

Desarrollo de Interfaces

Proyecto entregable 02 - Codificando con Dart



Autor: Sergi García

Actualizado Septiembre 2025



Licencia



Reconocimiento - No comercial - CompartirIgual (BY-NC-SA): No se permite un uso comercial de la obra original ni de las posibles obras derivadas, la distribución de las cuales se ha de hacer con una licencia igual a la que regula la obra original.

Nomenclatura

A lo largo de este tema se utilizarán diferentes símbolos para distinguir elementos importantes dentro del contenido. Estos símbolos son:

 **Importante**

 **Atención**

 **Interesante**

ÍNDICE

1. Objetivo del proyecto	3
2. Descripción general	3
3. Normas de realización OBLIGATORIAS	3
4. Instrucciones detalladas	4
5. Criterios de evaluación (Apto/No apto)	4
6. Entregables	5

PROYECTO ENTREGABLE 02 - CODIFICANDO CON DART - UNIDAD 07

1. OBJETIVO DEL PROYECTO

Este proyecto tiene como objetivo que el alumnado demuestre de forma individual su competencia en la resolución de problemas algorítmicos, su familiaridad con el lenguaje de programación Dart y su capacidad para comprender, adaptar, documentar y mejorar código basándose en feedback específico.

La evaluación se centrará en la funcionalidad, la calidad del código, su documentación y la comprensión de los conceptos.

2. DESCRIPCIÓN GENERAL

El alumnado deberá completar 15 ejercicios (katas) de la plataforma CodeWars <https://www.codewars.com/> utilizando el lenguaje Dart. **Estos 15 ejercicios deben ser 10 de nivel 8 (más fáciles) y 5 ejercicios MÍNIMO de nivel 7 (algo más difícil).** Una vez completados, se concertará una cita de evaluación con el profesorado para realizar una defensa oral de las soluciones, explicando las mismas y además la realización en directo de otros problemas que indique el profesor.

La calificación final del proyecto será Apto o No Apto.

3. NORMAS DE REALIZACIÓN OBLIGATORIAS

¡ATENCIÓN! ESTE ES UN TRABAJO EstrictAMENTE INDIVIDUAL.

QUEDA TERMINANTEMENTE PROHIBIDO:

- Utilizar cualquier tipo de Asistente de IA Generativa para generar código, explicaciones, comentarios o cualquier parte del proyecto. Esto incluye, pero no se limita a: GitHub Copilot, ChatGPT, DeepSeek, Grok, Gemini, Claude, Gwen, Kimi, etc.
- Cualquier otra herramienta similar basada en modelos de lenguaje grande (LLM).
- Utilizar código escrito por otras personas. Esto incluye:
 - Copiar soluciones de otros usuarios en CodeWars o en cualquier otro foro o página web.
 - Reutilizar código de compañeros de clase (ya sea de la misma edición o de anteriores).
 - Utilizar soluciones obtenidas a través de cualquier medio que no sea el propio trabajo intelectual.
- Recibir ayuda humana externa, excepto para:
 - Consultas específicas al profesorado sobre conceptos, dudas de enunciados o problemas técnicos con las herramientas.
- Se puede y recomienda:
 - Consultar la documentación oficial de Dart (dart.dev).
 - Buscar información en Stack Overflow o foros similares únicamente sobre errores de sintaxis o conceptos específicos del lenguaje, NUNCA sobre la solución específica a un kata.

Cualquier incumplimiento de estas normas supondrá la calificación automática de NO APTO en el proyecto y podrá conllevar medidas disciplinarias según el reglamento académico.

4. INSTRUCCIONES DETALLADAS

a) Selección y Completado de Katas (Fase de Ejecución)

El alumnado es responsable de seleccionar katas que aborden una variedad de conceptos fundamentales de Dart y de programación (ej: listas, mapas, funciones de orden superior, manejo de strings, clases, etc.).

Cada solución debe ser el código que pasa todos los tests propuestos por CodeWars para ese kata en concreto.

El código debe incluir abundantes comentarios que expliquen:

- La lógica general de la solución
- El propósito de funciones y bloques de código complejos
- La justificación de enfoques algorítmicos específicos
- El funcionamiento de estructuras de datos utilizadas

Se debe crear un repositorio Git privado (en GitHub, GitLab o similar) que contenga los 15 programas de Dart, cada uno en su propio archivo (ej: kata_1.dart, kata_2.dart). El nombre del archivo debe permitir identificar fácilmente el kata resuelto (usando el nombre o ID del kata de CodeWars).

El repositorio debe ser compartido en modo de acceso privado con la cuenta del profesorado antes de la sesión de evaluación.

b) Defensa oral y evaluación (Fase de Defensa)

El alumnado deberá concertar una cita con el profesorado para la defensa oral.

Durante la sesión, el profesorado seleccionará de forma aleatoria o dirigida un subconjunto de los 15 katas presentados.

El alumnado deberá estar preparado para:

- Explicar su solución: Desglosar el código, justificar la lógica utilizada y el porqué de las estructuras de datos y funciones elegidas.
- Recibir feedback y realizar modificaciones en tiempo real: El profesorado podrá solicitar modificaciones concretas al código (ej: "¿Cómo harías para que fuera más eficiente?", "Refactoriza este bucle usando una función de orden superior", "Modifica la solución para que maneje este caso límite").
- Demostrar comprensión: Responder a preguntas teóricas sobre los conceptos de Dart aplicados en sus soluciones.
- Justificar los comentarios: Explicar por qué se incluyeron comentarios específicos y cómo ayudan a la comprensión del código.

En la misma sesión, el profesor pedirá que resuelvan en directo uno o varios ejercicios de Codewars, con el fin de evaluar la aplicación práctica de los conocimientos adquiridos en programación.

5. CRITERIOS DE EVALUACIÓN (APTO/NO APTO)

Para obtener la calificación de APTO, el estudiante debe demostrar:

- Autenticidad y originalidad: El trabajo debe ser claramente propio y original, sin rastros de uso de IA o ayuda no permitida.
- Completitud: Haber resuelto correctamente los 15 katas, con código que supere todos los tests en CodeWars. Estos 15 ejercicios deben ser 10 de nivel 8 (más fáciles) y 5 ejercicios mínimo de nivel 7 (algo más difícil)
- Documentación: El código debe incluir comentarios abundantes y significativos que faciliten

la comprensión de las soluciones.

- Comprensión: Ser capaz de explicar claramente la lógica y el funcionamiento de las soluciones seleccionadas por el profesorado.
- Capacidad de Adaptación: Poder modificar su código siguiendo las indicaciones del profesorado durante la sesión de defensa, demostrando flexibilidad y dominio práctico del lenguaje.
- Calidad del Código: El código debe estar formateado correctamente, ser legible, seguir las convenciones básicas de Dart y estar adecuadamente comentado.
- Capacidad de resolver nuevos problemas de Codewars en directo ante el profesorado.
- **Resolución práctica en directo:** Capacidad para resolver uno o varios ejercicios de Codewars de nivel mínimo 7 durante la defensa oral, aplicando correctamente conceptos y técnicas aprendidas.

La calificación de NO APTO se asignará si:

- Se detecta el uso de IA o ayuda no permitida.
- No se completan los 15 katas.
- El código no supera los tests de CodeWars.
- El código carece de comentarios adecuados o estos son insuficientes para entender las soluciones.
- El alumnado no es capaz de explicar sus propias soluciones.
- El alumnado no puede realizar modificaciones simples solicitadas por el profesorado durante la defensa.
- El repositorio no es accesible para el profesorado en el momento de la evaluación.
- **Si no logra resolver en directo uno o varios ejercicios de Codewars** que le pide el profesor.

6. ENTREGABLES

- Repositorio Git Privado: Conteniendo los 15 archivos .dart con las soluciones adecuadamente comentadas. Debe estar compartido con el profesorado con permisos de acceso necesarios para su revisión.
- Asistencia y participación activa en la sesión de defensa oral en la fecha y hora acordadas.

Consejo: Este proyecto está diseñado para evaluar tu aprendizaje y tu capacidad para resolver problemas. Los comentarios no deben ser redundantes (evitar "incrementa i en 1") sino explicativos ("// Encuentra el primer número primo en el rango usando la Criba de Eratóstenes").

La mejor preparación es practicar explicando tu código en voz alta y entender profundamente por qué tomaste cada decisión de codificación.