban cómo cambia la luz al hacer ruido cerca del sensor.
- **Organización:** Grupos de 3-4
- **Producto:** Prototipo básico con sensor y LED que responde al sonido.
- **Tiempo:** 40 minutos
- **Rol del docente:** Supervisa conexiones, hace preguntas como "¿Qué pasa cuando el ruido aumenta? ¿Por qué creen que sucede eso?" y orienta la solución de problemas.

### Actividad 2: Programando el semáforo de control de ruido

- **Objetivo:** Diseñar y programar el sistema para controlar LEDs según niveles de ruido.
- **Instrucciones:**
  - **Docente:** Presenta un fragmento de código base y explica cómo programar los umbrales para luz roja, amarilla y verde.
  - Los equipos modifican y cargan el programa en el Arduino para que los tres LEDs indiquen diferentes niveles de ruido: verde (bajo), amarillo (moderado), rojo (alto).
  - Prueban el funcionamiento con diferentes sonidos.
- **Organización:** Grupos de 3-4
- **Producto:** Código funcional y prototipo con semáforo que responde a niveles de ruido.
- **Tiempo:** 50 minutos
- **Rol del docente:** Apoya con dudas de programación, sugiere pruebas, pregunta "¿Cómo podríamos mejorar la respuesta del semáforo?" y fomenta la colaboración.

### Actividad 3: Presentación y ajustes finales

- **Objetivo:** Colaborar para optimizar el prototipo y preparar una breve presentación.
- **Instrucciones:**
  - **Docente:** Solicita a los grupos que prueben su semáforo, identifiquen posibles mejoras y preparen una explicación sencilla sobre cómo funciona y para qué sirve.

- Los equipos hacen ajustes y ensayan su presentación.
- **Organización:** Grupos de 3-4
- **Producto:** Prototipo optimizado y presentación oral breve.
- **Tiempo:** 30 minutos
- **Rol del docente:** Observa, da retroalimentación específica y guía para mejorar la comunicación.

## Diferenciación

- **Estudiantes que terminan antes:** Pueden explorar modificar el código para que el semáforo también emita sonidos o usar un multímetro para medir voltajes en el circuito.
- **Estudiantes que necesitan apoyo:** Reciben ayuda personalizada para conexiones y programación básica, con explicaciones paso a paso y materiales visuales adicionales.

## Transiciones

**Docente:** Después de cada actividad, conecta con la siguiente preguntando cómo lo que aprendieron les ayudará a seguir mejorando su proyecto y motivando a continuar experimentando.

---

## Fase de Cierre

**Tiempo estimado:** 30 minutos

### Síntesis

**Docente:** Propone que cada grupo realice un mapa mental colectivo en una hoja grande donde plasmen las partes del semáforo, cómo funciona y qué aprendieron.

**Estudiantes:** Colaboran para elaborar el mapa mental, haciendo aportes y organizando la información.

### Reflexión metacognitiva

**Docente:** Formulan y escriben en su cuaderno las siguientes preguntas:

- ¿Qué parte del proyecto te resultó más fácil y por qué?
- ¿Qué dificultades encontraste al programar o armar el circuito y cómo las solucionaste?
- ¿Cómo crees que este semáforo puede ayudar en tu escuela o comunidad?

### Retroalimentación

**Docente:** Proporciona retroalimentación inmediata a cada grupo destacando sus logros y sugiriendo mejoras específicas, enfatizando el proceso y el trabajo en equipo.

### Transferencia

**Docente:** Invita a los estudiantes a pensar en otros problemas cotidianos que podrían resolver con Arduino y sensores, preparando el camino para futuros proyectos.

## Tarea o reto

**Docente:** Propone que en casa o en grupo piensen en un lugar donde el ruido sea un problema y escriban una breve propuesta de cómo podrían adaptar su semáforo para ese espacio, con dibujos o esquemas sencillos.

## Evaluación

### Tipo de evaluación:

- **Diagnóstica:** Durante la fase de inicio, mediante la pregunta detonadora para valorar conocimientos previos sobre ruido y control.
- **Formativa:** En la fase de desarrollo, observando la participación, la construcción del prototipo y la programación, con retroalimentación continua.
- **Sumativa:** En el cierre, mediante la presentación del prototipo, el mapa mental colectivo y la reflexión escrita.

### Criterios de evaluación:

- Capacidad para analizar y explicar el funcionamiento del sensor y semáforo (Objetivo 1).
- Diseño y programación correcta del sistema para responder a niveles de ruido (Objetivo 2).
- Construcción funcional del prototipo electrónico (Objetivo 3).
- Trabajo colaborativo efectivo y presentación clara (Objetivo 4).
- Reflexión sobre la utilidad y aplicación del dispositivo (Objetivo 5).

### Instrumentos sugeridos:

- Lista de cotejo para evaluación de prototipo y presentación.
- Rúbrica para valorar programación y diseño.
- Observación directa durante actividades en equipo.
- Autoevaluación y coevaluación con guías específicas para la reflexión.
- Portafolio con código, esquemas y mapas mentales elaborados.

### Evidencias de aprendizaje:

- Prototipo funcional del semáforo de control de ruido.
- Código programado y cargado en Arduino.
- Mapa mental colectivo que sintetiza el aprendizaje.
- Respuestas escritas en la reflexión metacognitiva.
- Presentación oral clara y organizada del proyecto.

## Enriquecimientos

### Inicio - Activar

**Actividad para Activar Conocimientos Previos: "¿Qué Sabemos Sobre el Ruido y los Semáforos?"**

**Duración:** 7 minutos

**Objetivo:** Que los estudiantes reflexionen sobre el concepto de ruido, su impacto y el uso de dispositivos electrónicos básicos (como semáforos) para controlar situaciones, conectando con el diseño y construcción de un semáforo de control de ruido con Arduino.

- **Materiales:** Pizarrón o pantalla para anotar, tarjetas o post-its, marcador.

- **Procedimiento:**

1. Iniciar con una pregunta abierta al grupo: "*¿Qué es el ruido y cómo nos afecta en nuestro día a día?*". Invitar a varios estudiantes a compartir sus ideas brevemente (2 minutos).
2. Luego, preguntar: "*¿Han visto o usado alguna vez un semáforo? ¿Para qué sirve un semáforo y cómo nos ayuda?*" Recoger respuestas rápidas (2 minutos).
3. Introducir la idea de que existen semáforos para diferentes situaciones, no solo para el tráfico, y preguntar: "*¿Cómo creen que podríamos usar un semáforo para controlar el ruido?*" (2 minutos).
4. Finalmente, anotar en el pizarrón las respuestas clave que conectan ruido, control y semáforos, preparando el terreno para el proyecto (1 minuto).

**Conexión con los Objetivos de Aprendizaje:** Esta actividad permite a los estudiantes activar y relacionar conocimientos previos sobre ruido, su impacto y el funcionamiento básico de un semáforo, preparando su comprensión para el diseño y la construcción de un semáforo inteligente con Arduino, en línea con la metodología de Aprendizaje Basado en Proyectos.

## Inicio - Contextualizar

### Contextualización para la Fase de Inicio

Imagina que estás en tu salón de clases o en la biblioteca de la escuela y de repente el ruido a tu alrededor sube tanto que te cuesta concentrarte para estudiar o escuchar a tu profesor. Seguramente te ha pasado que, cuando hay mucho ruido, es difícil mantener el enfoque y hacer las tareas bien. Además, en lugares públicos como bibliotecas, centros comerciales o incluso en casa, el nivel de ruido puede afectar nuestro bienestar y el de quienes nos rodean.

Hoy en día, en muchas ciudades y escuelas, existen dispositivos tecnológicos que ayudan a controlar el ruido para mantener ambientes tranquilos y saludables. Por ejemplo, en algunas bibliotecas hay semáforos que cambian de color para indicarnos cuándo el ruido está muy alto y debemos bajar la voz.

En esta sesión, vamos a diseñar y construir nuestro propio semáforo de control de ruido usando Arduino, una pequeña computadora que nos permite crear proyectos electrónicos. Con este proyecto aprenderemos a medir el sonido del ambiente y a usar luces para avisar cuando el ruido es demasiado fuerte, algo que podemos aplicar en nuestra escuela, en casa o en cualquier lugar donde queramos promover el respeto y la concentración.

Al trabajar en este proyecto, no solo aprenderás sobre electrónica y programación, sino que también estarás desarrollando habilidades para resolver problemas reales que afectan tu vida diaria y la de tu comunidad. ¿Están listos para crear un ambiente más silencioso y agradable mientras exploramos la tecnología? ¡Vamos a comenzar!

## Desarrollo - Ejemplos

## Ejemplos Prácticos para el Proyecto "¡Silencio en Verde!"

Estos ejemplos prácticos están diseñados para que los estudiantes de secundaria comprendan y apliquen conceptos de diseño y construcción de un semáforo de control de ruido con Arduino, alineados con la metodología de Aprendizaje Basado en Proyectos y la duración de la sesión.

### • Ejemplo 1: Control de ruido en la biblioteca escolar

- *Contexto:* La biblioteca de la escuela es un lugar donde los estudiantes deben mantener silencio para facilitar la concentración.
- *Aplicación:* Los estudiantes diseñan un semáforo que cambia de verde a rojo cuando el nivel de ruido supera un umbral definido, alertando a los usuarios para que bajen la voz.
- *Objetivo de aprendizaje relacionado:* Comprender la interacción entre sensores de sonido y actuadores (LEDs) usando Arduino para monitorear un ambiente real.

### • Ejemplo 2: Semáforo de control de ruido en el aula durante exámenes

- *Contexto:* Durante exámenes, es importante mantener un ambiente silencioso para que los estudiantes puedan concentrarse.
- *Aplicación:* El semáforo detecta niveles altos de ruido y enciende una luz roja para indicar que deben bajar el volumen o guardar silencio.
- *Objetivo de aprendizaje relacionado:* Aplicar programación básica en Arduino para procesar datos del sensor y controlar salidas visuales en un contexto escolar.

### • Ejemplo 3: Control de ruido en eventos deportivos escolares

- *Contexto:* Durante partidos o eventos deportivos, los niveles de ruido pueden ser muy altos y afectar la comunicación de los organizadores.
- *Aplicación:* El semáforo se utiliza para alertar a los espectadores cuando el ruido excede un nivel seguro, ayudando a mantener un ambiente controlado.
- *Objetivo de aprendizaje relacionado:* Desarrollar habilidades para ajustar parámetros del sensor y calibrar el sistema para diferentes contextos de ruido.

## Casos de Estudio para Análisis y Discusión

### • Caso de Estudio 1: Implementación de semáforos de control ambiental en escuelas

- Los estudiantes analizan un caso real de una escuela que implementó sensores para controlar niveles de ruido en diferentes espacios, discutiendo beneficios y retos.
- Se fomenta la reflexión sobre cómo la tecnología puede mejorar el ambiente escolar y el bienestar estudiantil.

### • Caso de Estudio 2: Arduino en proyectos sociales para mejorar el entorno

- Se presenta un proyecto comunitario donde jóvenes usan Arduino para monitorear contaminación acústica en su barrio.

- Los estudiantes comparan este proyecto con su semáforo de control de ruido, identificando similitudes en diseño y aplicación.

## **Desarrollo - Gamificar**

### **Elementos de Gamificación para la Fase de Desarrollo**

Para motivar a los estudiantes de secundaria (12-15 años) durante la fase de desarrollo del proyecto "¡Silencio en Verde! Diseño y Construcción de un Semáforo de Control de Ruido con Arduino", se proponen las siguientes mecánicas de juego. Estas están diseñadas para reforzar el aprendizaje en tecnología, pensamiento computacional y trabajo colaborativo, sin distraer del objetivo principal del proyecto.

- **Desafíos por Equipos:** Los estudiantes se organizarán en pequeños grupos y recibirán retos específicos relacionados con la construcción y programación del semáforo. Por ejemplo, programar correctamente la lectura del sensor de sonido, o lograr que el semáforo cambie de color según niveles de ruido. Cada desafío completado otorga puntos al equipo.
- **Sistema de Puntos y Recompensas:** A medida que los equipos avanzan en la construcción y programación, acumulan puntos por tareas realizadas correctamente, creatividad en las soluciones y colaboración efectiva. Los puntos pueden traducirse en insignias digitales (por ejemplo, "Programador Experto", "Ingeniero Creativo", "Colaborador Estrella").
- **Tiempo Limitado para Mini-Retos:** Se incluyen pequeños desafíos cronometrados (de 10 a 15 minutos) para resolver problemas puntuales, como identificar errores comunes en el código o mejorar la conexión de sensores. Esto genera dinamismo y mantiene la atención.
- **Banco de Pistas y Ayudas:** Para evitar frustración y mantener el flujo del proyecto, los equipos pueden "canjear" puntos para obtener pistas o ayudas sobre problemas técnicos o de programación.
- **Tablero de Progreso Visible:** Se colocará un tablero en el aula donde se muestre el avance de cada equipo, los puntos acumulados y las insignias obtenidas. Esto fomenta la competencia sana y la motivación continua.
- **Retroalimentación Instantánea:** Después de cada mini-reto o entrega parcial, el docente proporciona retroalimentación rápida y constructiva, destacando aciertos y orientando mejoras, reforzando así el aprendizaje continuo.

Estas mecánicas están diseñadas para integrarse naturalmente a la sesión de 3 horas, promoviendo el compromiso, la colaboración y la aplicación práctica de conocimientos en tecnología y pensamiento computacional, alineados con la metodología de Aprendizaje Basado en Proyectos.

## **Desarrollo - Evaluar**

### **Herramientas de Evaluación Formativa para el Plan de Clase**

Para monitorear el progreso de los estudiantes durante la sesión de 3 horas en el proyecto "¡Silencio en Verde! Diseño y Construcción de un Semáforo de Control de Ruido con Arduino", se proponen las siguientes herramientas de evaluación formativa rápidas, claras y adecuadas para estudiantes de secundaria (12-15 años). Estas herramientas

están alineadas a los objetivos de aprendizaje y permiten al docente identificar avances y dificultades en tiempo real.

1. Lista de Verificación (Checklist) de Habilidades y Conocimientos

El docente puede utilizar esta lista para observar y registrar si los estudiantes cumplen con pasos clave durante el desarrollo del proyecto.

| Actividad / Habilidad                                                         | Indicador de Logro                            | Marca (✓ / x) |
|-------------------------------------------------------------------------------|-----------------------------------------------|---------------|
| Identificación correcta de componentes electrónicos (Arduino, sensores, LEDs) | Reconoce y nombra cada componente             |               |
| Conexión adecuada del circuito básico en la protoboard                        | Realiza conexiones sin errores visibles       |               |
| Programación básica en Arduino IDE para controlar LEDs según nivel de ruido   | Escribe y carga el código sin errores graves  |               |
| Prueba funcional del semáforo de ruido                                        | Semáforo responde a variaciones de sonido     |               |
| Trabajo colaborativo y distribución de tareas                                 | Participa activamente y coopera con el equipo |               |

2. Preguntas de Autoevaluación Rápida (Check-in de 5 minutos)

Al menos dos veces durante la sesión (por ejemplo a la mitad y al final), el docente puede pedir a los estudiantes responder brevemente estas preguntas por escrito o en voz alta para reflexionar sobre su aprendizaje y detectar dudas.

- ¿Qué componente electrónico te fue más fácil conectar? ¿Por qué?
- ¿Qué parte de la programación te resultó más desafiante?
- ¿El semáforo responde correctamente al nivel de ruido? ¿Qué cambios harías?
- ¿Cómo contribuiste al trabajo en equipo?

3. Observación Directa y Feedback Inmediato

Durante el trabajo en grupo, el docente circula entre los equipos haciendo preguntas claves y observando el progreso, por ejemplo:

- ¿Cómo estás midiendo el nivel de ruido con Arduino?
- ¿Qué código usaste para cambiar el color del LED según el ruido?
- ¿Qué problemas encontraste y cómo los solucionaste?

Con base en las respuestas y observaciones, el docente ofrece retroalimentación puntual para corregir errores o reforzar conceptos.

4. Mini Presentación o Demostración Final

Al concluir la sesión, cada equipo realiza una breve demostración (3-5 minutos) mostrando su semáforo y explicando su funcionamiento. El docente evalúa:

- Claridad en la explicación del proyecto
- Funcionamiento del semáforo frente al ruido
- Trabajo colaborativo evidenciado en la presentación

Esta actividad sirve para consolidar el aprendizaje y permite al docente identificar necesidades de refuerzo para futuras sesiones.

Desarrollo - Tareas

Tareas Estructuradas para la Fase de Desarrollo

En esta fase, los estudiantes aplicarán conocimientos y habilidades para diseñar y construir un semáforo de control de ruido con Arduino. Cada tarea está orientada a avanzar en el proyecto, desarrollar competencias específicas y fomentar el trabajo colaborativo dentro del tiempo de 3 horas.

| Tarea                               | Instrucciones                                                                                                                                                                                                                                                                                                                                                                                 | Tiempo Estimado | Producto Esperado                                     | Objetivo de Aprendizaje                                                        |
|-------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------------------------------------------------|--------------------------------------------------------------------------------|
| 1. Diseño del Diagrama del Semáforo | <ul style="list-style-type: none"><li>• En equipos, dibujen un esquema simple que muestre cómo se conectarán los LEDs (rojo, amarillo, verde) y el sensor de sonido al Arduino.</li><li>• Identifiquen los pines que usarán y marquen el flujo de cómo el semáforo responderá a los niveles de ruido.</li><li>• Utilicen símbolos claros y etiquetas para facilitar la comprensión.</li></ul> | 30 minutos      | Diagrama esquemático del circuito en papel o digital. | Comprender y planificar la estructura básica del circuito y su funcionamiento. |

|                                           |                                                                                                                                                                                                                                                                                                                                                            |            |                                                                               |                                                                                         |
|-------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------|-------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------|
| 2. Conexión<br>Física del<br>Circuito     | <ul style="list-style-type: none"> <li>• Usando el diagrama, armen el circuito en una protoboard conectando LEDs, sensor de sonido y Arduino.</li> <li>• Verifiquen que las conexiones sean firmes y correctas.</li> <li>• Soliciten apoyo al docente para validar el armado antes de continuar.</li> </ul>                                                | 60 minutos | Montaje funcional del circuito en protoboard listo para programar.            | Desarrollar habilidades prácticas en el armado de circuitos electrónicos básicos.       |
| 3. Programación<br>Básica del<br>Semáforo | <ul style="list-style-type: none"> <li>• Escriban un código sencillo en el IDE de Arduino que encienda los LEDs en secuencia según niveles de ruido detectados por el sensor.</li> <li>• Prueben el código y ajusten los umbrales para que el semáforo responda correctamente.</li> <li>• Documenten brevemente qué hace cada parte del código.</li> </ul> | 60 minutos | Código básico cargado en Arduino que controla el semáforo según el ruido.     | Aplicar pensamiento computacional para programar respuestas condicionales con sensores. |
| 4. Prueba y<br>Ajuste Final               | <ul style="list-style-type: none"> <li>• Realicen pruebas de funcionamiento en situaciones reales simulando diferentes niveles de ruido.</li> <li>• Identifiquen errores o mejoras en el circuito y código.</li> <li>• Hagan ajustes necesarios y preparen una breve explicación para compartir con el grupo.</li> </ul>                                   | 30 minutos | Semáforo funcional ajustado y plan de presentación para compartir resultados. | Evaluar y mejorar prototipos mediante pruebas iterativas y trabajo colaborativo.        |

## Cierre - Reflexionar

### Preguntas de Reflexión Metacognitiva para el Cierre

- ¿Qué partes del semáforo de control de ruido con Arduino te parecieron más fáciles o difíciles de entender y por qué?
- ¿Cómo crees que el proyecto que realizamos puede ayudar a controlar los niveles de ruido en diferentes lugares?
- ¿Qué habilidades nuevas aprendiste durante la construcción y programación del semáforo?
- Si tuvieras que explicar a un amigo cómo funciona el semáforo de control de ruido, ¿qué le dirías?

- ¿Qué cambios o mejoras harías a tu proyecto si tuvieras más tiempo o recursos?
- ¿Cómo te sentiste trabajando en un proyecto que combina tecnología y programación? ¿Qué aprendiste sobre trabajar en equipo o resolver problemas?
- ¿Qué estrategias usaste para solucionar los problemas que encontraste al programar o armar el semáforo?

## Actividades de Reflexión Metacognitiva para el Cierre

- **Diario de Aprendizaje:** Pide a los estudiantes que escriban en su cuaderno o en una hoja una breve reflexión donde respondan a algunas de las preguntas anteriores, describiendo qué aprendieron y cómo se sintieron durante el proyecto.
- **Discusión en Parejas:** Organiza a los estudiantes en parejas para que compartan sus respuestas y experiencias sobre el proyecto. Luego, cada pareja puede contar brevemente a la clase un punto interesante que hayan discutido.
- **Mapa Conceptual Reflexivo:** Solicita a los estudiantes que creen un mapa conceptual que incluya los conceptos aprendidos, los pasos para construir el semáforo y las conexiones con la vida diaria, señalando lo que les resultó más importante o sorprendente.
- **Autoevaluación con Rúbrica Simple:** Proporciona una rúbrica sencilla para que los estudiantes evalúen su participación, comprensión y esfuerzo durante el proyecto, acompañada de un espacio para que escriban qué pueden mejorar en futuros proyectos.
- **Compartir Logros y Retos:** En un círculo, invita a cada estudiante a mencionar un logro del proyecto y un reto que enfrentó, fomentando un ambiente de apoyo y aprendizaje colaborativo.

## Cierre - Retroalimentar

### Estrategias de Retroalimentación para el Cierre

Para la sesión de 3 horas del proyecto "¡Silencio en Verde! Diseño y Construcción de un Semáforo de Control de Ruido con Arduino", las estrategias de retroalimentación al cierre deben ser constructivas, específicas y motivadoras, para que los estudiantes de secundaria comprendan sus logros y áreas de mejora en relación con los objetivos del proyecto.

- **Retroalimentación en equipo con autoevaluación guiada**
  - Al finalizar la construcción y prueba del semáforo, cada equipo realiza una breve presentación explicando cómo funcionó su diseño y qué dificultades encontraron.
  - El docente ofrece comentarios específicos sobre el diseño del circuito, la programación y el trabajo colaborativo, resaltando aspectos positivos y sugiriendo mejoras concretas para futuros proyectos.
  - Se guía a los estudiantes para que reflexionen sobre su propio desempeño con preguntas como:
    - ¿Qué parte del proyecto les pareció más fácil y por qué?
    - ¿Qué dificultades tuvieron y cómo las resolvieron?
    - ¿Qué harían diferente en un próximo proyecto?

- **Feedback individual breve y positivo**

- Durante la última media hora, el docente circula para dar retroalimentación personalizada sobre habilidades específicas, como la comprensión del código Arduino, la aplicación del pensamiento computacional o la colaboración en grupo.
- Se enfatizan los progresos individuales, por ejemplo: “Noté que mejoraste mucho en la identificación de los componentes electrónicos” o “Tu contribución para resolver el problema de programación fue muy valiosa”.

- **Uso de rúbrica sencilla y visual para el cierre**

- Se entrega una rúbrica simplificada que evalúa aspectos clave del proyecto: diseño del semáforo, programación, trabajo en equipo y presentación.
- Los estudiantes marcan con colores o stickers su autoevaluación y luego comparan con la evaluación del docente para identificar coincidencias y diferencias.
- Esta herramienta facilita una retroalimentación clara y concreta sobre lo que lograron y lo que pueden mejorar.

- **Dinámica de retroalimentación “Lo que aprendí y lo que me gustaría mejorar”**

- En plenaria, cada estudiante comparte en voz alta al menos un aprendizaje clave del proyecto y un aspecto que desea mejorar.
- El docente complementa con observaciones que refuercen los aprendizajes y sugiera metas concretas para próximos proyectos.
- Esta dinámica ayuda a consolidar el aprendizaje y genera un ambiente positivo para el crecimiento.