

**EXAMEN PARCIAL**  
**RÚBRICA DE TRABAJO Y/O PROYECTO**

Título del trabajo y/o proyecto a realizar	<b>Juego de Colores</b>
<b>Instrucciones</b>	<p><b>Modalidad: Individual</b></p> <ol style="list-style-type: none"><li>1. Revisar y leer los documentos en las carpetas Clase Desarrollada y/o Recursos de la semana 1 a la 8.</li><li>2. Lee cuidadosamente los requerimientos del examen parcial antes de responder. Este se encuentra en la siguiente liga: <a href="#">EXAMEN PARCIAL 2025-2.docx</a></li><li>3. Utiliza el lenguaje Kotlin y Android para completar cada punto.</li><li>4. Subir el examen resuelto a un repositorio Github. Nota: Este repositorio debe ser público, debe usar este repositorio exclusivamente para este examen parcial.</li><li>5. Se valorará de acuerdo a la rúbrica adjunta.</li><li>6. Responder la rúbrica adjunta.</li><li>7. Entregar el examen parcial anexando la liga del repositorio y la rúbrica en la sección correspondiente en google classroom.</li></ol> <p><b>Notas:</b></p> <ul style="list-style-type: none"><li>• Si <b>dos alumnos presentan el mismo código</b> o existe una <b>sospecha de generación de IA</b>, se <b>citará al alumno para que exponga su código, resuelva preguntas, etc;</b> si el alumno <b>no puede explicar su código</b> fuente <b>tendrá cero (00)</b>.</li><li>• Si entrega código utilizando algo <b>distinto</b> a un repositorio <b>Github</b>, ó el proyecto android se <b>encuentra incompleto</b>, tendrá <b>2 puntos menos</b>.</li></ul>

Criterios y puntaje	Bueno (5 puntos)	Regular (3 puntos)	Deficiente (0-1 puntos)
<b>1. Funcionamiento general del juego</b> (Navegación entre fragments, lógica del juego, temporizador, feedback al usuario, ciclo de vida fragmentos, no presenta problemas al rotar pantalla)	El juego es completamente funcional. Navegación fluida entre fragments. La lógica del juego es correcta y clara.	El juego funciona parcialmente. Algunos errores menores o navegación confusa.	El juego no funciona correctamente o no es posible completar la interacción.
<b>2. Interfaz de usuario</b> (ConstraintLayout, visibilidad, claridad, botones, retroalimentación visual)	La interfaz es clara, intuitiva y está bien organizada. Uso adecuado de ConstraintLayout.	La interfaz es comprensible pero con algunos problemas de alineación o visibilidad.	La interfaz es confusa, desorganizada o dificulta la interacción.
<b>3. Calidad del código</b> (Nombres de variables, organización, comentarios, modularidad)	El código está limpio, bien organizado y documentado. Buen uso de funciones y estructuras.	El código cumple pero presenta redundancias o falta de comentarios.	El código es difícil de leer, no está organizado ni documentado.
<b>4. Funcionalidad adicional-obligatoria.</b> (Ej. sonidos, animaciones, niveles, puntuación)	Se implementa correctamente al menos una funcionalidad adicional y mejora la experiencia del usuario.	La funcionalidad adicional existe pero está incompleta o no funciona bien.	No hay funcionalidad adicional o está mal implementada.