

Rúbrica Analítica para Introducción a la Programación: Algoritmos Cotidianos y Scratch

Rúbrica Analítica | Tecnología e Informática | Informática | 4 niveles

Descripción

Esta rúbrica está diseñada para evaluar el desempeño de estudiantes de secundaria (12-15 años) en la creación y comprensión de algoritmos cotidianos y proyectos en Scratch, considerando aspectos técnicos, creativos y de diversidad, equidad e inclusión (DEI).

Rúbrica

Rúbrica Analítica para Introducción a la Programación: Algoritmos Cotidianos y Scratch

Esta rúbrica está diseñada para evaluar el desempeño de estudiantes de secundaria (12-15 años) en la creación y comprensión de algoritmos cotidianos y proyectos en Scratch, considerando aspectos técnicos, creativos y de diversidad, equidad e inclusión (DEI).

Criterio	Excelente	Bueno	Aceptable	Bajo
Comprensión de Algoritmos Cotidianos Capacidad para identificar y describir pasos lógicos en situaciones diarias.	Describe claramente algoritmos completos con pasos detallados y secuenciales sin errores.	Describe algoritmos con pasos mayormente claros y ordenados, con pocos errores menores.	Describe algoritmos con pasos básicos pero con omisiones o desorden en la secuencia.	No logra identificar ni describir los pasos básicos de un algoritmo cotidiano.
Diseño y Programación en Scratch Habilidad para crear proyectos funcionales usando bloques de Scratch.	Programa un proyecto bien estructurado, funcional y creativo usando bloques correctamente.	Programa un proyecto funcional con algunos errores menores y creatividad adecuada.	Programa un proyecto básico con funcionalidad limitada y errores visibles.	El proyecto es incompleto, no funcional o no utiliza correctamente los bloques de Scratch.

Criterio	Excelente	Bueno	Aceptable	Bajo
Uso de Estructuras de Control Aplicación correcta de secuencias, decisiones y bucles en algoritmos y Scratch.	Utiliza secuencias, decisiones y bucles de forma precisa y efectiva en el proyecto.	Utiliza la mayoría de las estructuras correctamente, con pequeños errores en algunos casos.	Utiliza estructuras básicas pero con errores o falta de comprensión en decisiones o bucles.	No utiliza o utiliza incorrectamente las estructuras de control fundamentales.
Creatividad e Innovación Originalidad y aporte personal en la solución y presentación del proyecto.	Presenta ideas originales y creativas que enriquecen el proyecto significativamente.	Incluye elementos creativos que aportan valor, aunque con limitaciones evidentes.	Aplica ideas básicas con poca innovación o repetición de ejemplos vistos.	No demuestra creatividad ni aporta elementos originales al proyecto.
Claridad y Organización Presentación ordenada y clara del proyecto y explicación del algoritmo.	Presenta el trabajo de forma clara, ordenada y fácil de entender, sin confusiones.	Presenta el trabajo con claridad general, aunque con pequeños desordenes o ambigüedades.	Presenta el trabajo con algunas dificultades para seguir la lógica o encontrar información.	Presentación desorganizada o confusa que dificulta la comprensión del proyecto.
Colaboración y Trabajo en Equipo Participación activa y respeto en actividades grupales relacionadas con el proyecto.	Participa activamente, respeta ideas de otros y contribuye al trabajo colectivo efectivamente.	Participa y respeta al grupo, aunque su aporte es limitado en algunas ocasiones.	Participa de forma mínima y ocasionalmente presenta dificultades para colaborar.	No participa ni colabora en el trabajo en equipo o genera conflictos.

Criterio	Excelente	Bueno	Aceptable	Bajo
Incorporación de Diversidad, Equidad e Inclusión (DEI) Consideración de perspectivas diversas y respeto a la inclusión en el proyecto.	Integra explícitamente elementos que reflejan diversidad cultural, de género y accesibilidad en el proyecto.	Muestra alguna consideración hacia la diversidad o inclusión, aunque de forma limitada.	Reconoce la importancia de DEI, pero no logra incorporarla adecuadamente en el proyecto.	No considera ni respeta aspectos de diversidad, equidad o inclusión en su trabajo.
Resolución de Problemas y Depuración Capacidad para identificar y corregir errores en algoritmos y código Scratch.	Detecta y corrige errores de forma autónoma y eficiente, mejorando el proyecto.	Identifica y corrige la mayoría de errores con ayuda mínima.	Reconoce algunos errores pero corrige sólo con ayuda significativa.	No logra identificar ni corregir errores en el algoritmo o código.