**1.0 Introducción**

El presente documento es un informe técnico que detalla la funcionalidad, capacidades y, críticamente, las limitaciones de los planes gratuitos de tres plataformas: **GDevelop** (motor de creación), **Replit** (entorno de desarrollo en nube) y **Render** (plataforma de alojamiento/PaaS).

El objetivo principal de esta investigación fue validar una solución al problema de la **suspensión por inactividad** (tiempo de 5-10 minutos) observada en Replit, que hace inviable la operación continua de un proyecto. Se evalúa Render como alternativa de alojamiento.

**2.0 Análisis de GDevelop**

GDevelop es una *suite* de creación, un motor de juegos. Su paradigma operativo no se basa en código de programación tradicional, sino en un sistema de lógica visual.

**2.1 Modelo de Operación**

* **Lógica de Eventos:** El sistema funciona mediante hojas de eventos que siguen una estructura de "Condición -> Acción". Es un enfoque declarativo.
* **Ejemplo:**
  + **Condición:** El objeto 'Jugador' está en colisión con el objeto 'Enemigo'.
  + **Acción:** Reiniciar la escena.
* **Exportación:** Es una aplicación local (aunque tiene versión web). El producto final del trabajo es un *paquete* de archivos, comúnmente HTML5 (HTML, CSS, JavaScript) para la web, o compilaciones para escritorio (Windows, Linux).

**2.2 Análisis del Nivel Gratuito y Limitaciones**

El motor en sí es de código abierto y gratuito. Las limitaciones no residen en la capacidad creativa, sino en los servicios de *empaquetado* y *distribución* que ofrece la empresa detrás del software.

* **Funcionalidad Completa:** Acceso total al editor de eventos, objetos, comportamientos y exportación local.
* **Limitación 1 (Compilación):** Las compilaciones en la nube (para crear un .apk de Android, por ejemplo) están severamente restringidas a un número bajo por día (aprox. 2-10). La exportación a HTML5 es local e ilimitada.
* **Limitación 2 (Identidad):** Todas las exportaciones gratuitas incluyen una pantalla de inicio forzosa (un *splash screen*) con el logo de GDevelop.

**3.0 Análisis de Replit**

Replit no es una plataforma "no-code". Es un IDE (Entorno de Desarrollo Integrado) completo que opera en la nube. Provee un editor de código, una terminal y un servidor virtual (un "Workspace").

**3.1 Modelo de Operación**

* Se crea un proyecto ("Repl") en un lenguaje específico (ej. Python, Node.js).
* Replit asigna recursos de cómputo (CPU/RAM) compartidos para ejecutar dicho código.
* Es excelente para prototipado rápido, ya que el entorno de ejecución está acoplado al de edición.

**3.2 Análisis del Nivel Gratuito y Limitaciones**

El plan gratuito está diseñado para experimentación, no para producción.

* **Limitación 1 (Recursos):** Los "Workspaces" tienen especificaciones muy bajas (ej. 0.5 GiB de RAM, CPU compartida).
* **Limitación 2 (Persistencia):** Este es el punto crítico confirmado. Los "Repls" que ejecutan un servidor (un backend, un bot) entran en **estado de suspensión (sleep)** tras un período de inactividad de **5 a 10 minutos**.
* **Consecuencia:** El servicio deja de funcionar. Para "despertarlo", debe recibir una solicitud HTTP, lo que incurre en un tiempo de carga prolongado (*cold start*).
* **Función "Always On":** Esta funcionalidad, que prevenía la suspensión, fue movida a los planes de pago y ahora requiere el uso de "Cycles" (moneda interna).

**4.0 Análisis de Render**

Render es una Plataforma como Servicio (PaaS) moderna, un competidor directo de Heroku. Su función es tomar código fuente (generalmente desde un repositorio Git) y alojarlo.

**4.1 Modelo de Operación**

* Render se vincula a un repositorio (GitHub, GitLab).
* Detecta automáticamente el tipo de proyecto (Node.js, Python, o un sitio estático).
* Realiza la compilación (*build*) y el despliegue (*deploy*).
* Proporciona una URL pública y gestiona el servidor.

**4.2 Análisis del Nivel Gratuito y Limitaciones**

El análisis de Render es complejo, ya que ofrece dos tipos de servicios gratuitos con políticas de inactividad diametralmente opuestas.

**4.2.1 Servicio Gratuito: Sitios Estáticos (Static Sites)**

* **Qué aloja:** Archivos que no requieren ejecución de backend. Esto incluye HTML/CSS/JS, o **un juego de GDevelop exportado a HTML5**.
* **Política de Inactividad:** **NO SE SUSPENDEN.** Los sitios estáticos están siempre activos.
* **Funciones Adicionales:**
  + CDN Global (Red de Distribución de Contenido) para alta velocidad.
  + 100 GB de ancho de banda mensual.
  + Dominios personalizados y HTTPS automático.
* **Limitaciones:** No puede ejecutar código de servidor (como un bot de Python).

**4.2.2 Servicio Gratuito: Servicios Web (Web Services)**

* **Qué aloja:** Aplicaciones de backend (Python, Node.js, Go, Ruby). Un bot de Discord o una API RESTful iría aquí.
* **Política de Inactividad:** **SÍ SE SUSPENDEN.** El servicio entra en estado de suspensión tras **15 minutos de inactividad**.
* **Consecuencia:** Similar a Replit, pero con más tiempo de gracia. La primera solicitud después de la suspensión sufrirá un "arranque en frío" (*cold start*) notable (20-30 segundos).
* **Limitaciones Adicionales:**
  + Bases de datos (PostgreSQL) gratuitas **se eliminan automáticamente después de 90 días**. Son estrictamente para pruebas.
  + Recursos de cómputo compartidos (512 MB RAM).

**5.0 Tabla Comparativa de Persistencia Gratuita**

| **Plataforma** | **Tipo de Servicio** | **¿Se Suspende por Inactividad?** | **Tiempo de Inactividad (Aprox.)** |
| --- | --- | --- | --- |
| GDevelop | Motor de Juego | No aplica (Software local) | N/A |
| Replit | IDE en la Nube (Backend) | **Sí** | 5 - 10 minutos |
| Render | Hosting Estático (HTML5) | **No** | N/A (Siempre activo) |
| Render | Hosting de Backend (Python/Node) | **Sí** | 15 minutos |

**6.0 Conclusión del Informe**

El análisis técnico confirma la problemática de Replit (suspensión en 5-10 minutos).

Render presenta una solución viable, pero **depende críticamente del tipo de proyecto**:

1. Para **proyectos estáticos** (como un juego de GDevelop exportado a web), el plan "Static Sites" de Render es una solución definitiva y superior, ya que **no tiene política de suspensión** y ofrece una CDN global gratuita.
2. Para **proyectos dinámicos** (backends, bots, APIs), el plan "Web Services" de Render **no soluciona el problema de raíz**. Simplemente extiende el tiempo de inactividad a 15 minutos, pero el servicio se sigue suspendiendo.

Para mantener un proyecto *dinámico* (un bot) activo 24/7, los planes gratuitos de Replit y Render son insuficientes debido a sus políticas de suspensión por inactividad.