

# Guía de Configuración de Credenciales y Variables de Entorno para QodeIA

Este documento detalla las credenciales y variables de entorno necesarias para el correcto funcionamiento de los tres repositorios de QodeIA: Mi-agente-QodeIA- , Plataforma-qd y Web-QodeIA- .

## 1. Variables de Entorno Comunes y Específicas

A continuación, se presenta una tabla con todas las variables de entorno identificadas, su propósito, el repositorio al que pertenecen y si requieren configuración manual o están pre-configuradas.

Variable de Entorno	Propósito	Repositorio(s)	Tipo de Configuración	Notas
NEXT_PUBLIC_SUPABASE_URL	URL de Supabase para el Agente QodeIA	Mi-agente-QodeIA- , Web-QodeIA-	Pre-configurada	https://nknevqndawnokiaickkl.supabase.co
NEXT_PUBLIC_SUPABASE_ANON_KEY	Clave pública de Supabase para el Agente QodeIA	Mi-agente-QodeIA- , Web-QodeIA-	Pre-configurada	Clave anónima del proyecto Supabase del Agente
SUPABASE_SERVICE_ROLE_KEY	Clave de rol de servicio de Supabase para el Agente QodeIA	Mi-agente-QodeIA-	Manual	Obtener del dashboard de Supabase (Configuración -> API Keys -> service_role secret key )
HOWARD_OS_SUPABASE_URL	URL de Supabase para Howard OS (Base de Conocimiento)	Mi-agente-QodeIA-	Pre-configurada	https://tztypjxqklxygfzbpkmm.supabase.co
HOWARD_OS_SUPABASE_KEY	Clave de Supabase para Howard OS (Base de Conocimiento )	Mi-agente-QodeIA-	Pre-configurada	Clave anónima del proyecto Supabase de Howard OS

NEXT_PUBLIC_KNOWLEDGE_SUPABASE_URL	URL de Supabase para Howard OS (Base de Conocimiento)	Web-QodeIA-	Pre-configurada	<a href="https://tztypjxqklxygfzbpkmm.supabase.co">https://tztypjxqklxygfzbpkmm.supabase.co</a>
NEXT_PUBLIC_KNOWLEDGE_SUPABASE_ANON_KEY	Clave pública de Supabase para Howard OS (Base de Conocimiento )	Web-QodeIA-	Pre-configurada	Clave anónima del proyecto Supabase de Howard OS
VITE_SUPABASE_URL	URL de Supabase para Howard OS (Base de Conocimiento)	Plataforma-qd	Pre-configurada	<a href="https://tztypjxqklxygfzbpkmm.supabase.co">https://tztypjxqklxygfzbpkmm.supabase.co</a>
VITE_SUPABASE_ANON_KEY	Clave pública de Supabase para Howard OS (Base de Conocimiento )	Plataforma-qd	Pre-configurada	Clave anónima del proyecto Supabase de Howard OS
VITE_OPERATIONAL_SUPABASE_URL	URL de Supabase para el Agente QodeIA	Plataforma-qd	Pre-configurada	<a href="https://nknevqndawnokiaickkl.supabase.co">https://nknevqndawnokiaickkl.supabase.co</a>
VITE_OPERATIONAL_SUPABASE_ANON_KEY	Clave pública de Supabase para el Agente QodeIA	Plataforma-qd	Pre-configurada	Clave anónima del proyecto Supabase del Agente
OPENAI_API_KEY	Clave API de OpenAI	Mi-agente-QodeIA-	Manual	Tu clave API personal de OpenAI
GOOGLE_CLIENT_ID	ID de Cliente de Google OAuth	Mi-agente-QodeIA- , Web-QodeIA-	Manual	Obtener de Google Cloud Console
GOOGLE_CLIENT_SECRET	Secreto de Cliente de Google OAuth	Mi-agente-QodeIA-	Manual	Obtener de Google Cloud Console

NEXT_PUBLIC_URL	URL pública de la aplicación web	Mi-agente-QodeIA- , Web-QodeIA-	Pre-configurada	https://web-qode-ia.vercel.app
NEXT_PUBLIC_AGENT_URL	URL del Agente QodeIA	Web-QodeIA-	Pre-configurada	https://mi-agente-qode-ia.vercel.app
NEXT_PUBLIC_HOWARD_OS_URL	URL de Howard OS (Plataforma QD )	Mi-agente-QodeIA- , Web-QodeIA-	Pre-configurada	https://plataforma-qd.vercel.app
NEXT_PUBLIC_IDE_URL	URL del IDE (Plataforma QD )	Web-QodeIA-	Pre-configurada	https://plataforma-qd.vercel.app
NEXT_PUBLIC_SITE_URL	URL principal del sitio web	Web-QodeIA-	Pre-configurada	https://qodeia.com
NEXT_PUBLIC_CONTACT_EMAIL	Correo electrónico de contacto	Web-QodeIA-	Pre-configurada	qodeia_info@gmail.com
VERCEL_TOKEN	Token de API de Vercel	Mi-agente-QodeIA-	Manual	Tu token de API personal de Vercel
VERCEL_TEAM_ID	ID del equipo de Vercel	Mi-agente-QodeIA-	Pre-configurada	team_JAdXWfQ7CTEn4X65PX7iNJ5E
VERCEL_PROJECT_ID	ID del proyecto Vercel	Mi-agente-QodeIA-	Pre-configurada	prj_He7Xk8zyji0mdREOS2IB20H7uKUH
GITHUB_OWNER	Propietario del repositorio de GitHub	Mi-agente-QodeIA-	Pre-configurada	dgr198213-ui
GITHUB_REPO	Nombre del repositorio de GitHub	Mi-agente-QodeIA-	Pre-configurada	Mi-agente-QodeIA-
GITHUB_TOKEN	Token de GitHub	Mi-agente-QodeIA-	Manual	Tu token de acceso personal de GitHub

HOWARD_OS_NOTEBOOK_URL	URL del notebook de Howard OS en NotebookLM	Mi-agente-QodeIA-	Manual	URL específica del notebook
SOLUCIONES_NOTEBOOK_URL	URL del notebook de Soluciones en NotebookLM	Mi-agente-QodeIA-	Manual	URL específica del notebook
NOTEBOOKLM_COOKIE	Cookie de autenticación de NotebookLM	Mi-agente-QodeIA-	Manual	Cookie de sesión para autenticación
VITE_ENCRYPTION_KEY	Clave de cifrado para Plataforma QD	Plataforma-qd	Pre-configurada	Clave de cifrado interna
VITE_AGENT_URL	URL del Agente QodeIA	Plataforma-qd	Pre-configurada	https://mi-agente-qode-ia.vercel.app

## 2. Configuración en Vercel

Los proyectos de QodeIA están diseñados para ser desplegados en Vercel. La configuración de las variables de entorno se puede realizar de dos maneras:

### Opción 1: Usando los Scripts Automáticos `configure-vercel-env.sh`

Cada repositorio contiene un script `configure-vercel-env.sh` que automatiza la adición de la mayoría de las variables de entorno a tu proyecto Vercel. Para usarlo:

1. Navega al directorio raíz de cada repositorio clonado en tu terminal.
2. Ejecuta el script:

Bash

```
./configure-vercel-env.sh
```

**Nota:** Este script intentará añadir las variables para los entornos `production`, `preview` y `development`. Si una variable ya existe, mostrará una advertencia.

### Opción 2: Configuración Manual en el Dashboard de Vercel

Para las variables que requieren configuración manual (marcadas como "Manual" en la tabla anterior ) o si prefieres una configuración manual, sigue estos pasos:

1. Ve a [Vercel Dashboard](#).
2. Selecciona el proyecto correspondiente ( `mi-agente-qode-ia` , `plataforma-qd` o `web-qode-ia` ).
3. Navega a `Settings` → `Environment Variables` .
4. Añade cada variable manualmente, asegurándote de seleccionar los entornos ( `Production` , `Preview` , `Development` ) donde debe estar disponible.

## 3. Configuración Específica de Credenciales Manuales

### 3.1. Supabase `SUPABASE_SERVICE_ROLE_KEY`

Esta clave es crucial para operaciones de backend con privilegios elevados. Debes obtenerla directamente desde tu proyecto Supabase:

1. Inicia sesión en tu [Dashboard de Supabase](#).
2. Selecciona el proyecto del Agente QodeIA ( `nknevqndawnokiaickkl` ).
3. Ve a `Project Settings` (icono de engranaje) → `API Keys` .
4. Busca la clave etiquetada como `service_role` (también conocida como `secret key`). Cópiala y configúrala como `SUPABASE_SERVICE_ROLE_KEY` en tu proyecto Vercel `Mi-agente-QodeIA-` .

### 3.2. OpenAI `OPENAI_API_KEY`

Para que el Agente QodeIA pueda interactuar con los modelos de OpenAI, necesitas tu clave API:

1. Inicia sesión en tu [cuenta de OpenAI](#).
2. Genera una nueva clave secreta si no tienes una.
3. Copia la clave y configúrala como `OPENAI_API_KEY` en tu proyecto Vercel `Mi-agente-QodeIA-` .

### 3.3. Google OAuth ( `GOOGLE_CLIENT_ID` , `GOOGLE_CLIENT_SECRET` )

La autenticación con Google OAuth es necesaria para ciertas funcionalidades. Sigue estos pasos para obtener las credenciales:

1. Ve a [Google Cloud Console](#).
2. Crea un nuevo proyecto o selecciona uno existente.

3. Habilita la API de Google OAuth 2.0 (si no está habilitada).
4. Navega a `APIs & Services` → `Credentials` .
5. Haz clic en `CREATE CREDENTIALS` y selecciona `OAuth client ID` .
6. Configura el tipo de aplicación como `Web application` .
7. Añade los siguientes `Authorized JavaScript origins` y `Authorized redirect URIs` :
  - **Authorized JavaScript origins:** `https://web-qode-ia.vercel.app`
  - **Authorized redirect URIs:** `https://web-qode-ia.vercel.app/api/mcp/auth/callback`
8. Copia el `Client ID` y el `Client Secret` generados.
9. Configura `GOOGLE_CLIENT_ID` y `GOOGLE_CLIENT_SECRET` en tu proyecto Vercel `Mi-agente-QodeIA-` .

### 3.4. Vercel `VERCEL_TOKEN`

El `VERCEL_TOKEN` es necesario para que el Agente QodeIA pueda interactuar con la API de Vercel (por ejemplo, para actualizar variables de entorno o desplegar ).

1. Ve a [Vercel Account Settings](#).
2. Genera un nuevo token de API personal.
3. Copia el token y configúralo como `VERCEL_TOKEN` en tu proyecto Vercel `Mi-agente-QodeIA-` .

### 3.5. GitHub `GITHUB_TOKEN`

Si el Agente QodeIA necesita interactuar con GitHub (por ejemplo, para leer o escribir código), necesitarás un token de acceso personal de GitHub.

1. Ve a [GitHub Developer Settings](#).
2. Genera un nuevo token de acceso personal (clásico) con los permisos adecuados (por ejemplo, `repo` ).
3. Copia el token y configúralo como `GITHUB_TOKEN` en tu proyecto Vercel `Mi-agente-QodeIA-` .

### 3.6. NotebookLM ( `HOWARD_OS_NOTEBOOK_URL` , `SOLUCIONES_NOTEBOOK_URL` , `NOTEBOOKLM_COOKIE` )

Estas variables son para la integración con NotebookLM. Las URLs de los notebooks deben ser las URLs directas a tus notebooks específicos en NotebookLM. La `NOTEBOOKLM_COOKIE` es una cookie de autenticación que permite el acceso programático. Esta última es una credencial sensible y debe manejarse con extrema precaución.

1. **HOWARD\_OS\_NOTEBOOK\_URL** y **SOLUCIONES\_NOTEBOOK\_URL** : Navega a tus notebooks en NotebookLM y copia la URL completa de cada uno.
2. **NOTEBOOKLM\_COOKIE** : Esta es una credencial avanzada y sensible. Generalmente, se obtiene inspeccionando las cookies de tu navegador mientras estás autenticado en NotebookLM. Busca la cookie de sesión relevante. **Advertencia:** Compartir o exponer esta cookie puede comprometer tu cuenta de NotebookLM. Asegúrate de que solo sea accesible por el Agente QodelA en un entorno seguro.

## 4. Verificación de Configuración

Después de configurar todas las variables, es recomendable verificar que todo esté funcionando correctamente:

- **Verificar Variables de Entorno en Vercel:** Puedes usar el comando `vercel env ls` en tu terminal (después de instalar Vercel CLI y autenticarte) para listar las variables configuradas en tu proyecto Vercel.
- **Probar Conexión con Supabase:** Desde la consola del navegador de tu aplicación `Web-QodelA-`, puedes ejecutar `console.log(process.env.NEXT_PUBLIC_SUPABASE_URL)` para verificar que la URL de Supabase se carga correctamente.
- **Probar Conexión con Agente:** Puedes intentar acceder al endpoint del agente desde tu navegador o con `curl` :

Bash

```
curl https://web-qode-ia.vercel.app/api/agent
```

- **Probar Conexión con Howard OS:** De manera similar, puedes probar el endpoint de Howard OS:

Bash

```
curl https://plataforma-qd.vercel.app
```

## 5. Próximos Pasos y Consideraciones

1. Asegúrate de que todos los proyectos estén desplegados en Vercel después de configurar las variables de entorno. Puedes forzar un redespliegue con `vercel --prod` desde el directorio de cada proyecto.
2. Mantén tus claves API y secretos seguros. No los expongas en el código fuente ni en repositorios públicos.

3. Revisa periódicamente los permisos de tus tokens (GitHub, Vercel ) y claves API (OpenAI, Google) para asegurarte de que solo tengan los accesos mínimos necesarios.

Con esta guía, deberías poder configurar todas las credenciales y variables de entorno para que tus aplicaciones QodelA funcionen en su totalidad.