

¡Crea y Juega con Scratch! Aprendiendo Pensamiento Computacional

Tecnología e Informática | Pensamiento Computacional | Aprendizaje Basado en Proyectos

Descripción

Este plan de clase invita a los estudiantes de primaria a descubrir y aplicar el pensamiento computacional utilizando Scratch, una plataforma de programación visual divertida y accesible para niños. A través de un proyecto colaborativo, los estudiantes aprenderán a descomponer problemas, diseñar soluciones paso a paso y crear animaciones o juegos sencillos que expresen sus ideas. Este aprendizaje es relevante porque potencia habilidades de lógica, creatividad y resolución de problemas que se utilizan en la vida diaria, desde organizar tareas hasta entender cómo funcionan los juegos y aplicaciones que usan. Además, trabajar en equipo y desarrollar un proyecto tangible les permitirá experimentar el poder de la tecnología como una herramienta para crear y comunicar.

Objetivos de Aprendizaje

- Identificar y explicar los conceptos básicos del pensamiento computacional en la creación de proyectos con Scratch.
- Diseñar y construir una animación o juego sencillo utilizando bloques de programación en Scratch.
- Colaborar en equipo para planificar y desarrollar un proyecto creativo que resuelva un problema o cuente una historia.
- Evaluar y mejorar su proyecto con base en la retroalimentación de compañeros y docente.

Recursos Necesarios

- Computadoras o tablets con acceso a internet y Scratch (<https://scratch.mit.edu>).
- Cuenta de Scratch para cada estudiante o grupo (opcional, puede usarse sin cuenta).
- Pizarrón o rotafolio para explicar conceptos y planificar.
- Hojas impresas con plantillas de diseño para planificar el proyecto (guion gráfico o storyboard).
- Proyector o pantalla para demostraciones.
- Materiales para anotaciones: lápices, colores, borradores.
- Ejemplos impresos o digitales de proyectos simples en Scratch.

Requisitos Previos

- Conocimiento básico del uso de la computadora o tablet (mouse, teclado, navegación básica).
- Habilidades básicas de lectura y comprensión para seguir instrucciones.
- Experiencia previa mínima con actividades grupales y trabajo colaborativo.

- Habilidades para expresarse oralmente para compartir ideas con compañeros.

Actividades

Sesión 1: Descubriendo el Pensamiento Computacional y Scratch

Fase de Inicio

Tiempo estimado: 10 minutos

Propósito de la sesión: Dar la bienvenida, activar conocimientos previos y motivar a los estudiantes para que conozcan qué es el pensamiento computacional y Scratch, preparando el terreno para crear un proyecto.

Activación de conocimientos previos:

- **Docente:** "¿Quién aquí ha jugado un videojuego o ha usado una aplicación en una tableta o teléfono? ¿Qué creen que hace que esos juegos funcionen?"
- **Estudiantes:** Responden brevemente, mencionando juegos o aplicaciones que conocen.

Motivación y enganche:

- **Docente:** "Hoy vamos a aprender a crear nuestros propios juegos y animaciones usando bloques de colores en Scratch, ¡como si fuéramos magos de la computadora!"
- **Estudiantes:** Muestran curiosidad y expectativa.

Contextualización:

- **Docente:** "El pensamiento computacional es como pensar paso a paso para decirle a la computadora qué hacer y resolver problemas. Esto nos ayuda no solo con la tecnología, sino también en la vida diaria, como organizar tareas o armar un rompecabezas."
- **Estudiantes:** Escuchan y relacionan con experiencias propias.

Fase de Desarrollo

Tiempo estimado: 45 minutos

Presentación del contenido:

- **Docente:** Presenta Scratch en la pantalla, mostrando la interfaz básica: escenario, bloques de programación, personajes (sprites) y el área de scripts.
- Explica en lenguaje sencillo los bloques de movimiento y apariencia.

Actividades de aprendizaje activo:

Actividad 1: Explorando Scratch

- **Objetivo específico:** Identificar los elementos básicos de la interfaz de Scratch y ejecutar comandos simples.
- **Instrucciones:**

- **Docente:** "Vamos a abrir Scratch y juntos vamos a mover un personaje. Primero, busquen el bloque que dice 'mover 10 pasos'. Arrástrerlo al área de scripts y presionen la bandera verde."
- **Estudiantes:** En computadoras, exploran y ejecutan el bloque, observando cómo el personaje se mueve.
- **Docente:** "¿Qué pasó con el personaje? ¿Qué creen que significa 'mover 10 pasos'?"
- **Organización:** Individual
- **Producto:** Primer script que mueve un personaje.
- **Tiempo:** 15 minutos
- **Rol docente:** Circula apoyando, haciendo preguntas para que los niños expliquen sus acciones y motivándolos.

Actividad 2: Creando una pequeña historia animada

- **Objetivo específico:** Aplicar bloques básicos para narrar una secuencia simple.
- **Instrucciones:**
 - **Docente:** "Ahora vamos a hacer que nuestro personaje cuente algo. Usen el bloque 'decir' y combínenlo con movimiento para contar una pequeña historia, por ejemplo: 'Hola, soy un gato y bailo'."
 - **Estudiantes:** Experimentan con bloques para que su personaje hable y se mueva.
 - **Docente:** "¿Qué historia crearon? Compartan con un compañero."
- **Organización:** Parejas
- **Producto:** Mini historia animada con bloques de movimiento y diálogo.
- **Tiempo:** 30 minutos
- **Rol docente:** Observa, anima, sugiere mejoras y pregunta sobre el uso de bloques y la secuencia lógica.

Diferenciación:

- Para quienes terminan antes: explorar bloques de sonido y efectos visuales para enriquecer su historia.
- Para quienes necesitan apoyo: ofrecer ayuda directa para arrastrar y conectar bloques, usar ejemplos visuales y acompañamiento individual.

Transición: El docente invita a los estudiantes a compartir sus historias en la próxima sesión para iniciar un proyecto más grande y colaborativo.

Fase de Cierre

Tiempo estimado: 5 minutos

Síntesis: Cada estudiante comparte una idea clave que aprendió hoy, usando una palabra o frase.

Reflexión metacognitiva:

- "¿Qué fue lo que más te gustó de usar Scratch hoy?"
- "¿Cómo crees que el pensamiento paso a paso nos ayuda a crear cosas en la computadora?"
- "¿Qué te gustaría crear con Scratch en las próximas sesiones?"

Retroalimentación: El docente reconoce los esfuerzos y logros, destacando la importancia de la creatividad y la lógica.

Transferencia: Se explica que en la siguiente sesión usarán lo aprendido para diseñar un proyecto en equipo.

Sesión 2: Planificando Nuestro Proyecto en Scratch

Fase de Inicio

Tiempo estimado: 10 minutos

Propósito de la sesión: Recordar lo aprendido y preparar a los estudiantes para planear un proyecto colaborativo con Scratch.

Activación de conocimientos previos:

- **Docente:** "¿Quién recuerda cómo hicimos que el personaje hablara y se moviera? Hoy vamos a pensar en una idea para nuestro proyecto, que puede ser un juego o una historia."
- **Estudiantes:** Responden y comparten brevemente.

Motivación y enganche:

- **Docente:** "Vamos a imaginar que somos creadores de juegos o cuentos animados. ¿Qué les gustaría crear que sea divertido o que ayude a alguien?"
- **Estudiantes:** Expresan ideas y entusiasman.

Fase de Desarrollo

Tiempo estimado: 45 minutos

Presentación del contenido: Introducir la idea de planificar antes de crear, usando un storyboard o guion gráfico sencillo para organizar ideas.

Actividad 1: Lluvia de ideas y selección de proyecto

- **Objetivo específico:** Generar y seleccionar una idea de proyecto para trabajar en grupo.
- **Instrucciones:**
 - **Docente:** "En grupos de 3 o 4, piensen en ideas para un juego o historia en Scratch. Anoten palabras o dibujen lo que les gustaría crear."
 - **Estudiantes:** Discuten, anotan y dibujan ideas en hojas.
 - **Docente:** Circula, hace preguntas: "¿Qué problema resuelve su juego? ¿Quién será el personaje principal?"
- **Organización:** Grupos de 3-4
- **Producto:** Lista o dibujo con idea de proyecto.
- **Tiempo:** 20 minutos
- **Rol docente:** Facilita, guía y ayuda a enfocar ideas.

Actividad 2: Creando el guion gráfico

- **Objetivo específico:** Planificar la secuencia de acciones o escenas del proyecto.
- **Instrucciones:**
 - **Docente:** "Usen la hoja de guion gráfico para dibujar o escribir qué pasará en cada parte de su juego o historia. ¿Qué hará el personaje primero, después y al final?"
 - **Estudiantes:** Completar guion gráfico en equipo.
 - **Docente:** Pregunta: "¿Qué bloques creen que necesitarán para hacer esto?"
- **Organización:** Grupos de 3-4
- **Producto:** Guion gráfico del proyecto.
- **Tiempo:** 25 minutos
- **Rol docente:** Apoya con ejemplos y verifica comprensión.

Diferenciación:

- Para estudiantes que terminan rápido: sugerir agregar detalles o efectos especiales a su guion.
- Para estudiantes que necesitan apoyo: ofrecer asistencia para organizar ideas y dibujos, usar ejemplos concretos.

Transición: El docente invita a los grupos a preparar su plan para comenzar a programar en la siguiente sesión.

Fase de Cierre

Tiempo estimado: 5 minutos

Síntesis: Cada grupo comparte con la clase su idea y guion gráfico de manera breve.

Reflexión metacognitiva:

- "¿Cómo les ayudó planear antes de crear?"
- "¿Qué fue más fácil o difícil al pensar en su proyecto?"

Retroalimentación: El docente resalta la importancia de planificar y felicita la creatividad.

Transferencia: En la próxima sesión comenzarán a programar su proyecto en Scratch.

Sesión 3: Programando Nuestro Proyecto en Scratch

Fase de Inicio

Tiempo estimado: 10 minutos

Propósito de la sesión: Recordar la planificación y preparar para iniciar la programación en Scratch.

Activación de conocimientos previos:

- **Docente:** "¿Quién puede contar qué ideas tiene su proyecto? Recordemos la historia o el juego que planeamos."
- **Estudiantes:** Comparten brevemente.

Motivación y enganche:

- **Docente:** "Hoy vamos a usar bloques para hacer que nuestro proyecto cobre vida. ¡Manos a la obra!"

Fase de Desarrollo

Tiempo estimado: 45 minutos

Presentación del contenido: Introducir bloques de control, eventos y apariencia para enriquecer la programación.

Actividad 1: Programando la primera escena o acción

- **Objetivo específico:** Construir la primera parte del proyecto usando bloques básicos y de control.
- **Instrucciones:**
 - **Docente:** "Cada grupo abrirá Scratch y empezará a programar la primera escena de su proyecto según el guion gráfico."
 - **Estudiantes:** Programan en grupo, usando bloques de movimiento, apariencia y eventos.
 - **Docente:** Observa, pregunta: "¿Qué quiere que haga el personaje primero? ¿Qué pasa cuando toco la bandera verde?"
- **Organización:** Grupos de 3-4
- **Producto:** Código inicial funcional que representa la primera escena.
- **Tiempo:** 25 minutos
- **Rol docente:** Apoya, responde dudas y sugiere soluciones.

Actividad 2: Compartiendo avances y solucionando problemas

- **Objetivo específico:** Evaluar avances y resolver dificultades en programación.
- **Instrucciones:**
 - **Docente:** "Cada grupo mostrará a otro grupo su avance y explicará qué hizo. Escuchen y den una idea para mejorar o solucionar problemas."
 - **Estudiantes:** Presentan y dan retroalimentación entre grupos.
 - **Docente:** Facilita la discusión y ofrece sugerencias.
- **Organización:** Grupos de 3-4 en parejas de grupos
- **Producto:** Retroalimentación escrita o verbal para mejorar el proyecto.
- **Tiempo:** 20 minutos
- **Rol docente:** Modera, guía la crítica constructiva.

Diferenciación:

- Para quienes terminan rápido: agregar sonidos o interacción con teclado.
- Para quienes necesitan apoyo: acompañamiento directo para conectar bloques y comprender errores.

Transición: El docente invita a seguir programando en la próxima sesión y preparar para presentar el proyecto.

Fase de Cierre

Tiempo estimado: 5 minutos

Síntesis: Reflexión rápida con preguntas: "¿Qué aprendimos hoy sobre programar en Scratch?"

Reflexión metacognitiva:

- "¿Qué parte fue más divertida al programar?"
- "¿Qué bloque nuevo aprendimos a usar?"

Retroalimentación: El docente felicita el esfuerzo y recuerda que la práctica mejora la programación.

Transferencia: La próxima sesión continuarán mejorando y finalizando su proyecto.

Sesión 4: Mejorando y Finalizando Nuestro Proyecto

Fase de Inicio

Tiempo estimado: 10 minutos

Propósito de la sesión: Preparar para revisar, mejorar y concluir el proyecto.

Activación de conocimientos previos:

- **Docente:** "¿Qué partes han programado ya? ¿Qué les gustaría mejorar o agregar?"
- **Estudiantes:** Comparten ideas y retos.

Motivación y enganche:

- **Docente:** "Hoy vamos a hacer que nuestro proyecto sea aún mejor y más divertido. ¡Vamos a sorprender a todos!"

Fase de Desarrollo

Tiempo estimado: 45 minutos

Presentación del contenido: Explicar brevemente cómo probar, detectar errores y mejorar el código.

Actividad 1: Prueba y mejora en equipo

- **Objetivo específico:** Identificar errores y aplicar mejoras al proyecto.
- **Instrucciones:**
 - **Docente:** "Prueben su proyecto varias veces, observen qué funciona y qué no, y hagan cambios para que sea mejor."
 - **Estudiantes:** Ejecutan su proyecto, toman notas, y corrigen errores.
 - **Docente:** Apoya en solución de problemas y sugiere ideas para mejorar.
- **Organización:** Grupos de 3-4
- **Producto:** Proyecto corregido y mejorado.
- **Tiempo:** 30 minutos

- **Rol docente:** Asiste, propone retos adicionales para ampliar funcionalidades.

Actividad 2: Preparando la presentación del proyecto

- **Objetivo específico:** Organizar cómo compartir el proyecto con la clase.
- **Instrucciones:**
 - **Docente:** "Decidan quién hablará y qué mostrarán. Practiquen explicar qué hace su proyecto y cómo lo hicieron."
 - **Estudiantes:** Ensayan presentación en grupo.
 - **Docente:** Da consejos para hablar en público y destacar puntos importantes.
- **Organización:** Grupos de 3-4
- **Producto:** Plan de presentación.
- **Tiempo:** 15 minutos
- **Rol docente:** Apoya con recomendaciones y refuerza confianza.

Diferenciación:

- Para estudiantes rápidos: explorar bloques avanzados o personalizar personajes.
- Para estudiantes con dificultades: apoyo individual en corrección y práctica de presentación.

Transición: Se prepara la sesión final para compartir y evaluar los proyectos.

Fase de Cierre

Tiempo estimado: 5 minutos

Síntesis: Resumen de los avances y motivación para la presentación final.

Reflexión metacognitiva:

- "¿Qué mejora hicieron en su proyecto?"
- "¿Qué aprendieron sobre trabajar en equipo?"

Retroalimentación: El docente reconoce el esfuerzo para mejorar y anima a compartir sus logros.

Sesión 5: Presentamos y Reflexionamos sobre Nuestros Proyectos en Scratch

Fase de Inicio

Tiempo estimado: 10 minutos

Propósito de la sesión: Preparar emocionalmente y repasar cómo será la presentación y reflexión final.

Activación de conocimientos previos:

- **Docente:** "Hoy mostraremos a todos lo que creamos. Recordemos cómo hacer una buena presentación y cómo dar retroalimentación amable."
- **Estudiantes:** Expresan expectativas y preparan materiales.

Fase de Desarrollo

Tiempo estimado: 45 minutos

Actividad 1: Presentación de proyectos

- **Objetivo específico:** Comunicar el proyecto creado y explicar su funcionamiento.
- **Instrucciones:**
 - **Docente:** Invita a cada grupo a presentar su proyecto mostrando el juego o historia en Scratch y explicando su diseño.
 - **Estudiantes:** Presentan su proyecto, responden preguntas.
 - **Docente:** Facilita el ambiente respetuoso y hace preguntas para profundizar.
- **Organización:** Plenaria
- **Producto:** Presentación oral con demostración del proyecto.
- **Tiempo:** 35 minutos
- **Rol docente:** Modera, motiva y evalúa participaciones.

Actividad 2: Retroalimentación y reflexión en grupo

- **Objetivo específico:** Evaluar el proceso y producto, y reflexionar sobre aprendizajes.
- **Instrucciones:**
 - **Docente:** "Cada grupo dirá qué aprendió y qué le gustaría mejorar. Los demás pueden decir algo positivo y una sugerencia."
 - **Estudiantes:** Expresan su aprendizaje y escuchan a compañeros.
- **Organización:** Plenaria
- **Producto:** Comentarios orales y reflexión escrita en ficha.
- **Tiempo:** 10 minutos
- **Rol docente:** Guía la reflexión, asegura clima positivo.

Fase de Cierre

Tiempo estimado: 5 minutos

Síntesis: Elaborar un breve mapa mental colectivo con las ideas principales aprendidas sobre pensamiento computacional y Scratch.

Reflexión metacognitiva:

- "¿Qué es lo que más te gustó de crear con Scratch?"
- "¿Cómo te ayudó trabajar en equipo?"
- "¿Para qué crees que sirve el pensamiento computacional en tu vida?"

Retroalimentación: El docente felicita a los estudiantes y los anima a seguir creando y aprendiendo.

Transferencia: Se invita a seguir explorando Scratch en casa o en otros espacios.

Tarea o reto: Crear en casa una animación o historia corta en Scratch para compartir en la próxima clase.

Evaluación

Tipo de evaluación:

- Diagnóstica: Sesión 1, durante la activación de conocimientos previos para conocer las experiencias previas con tecnología y pensamiento computacional.
- Formativa: Durante todas las sesiones, a través de la observación directa, preguntas guía, revisión de guiones gráficos, avances en proyectos y presentaciones intermedias.
- Sumativa: Sesión 5, mediante la presentación final del proyecto, reflexión grupal y evidencia del producto terminado en Scratch.

Criterios de evaluación:

- Identifica y explica conceptos básicos del pensamiento computacional (Objetivo 1).
- Diseña y construye un proyecto funcional en Scratch que refleje una idea organizada (Objetivo 2).
- Participa activamente y colabora en equipo para desarrollar el proyecto (Objetivo 3).
- Recibe y aplica retroalimentación para mejorar su trabajo (Objetivo 4).

Instrumentos sugeridos:

- Lista de cotejo para seguimiento del proceso y participación.
- Rúbrica simple para evaluar el proyecto final (organización, creatividad, funcionalidad).
- Observación directa del docente durante actividades y presentaciones.
- Autoevaluación y coevaluación mediante preguntas guiadas al final del proyecto.

Evidencias de aprendizaje:

- Scripts y proyectos creados en Scratch (producto digital).
- Guion gráfico o storyboard que muestra planificación.
- Presentaciones orales y participación en discusiones.
- Reflexiones escritas o verbales sobre el proceso y los aprendizajes.