

Fish Collector: Idle

2D · IDLE · INCREMENTAL · ONLINE · FREE

Videojuego incremental 2D con servidor Firestore

Godot 4.6

Firebase

Android · Windows

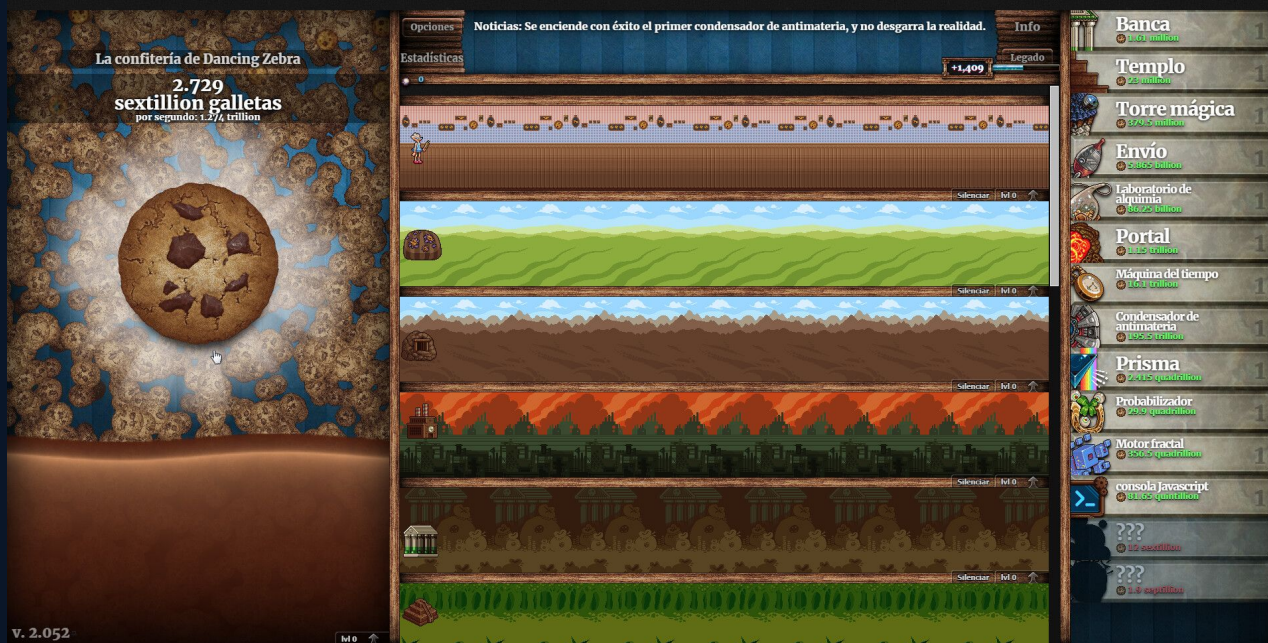
Autor: Joel Lozano Barbancho | **Tutor:** Luis Elía

github.com/joellozanobarbancho/FishCollectorIdle

Introducción y contexto:

¿Qué es un juego Idle?

Género de videojuegos en que, por medio de la repetición continuada de acciones simples, se consigue un bien virtual.



Introducción y contexto:



Objetivos del Proyecto:

Objetivo general: Crear un videojuego 2D idle multiplataforma con funcionalidades sociales en tiempo real, respaldado por Firebase como backend sin servidor propio.

Objetivos específicos:

1 **Mecánica de pesca**

2 **Inventario de peces**

3 **Sistema de Progresión**

4 **Backend Firebase**

5 **Chat de jugadores en tiempo real**

6 **Mercado de intercambio**

Tecnologías utilizadas:

Godot 4.6 + GDScript



- Motor open-source orientado al 2D
- Sintaxis sencilla, similar a Python
- Exporta a Windows y Android de forma nativa

Firebase Auth + Firestore



- Autenticación
- Base de datos NoSQL en la nube
- Listeners en tiempo real sin gestionar infraestructura propia

Otras herramientas:

Figma + Taiga + GitHub



- Herramientas gratuitas
- Diseño UI
- Gestión de trabajo
- Control de versiones y release v1.0

Metodología de desarrollo:

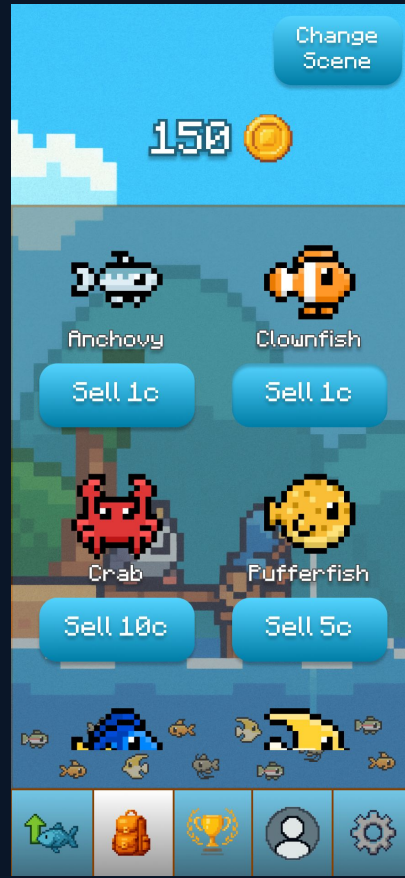
The image shows a Kanban board interface with the following structure:

- Header:** "Kanban" title and a search filter "Filters" with the placeholder "subject or reference".
- Columns:** Five columns representing workflow stages: "NEW", "READY", "IN PROGRESS", "READY FOR TEST", and "DONE". Each column has a count of items in the top right corner.
- Task Cards:** Each card includes a task ID, a description, and a small green bird icon. The tasks are numbered 1 through 20.
- Task Distribution:**
 - NEW (1 item):** #20 Mejoras elementos UI + animaciones
 - READY (4 items):** #16 Implementación sistema de logros, #17 Implementación sistema de tradeo, #18 Implementación opciones de juego, #19 Añadir sonidos y efectos
 - IN PROGRESS (1 item):** #15 Implementación sistema de tienda / mejoras
 - READY FOR TEST (1 item):** #14 Implementación chat global
 - DONE (11 items):** #1 Crear repositorio de Github, #2 Estructura de proyecto, #3 Estructura de datos (Json), #11 Prototipo Frontend Ionic+VUE, #8 Definir sistema de tienda / mejoras, #9 Definir sistema de logros, #10 Definir atributos de usuario
- UI Elements:** A vertical sidebar on the left contains navigation icons (home, search, etc.). A vertical bar on the right is labeled "ARCHIVED (ARCHIVED)".

Metodología de desarrollo:

Fases de Desarrollo

- F1 Investigación y definición del proyecto
- F2 Diseño conceptual y técnico
- F3 Prototipado técnico
- F4 Desarrollo del núcleo jugable
- F5 Integración del servidor y funcionalidades sociales
- F6 Misiones, logros y contenido adicional
- F7 Pruebas y validación
- F8 Documentación y entrega (release v1.0)



Arquitectura y diseño:

CLIENTE · Godot 4.6

FirestoreManager.gd

Auth via godot-firebase, REST HTTP

File.gd / Data.gd

Persistencia local

DataManager.gd

Bases de datos JSON

UpgradeManager.gd

Mejoras y sus efectos

InventoryManager.gd

Adición, venta y consumo de peces

QuestManager.gd / LevelManager.gd

Misiones, XP y niveles

SERVIDOR · Firebase

Firestore Auth

Registro, login y recuperación contraseña

Firestore: user_saves

Archivos de guardado

Firestore: global_chat_messages

Registro del chat en tiempo real

Firestore: online_presence

Heartbeat de presencia online

Firestore: trade_offers

Mercado de jugadores

Demostración del Juego:



Menú principal

Registro/Inicio de sesión
Reseteo de contraseña



Pantalla de pesca

Pantalla principal del juego
HUD nivel/stamina/monedas
Cambio de hábitat



Tienda de mejoras

Fishing Rod
Bait Box
Fishing Hook
Stamina Charm



Inventario

Sell · Sell All · Eat · Eat All
Sincronizado con Firestore



Chat global

Foro de mensajes en tiempo real
Contador de jugadores online



Mercado de intercambio

Ofertas activas con peces
ofrecidos/solicitados



Menú de opciones

Salir del juego
Eliminar datos de guardado



Fishpedia

Enciclopedia de peces

Evaluación y conclusiones:

Consecución de objetivos

Objetivo	Resultado	Estado
Videojuego 2D multiplataforma	Juego funcional, exportaciones a Windows + Android	✓ Conseguido
Backend Firebase	Auth, Firestore y guardado en la nube	✓ Conseguido
Chat global en tiempo real	Listener Firestore, reglas de seguridad activas	✓ Conseguido
Mercado de intercambio	Publicar, aceptar, validación de transacciones	✓ Conseguido

Trabajo a futuro:



Publicación en itch.io



Ampliación de contenido



Sistema de amigos y rankings



Modo offline



Push notifications (FCM)



Escalado al plan Blaze de Firebase

Muchas gracias por vuestra atención

¿Preguntas?

Joel Lozano Barbancho · github.com/joellozanobarbancho/FishCollectorIdle

Institut Puig Castellar · DAM2A · 2025-2026 · Tutor: Luis Elía

